

MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
NATIONAL TECHNICAL UNIVERSITY OF UKRAINE
“IGOR SIKORSKY KYIV POLYTECHNIC INSTITUTE”

COMPUTER GRAPHICS IN AUTOCAD: AUTOMATION WITH VISUAL BASIC FOR APPLICATION

Textbook

Approved by the Academic Council of the Igor Sikorsky Kyiv Polytechnic Institute
as a textbook for bachelor's applicants
on the specialty E6 “Applied Physics and Nanomaterials”

Electronic network educational edition

KYIV
IGOR SIKORSKY KYIV POLYTECHNIC INSTITUTE
2026

UDC 744:004.925.8(075.8)

C63

- Authors: *Gennadii Virchenko*, D.Sc. (Engineering), Professor
Hanna Shepel
Olga Lebedieva
Petro Yablonskyi, Ph.D. (Engineering), Associate Professor
Tetiana Nadkernychna
- Reviewer *Svitlana Botvinovska*, D.Sc. (Engineering), Professor
Head of the Department of Descriptive Geometry and Engineering
Graphics, Kyiv National University of Construction and Architecture
Yaroslav Kornaha, D.Sc. (Engineering), Professor
Dean of the Faculty of Informatics and Software Engineering,
National Technical University of Ukraine
“Igor Sikorsky Kyiv Polytechnic Institute”
Victor Nesvidomin, D.Sc. (Engineering), Professor
Professor of the Department of Descriptive Geometry,
Computer Graphics and Design, National University of Life and
Environmental Sciences of Ukraine
- Responsible editor *Olha Holova*, Ph.D. (Engineering), Associate Professor

*The approval was granted by the Academic Council of Igor Sikorsky Kyiv Polytechnic Institute
(minutes of meeting No. 2 dated 09.02.2026)*

- C63 **Computer** Graphics in AutoCAD: Automation with Visual Basic for Applications
[Electronic resource] : textbook for bachelor's applicants in the specialty E6 “Applied
Physics and Nanomaterials” / G. Virchenko, H. Shepel, O. Lebedieva, P. Yablonskyi,
T. Nadkernychna ; Igor Sikorsky Kyiv Polytechnic Institute. – Electronic text data
(1 file). – Kyiv : Igor Sikorsky Kyiv Polytechnic Institute, Publishing House
“Polytechnica”, 2026. – 292 p.

ISBN 978-966-990-258-0

The textbook presents the general foundations of modern computer-aided shape generation, including the use of geometric models and methods for constructing various lines, surfaces, and solids. The corresponding graphical tools are illustrated using the AutoCAD system, emphasizing the possibilities for increasing the level of automation through the Visual Basic for Applications programming language. The material is linked to the training exercises provided in the appendices. Compared with other publications, the advantage of this textbook lies in the integration of the foundations of computer-aided shape modeling with practical techniques for their implementation and automated execution through user-developed programs.

Textbook for bachelor's degree students in Specialty E6 Applied Physics and Nanomaterials.

UDC 744:004.925.8(075.8)

ISBN 978-966-990-258-0

- © G. Virchenko, H. Shepel, O. Lebedieva,
P. Yablonskyi, T. Nadkernychna, 2026
© Igor Sikorsky Kyiv Polytechnic Institute, 2026

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

КОМП'ЮТЕРНА ГРАФІКА В AutoCAD: АВТОМАТИЗАЦІЯ ЗАСОБАМИ VISUAL BASIC FOR APPLICATIONS

Підручник

Затверджено Вченою радою КПІ ім. Ігоря Сікорського
як підручник для здобувачів ступеня бакалавра
за спеціальністю Е6 «Прикладна фізика та наноматеріали»

Електронне мережеве навчальне видання

Київ
КПІ ім. Ігоря Сікорського
2026

УДК 744:004.925.8(075.8)

K63

Автори: *Геннадій Вірченко*, д-р техн. наук, проф.
Ганна Шепель
Ольга Лебедева
Петро Яблонський, канд. техн. наук, доц.
Тетяна Надкернична

Рецензенти: *Світлана Ботвіновська*, д-р техн. наук, проф.,
завідувач кафедри нарисної геометрії та інженерної графіки
Київського національного університету будівництва і архітектури
Ярослав Корнага, д-р техн. наук, проф.,
декан факультету інформатики та програмної інженерії
Національного технічного університету України
«Київський політехнічний інститут імені Ігоря Сікорського»
Віктор Несвідомін, д-р техн. наук, проф.,
професор кафедри нарисної геометрії, комп'ютерної графіки
та дизайну Національного університету біоресурсів
і природокористування України

Відповідальний
редактор *Ольга Голова*, канд. техн. наук, доц.

*Гриф надано Вченою радою КПІ ім. Ігоря Сікорського
(протокол № 2 від 09.02.2026 р.)*

K63 **Комп'ютерна графіка в AutoCAD: автоматизація засобами Visual Basic for Applications [Електрон. ресурс] : підруч. для здобувачів ступеня бакалавра за спец. Е6 «Прикладна фізика та наноматеріали» / Г. Вірченко, Г. Шепель, О. Лебедева, П. Яблонський, Т. Надкернична ; КПІ ім. Ігоря Сікорського. – Електрон. текст. дані (1 файл). – Київ : КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2026. – 292 с.**
ISBN 978-966-990-XXX-X (Ел.)

Викладено загальні основи сучасного комп'ютерного формоутворення, зокрема використання геометричних моделей і методів побудови різноманітних ліній, поверхонь і тіл. Відповідні графічні засоби проілюстровано на прикладі системи AutoCAD з акцентом на можливостях підвищення рівня автоматизації засобами мови програмування Visual Basic for Applications. Матеріал пов'язано з навчальними вправами, наведеними у додатках. Порівняно з іншими виданнями перевага цього підручника полягає в інтеграції основ комп'ютерного моделювання форм із практичними прийомами їх реалізації та автоматизованого виконання за допомогою програм, розроблених користувачем.

Для здобувачів ступеня бакалавра за спеціальністю Е6 «Прикладна фізика та наноматеріали».

УДК 744:004.925.8(075.8)

ISBN 978-966-990-258-0 (Ел.)

© Г. Вірченко, Г. Шепель, О. Лебедева,
П. Яблонський, Т. Надкернична, 2026
© КПІ ім. Ігоря Сікорського, 2026

TABLE OF CONTENTS

INTRODUCTION	10
1. GENERAL INFORMATION ON COMPUTER-AIDED ENGINEERING	
GRAPHICS	14
1.1. Technical Equipment.....	14
1.2. Types of Computer Images	15
1.2.1. Raster Graphics	15
1.2.2. Vector Images.....	16
1.3. Use of Color	16
1.3.1. RGB Model	17
1.3.2. CMY Method.....	17
1.4. Application Software for Computer Graphics	18
1.5. General Description of Computer Geometric Modeling.....	18
1.5.1. Fundamental Concepts	19
1.5.2. Geometric Objects	19
1.5.3. Construction Methods	20
1.6. Emerging Trends and Directions of Computer Graphics Application in Modern Computer-Aided Design Systems	21
1.7. Conclusions	22
Control Questions	23
2. COMPUTER-BASED SHAPE MODELING IN THE AUTOCAD SYSTEM....	24
2.1. Getting Started	24
2.1.1. Computer Requirements.....	24
2.1.2. AutoCAD Files.....	24
2.1.3. Starting the Program.....	24
2.2. AutoCAD Interface.....	28
2.2.1. Drawing area	29
2.2.2. Application Menu	30
2.2.3. Quick Access Toolbar	33
2.2.4. Ribbon	34
2.2.5. Command Window.....	35
2.2.6. Model Space and Layout Tabs	36
2.2.7. Status Bar.....	37
2.2.8. Context menu.....	41

2.3. Two-Dimensional Modeling	42
2.3.1. Coordinate System	42
2.3.2. Units of Measurement	45
2.3.3. Drawing Limits.....	47
2.3.4. Display Control	48
2.3.5. Object Properties	49
2.3.6. Layer Management.....	50
2.3.7. Object Selection.....	54
3.3.8. Coordinate Entry Methods	56
2.3.9. Construction of Geometric Primitives.....	56
2.3.9.1. Points	56
2.3.9.2. Line Segments	57
2.3.9.3. Circles and Arcs.....	58
2.3.9.4. Ellipses and Elliptical Arcs.....	60
2.3.9.5. Polylines	62
2.3.9.6. Rectangles.....	63
2.3.9.7. Polygons	64
3.3.9.8. Donuts.....	65
2.3.9.9. Splines	66
2.3.9.10. Regions	67
2.3.10. Editing Objects	67
2.3.10.1. Moving Objects	68
2.3.10.2. Rotating Objects	68
2.3.10.3. Mirror.....	69
2.3.10.4. Copying	70
2.3.10.5. Deleting Objects	71
2.3.10.6. Scaling	72
2.3.10.7. Grips	72
2.3.10.8. Creating Similar Objects	73
2.3.10.9. Creating Chamfers.....	74
2.3.10.10. Filletting.....	75
3.3.10.11. Trimming and Extending.....	76
3.3.10.12. Breaking Objects	77
2.3.10.13. Stretching.....	78

3.3.10.14. Arrays of Geometric Objects	78
2.3.11. Text and Annotations	80
2.3.11.1. Text Styles.....	80
2.3.11.2. Creating Text.....	81
2.3.11.3. Editing Text.....	84
2.3.11.4. Dimension Styles	84
3.3.11.5. Dimensioning.....	86
2.3.11.6. Editing Dimensions	87
2.3.12. Drawing Template Files	88
2.3.13. Examples of Constructing Part Contours	89
2.3.13.1. Support Bracket	89
2.3.13.2. Support Bracket	96
2.3.14. Parametric Constraints	99
2.3.14.1. Geometric Constraints	99
2.3.14.2. Automatic Parametrization	100
2.3.14.3. Dimensional Constraints	101
2.3.15. Projection Drawing.....	104
2.3.15.1. Tracking Method.....	104
2.3.15.2. Using the XLINE and RAY Commands.....	105
2.3.15.3. Hatching.....	106
2.3.15.4. Example of Constructing an Orthographic Drawing.....	114
2.4. Solid Modeling.....	119
2.4.1. 3D Modeling Workspace.....	119
2.4.2. Viewport Controls	119
2.4.3. ViewCube	121
2.4.4. Three-Dimensional Coordinates	122
2.4.5. Coordinate System Management	122
2.4.6. Visualization Control of Geometric Objects.....	123
2.4.7. Creating Solids (Primitives)	126
2.4.7.1. Box.....	126
2.4.7.2. Cylinder	127
2.4.7.3. Cone	128
2.4.7.4. Sphere	129
2.4.7.5. Pyramid.....	131

2.4.7.6. Wedge	132
2.4.7.7. Torus	133
2.4.8. Projection Views of a 3D Object in the Layout Space.....	133
2.4.9. 3D Primitives and Development of Their Drawings.....	135
2.4.10. Commands for Constructing Complex Objects	138
2.4.10.1. Extrude.....	138
2.4.10.2. Revolve	140
2.4.10.3. Presspull.....	142
2.4.11. Boolean Operations	143
2.4.11.1. Union	143
2.4.11.2. Subtract	144
2.4.11.3. Intersect.....	144
2.4.12. Editing 3D Objects	145
2.4.12.1. Using the Gizmo	145
2.4.12.2. 3D Mirror.....	148
2.4.12.3. Chamfer Edge	149
2.4.12.4. Fillet Edge.....	150
2.4.13. Example of Creating a Solid Model from Its Projections	151
2.5. Conclusions	154
Control Questions	155
3. AUTOMATION OF CONSTRUCTIONS BY MEANS OF VISUAL BASIC FOR APPLICATIONS.....	156
3.1. Visual Basic for Applications in AutoCAD	156
3.2. Construction of Basic Figures	162
3.2.1. Points	163
3.2.2. Line Segments	166
3.2.3. Rectangles.....	169
3.2.4. Polygons	171
3.2.5. Circles and Their Arcs.....	180
3.2.6. Ellipses and Their Arcs	183
3.2.7. Splines	187
3.3. Modification of Geometric Objects	195
3.3.1. Translation.....	196
3.3.2. Rotation	197
3.3.3. Scaling	199

3.3.4. Compositions of Coordinate Transformations	201
3.3.5. Properties and Methods of Geometric Objects in Visual Basic for Applications	204
3.4. Conclusions	204
Control Questions	205
GENERAL CONCLUSIONS	207
LIST OF REFERENCES	208
INDEX OF TERMS	235
APPENDIX A. BASIC REQUIREMENTS FOR DRAWING LAYOUT	236
APPENDIX B. INDEPENDENT WORK ASSIGNMENTS	248
B.1. Part Outline.....	248
B.2. Projection Drawing	258
B.3. Three-Dimensional Constructions.....	268
B.4. Creating a Solid Model from Its Projections.....	272

INTRODUCTION

The discipline “Computer Graphics” encompasses a wide range of human activities related to the use of computer technology for creating, modifying, and storing graphical images. Among its branches are artistic, advertising, scientific, and other types of graphics, each having its own field of application.

For technical specialists – such as designers, technologists, and operators across various sectors of industry – engineering graphics plays an essential role. Its tools constitute integral components of almost all modern Computer-Aided Design (CAD) systems. The most popular among them include AutoCAD, Inventor, SolidWorks, CATIA, and several others. These programs enable the construction of diverse computer-based geometric models and provide the necessary means for generating images used in the development of relevant objects and processes.

The purpose of this textbook is to provide bachelor’s degree students majoring in Applied Physics and Nanomaterials with the competencies required to solve professional problems using computer-aided engineering graphics; to master the methods of projection drawing and the requirements of existing standards for the development of technical documentation; to acquire skills in creating custom software add-ins within Computer-Aided Design (CAD) environments; and to develop competencies in the field of computer information technologies.

The content of this textbook includes three chapters: General Information on Computer-Aided Engineering Graphics; Computer-Based Shape Generation in the AutoCAD System; Automation of Constructions by Means of Visual Basic for Applications.

The appendices contain the requirements for the preparation of drawings and assignments for students’ independent work in automated contour creation, projection drawing, and solid modeling.

The first part provides basic information on the hardware and image types used in computer graphics, as well as on color representation methods. It presents an overview of the processes of automated shape generation, relevant geometric models (wireframe, surface, and solid), and methods for their construction, as well as the prospects of computer-aided engineering graphics within modern CAD environments.

The second chapter describes the installation of the AutoCAD system, analyzes its interface, and examines the basic commands for two-dimensional and three-dimensional modeling. It explains the creation of such geometric entities as points, lines, polylines, conic curves, splines, and others, as well as the necessary procedures for their modification. It covers the performance of projection drawings and the application of parametric constraints. The chapter includes material on three-dimensional modeling – for example, the creation of primitives (parallelepipeds, cylinders, cones, pyramids, spheres, tori, etc.), the use of extrusion and revolution methods, Boolean operations, and editing of the mentioned objects – and describes the development of design documentation.

The third chapter is devoted to the fundamentals of programming using Visual Basic for Applications for automating the geometric modeling of various figures and performing their modifications. It introduces the code editor, the basic concepts of automation, data types, programming commands, and operators used in this environment.

The base version of the CAD system selected for this textbook is AutoCAD 2023, chosen for two reasons: first, it provides sufficient functionality for presenting the necessary educational material; and second, it requires less powerful hardware compared to later versions of the software package. It should be noted that the capabilities of subsequent releases remain practically unchanged in the context of the issues considered herein.

For comparison with other relevant sources, the description of the AutoCAD system on the official Autodesk website was chosen as a reference (see sources [1–368]). This choice was determined by two main factors.

Firstly, it is important to train students to work efficiently with Internet-based information resources related to specific CAD systems. The volume of such data is significant, and the cited references represent only a small fraction of the total available information.

Secondly, as indicated in the title of this textbook, its goal is to provide a comprehensive presentation of computer-aided engineering graphics in the AutoCAD environment, combined with the application of Visual Basic for Applications tools to enhance the level of automation in design processes. Unfortunately, there are no comparable publications by other authors that address this combination of topics. The aspects emphasized above represent the main distinctions of this textbook from other educational publications on similar subjects.

The presented list of references consists of two main sections. The first, “AutoCAD 2023 Help” [1–190], includes information on tools available to end users, while the second, “AutoCAD 2023 Developer and ObjectARX Help” [191–368], provides material for developing custom software add-ins using Visual Basic for Applications.

The exposition begins with a traditional approach explaining how to start working in AutoCAD [1]. It should be noted that the subsequent material follows the sequence of topics as presented on the official Autodesk website but in a significantly condensed form, focusing only on aspects relevant to the objectives of this textbook.

For the user, the introductory information includes an overview of the program interface, procedures for creating new or opening existing models, and the construction of basic geometric objects. Further sections present information on the Quick Access Toolbar [2], the Ribbon [3], the Start Tab [4], the Status Bar [5], workspaces [6], model templates [7], measurement units [8], panning and zooming [9], the navigation bar [10], the ViewCube [11], visualization of three-dimensional objects [12–14, 105, 116–123], coordinate systems [15, 16], object snap tracking [17], object snaps [18–20, 22, 23], measurement of distances, lengths, and angles [21, 24], calculation of areas and mass properties [25], creation of geometric figures [26, 28–37, 89, 91–104, 111, 112], isometric drawings [27], parametric models [38–49], object selection [50, 51, 107, 115], object modification [52–67, 106, 108–110, 113, 114], object properties [68, 69, 71–73, 90], layers [70], annotation [74–81], dimensioning [82–87], hatching [88], model printing [124–134], and system commands [135–190].

The developer-oriented information covers defining components in a Visual Basic for Applications project [191], using the code window [192], creating new macros [193], their execution [194–195], and editing [196]; project structure [197]; accessing form-related code [198]; launching the Interactive Development Environment [199]; introductory exercises [200]; terminology [201]; AutoCAD Visual Basic for Applications Command Reference [202]; object hierarchy [203]; collections [204–207]; data [208, 209]; window management [210]; system variables [211]; geometry updating [212, 213, 238]; working with object snaps [214, 215]; area calculation [216]; user queries [217–220]; command-line access [221]; file export [222]; object containers [223]; creation of specific figures [224–

227, 240–244, 301–304, 321–326, 329–336, 345]; modification of their properties [239, 245–253, 276, 305–308]; hatching [228, 229, 254, 255, 327, 328]; selection sets [230–237]; text [256–263]; model layers [264–272, 274, 277–279, 343, 344]; color management [273, 275]; dimensioning [280–293, 337–342]; menus [294–297]; coordinates [298–300, 346]; printing [309–315]; raster images [316, 317]; general Visual Basic for Applications information [318–320]; and specific methods [347–368].

Thus, the approach presented in this textbook ensures both general understanding of computer-aided engineering graphics – including the applied technical and software tools – and detailed constructions within a specific CAD environment, exemplified by AutoCAD. At the same time, special emphasis is placed on developing students' ability to effectively utilize official documentation from the developer's website and to increase the level of design automation using Visual Basic for Applications.

1. GENERAL INFORMATION ON COMPUTER-AIDED ENGINEERING GRAPHICS

In the Introduction, it was noted that engineering graphics is widely used in the work of many technical professionals, such as designers, technologists, operators, and other specialists. In this respect, it differs significantly from other types of computer graphics, including artistic, advertising, and scientific graphics. However, all of these fields share certain common tools, such as the appropriate hardware, types of computer-generated images, color modeling techniques, and specific methods and approaches to geometric constructions. These invariant components constitute the subject of a brief overview presented in this chapter of the textbook.

1.1. Technical Equipment

The hardware tools of computer graphics include devices used to create, visualize, edit, and document graphical images – that is, to produce hard copies. There exists a wide variety of such tools, among which the most common are graphic displays, keyboards, computer mice, other pointing devices, scanners, digital cameras, video cameras, plotters, printers, and others.

Graphic displays (monitors) are divided into vector and raster types. They provide temporary images on the screen that can be modified as required.

Modern computers employ color displays that reproduce graphics using cathode-ray tubes, gas-discharge (plasma), electroluminescent, liquid crystal (LCD), and other types of panels. The main technical characteristics of monitors include screen size, pixel count (raster resolution), minimum pixel dimensions and spacing, refresh rate, and the number of reproducible colors.

Keyboards are sets of keys marked with specific symbols (such as those of a standard typewriter, cursor control, and functional keys) designed for entering both textual and graphical information into a computer.

Computer mice are devices used for dynamic control of data displayed on the screen. The main element of a mechanical mouse is a ball located inside a casing convenient for user manipulation. The movement of the casing is tracked by the corresponding movement of the cursor on the screen. The mouse buttons allow the user to select desired objects. Nowadays, optical mice, in which most

mechanical components have been replaced by optical sensors, are the most widespread.

Trackballs are essentially inverted mice, where the ball is rotated directly by the user's hand rather than by moving the casing. In laptops (portable computers), touchpads are used, which translate the movement of the fingers into corresponding cursor motion on the screen.

Scanners are devices that convert graphical images from paper or similar media into electronic form.

Existing software provides a user-friendly interface between *digital cameras* and *video cameras* and computer systems.

Previously, to produce large-format drawings, vector pen plotters of flatbed or drum type were widely used. Currently, *printers*, including large-format ones, are employed to produce hard copies of images, with *inkjet* and *laser* printers being the most common. The key characteristics of these devices include print area size, capability for both monochrome and color printing, print quality, and printing speed.

At present, *multifunctional devices* that combine several units – such as a scanner and a printer – have become very popular.

1.2. Types of Computer Images

A person obtains the greatest amount of information about the surrounding world through vision. Other senses – such as hearing, touch, and others – merely complement the overall perception of objective reality. Therefore, visual perception of information is generally the most effective for humans. This is why graphical representation has become an essential component of virtually all modern Computer-Aided Design (CAD) systems, and an appropriate graphical interface has become a software standard, extending even to operating systems, such as Microsoft Windows.

Two main types of computer-generated images can be distinguished according to their method of formation: *raster* and *vector* images.

1.2.1. Raster Graphics

On a monitor screen, an image is represented by a set of discrete points – *pixels* – of circular, square, or other shapes. The total number of pixels forms a rectangular grid of size $(n+1) \times (m+1)$, where $0 \dots n$ and $0 \dots m$ are, respectively,

the row and column indices of the current pixel; each pixel may have its own *color*. Such a matrix covering the entire display is called a *raster*; hence the name of these images, whose principal drawback is the large memory requirements needed to achieve high quality.

Raster graphic information is produced by scanning, by digital cameras and camcorders, by specialized software, and in the process of printing on different types of printers. For its processing, appropriate graphics editors are used, allowing results to be saved in different file formats (GIF, JPEG, PNG, TIFF, etc.), each with particular advantages and limitations. A detailed discussion of these formats lies beyond the scope of this textbook.

1.2.2. Vector Images

Unlike raster graphics, which represent matrices of points, vector images are based on *lines* that possess specific *attributes* or properties, such as *thickness*, *style* (solid, dashed, etc.), *color*, and others. Additionally, for closed contours, it is possible to fill the enclosed areas with desired shades, textures, or similar features.

Thus, in vector graphics, an image is a collection of geometric primitives (points and lines) described by corresponding *analytical expressions* in vector form. As a result, the required computer memory is significantly reduced and does not depend on the absolute dimensions of the modeled figures.

The advantages of vector graphics also include:

- Suitability for scaling without loss of image quality, which allows for the refinement of small details.
- Convenient and efficient editing due to the manipulation of individual primitives.

These features are particularly important for modern Computer-Aided Design (CAD) systems.

The main disadvantage of the vector approach lies in its limited capabilities for producing artistic or photorealistic images.

1.3. Use of Color

In nature, colors are formed either by light-emitting objects or as a result of the absorption and reflection of light by the objects surrounding us. The different wavelengths of light are perceived by humans as specific colors.

1.3.1. RGB Model

This method is based on the ability of the human eye to recognize radiation of three primary colors – *red* (R), *green* (G), and *blue* (B). The specific coloration of objects arises from the brain's analysis of the relative excitation levels of the corresponding visual receptors. The RGB model is widely used for generating the required color palette in technical devices that create graphical images by means of emitted light (for example, in computer displays).

One of the methods for implementing this approach is as follows: each pixel on the screen consists of three subareas – red, green, and blue – and the perceived color depends on the balance of their intensity levels. If all three components are equal to zero, the result is black; when all reach their maximum values, the result is white.

By mixing red and green, we obtain yellow, and by mixing red and blue – magenta (violet). Thus, a certain color C can be represented by the radius vector $C(R, G, B)$ in a rectangular coordinate system $Orgb$, where O is the origin, and r , g , and b are the axes corresponding to the intensities of red, green, and blue components, respectively.

In this representation, the direction of the vector C defines the chosen color, while its length determines the color's intensity. If color encoding is performed using three bytes per pixel (that is, eight bits each for R, G, and B), the brightness of every component takes values from 0 to 255 ($2^8=256$).

This provides the ability to display on the monitor screen more than 16 million colors ($256 \times 256 \times 256$).

Since the resulting brightness increases with the addition (mixing) of the component intensities, the RGB model is referred to as an *additive* color model.

1.3.2. CMY Method

This method is based on the absorption of portions of white sunlight by non-luminous objects that do not emit light themselves. Since white light contains all colors, the resulting color of an object depends on which wavelengths are reflected and which are absorbed.

In the CMY model (Cyan, Magenta, Yellow), the main colors are obtained by subtracting the primary additive colors of the RGB model from white: Cyan = White – Red, Magenta = White – Green, Yellow = White – Blue.

Hence, the CMY method utilizes the property of subtracting certain colors from white light, which is why it is referred to as a *subtractive* color model. It provides excellent color perception in reflected light, making it widely used in printing technologies. In practice, obtaining black by mixing other colors is costly and inefficient; therefore, black ink (K – black) is added as a fourth component, resulting in the CMYK model.

Color reproduction is not limited to the RGB and CMY methods; there are also other approaches, which can be studied in specialized literature.

1.4. Application Software for Computer Graphics

According to the types of images discussed in Sections 1.2.1 and 1.2.2, corresponding graphic software packages can be conventionally divided into two classes: raster and vector. It is important to understand their main distinctions, advantages, and disadvantages.

An image generated by a raster graphics package typically consists of a large number (hundreds of thousands or more) of pixels, each of which can have an arbitrary color (usually chosen from more than 16 million possible shades). As a result, the quality of raster images is quite high and corresponds well to photographic representations of real-world objects. However, in addition to their large memory requirements, raster programs have the disadvantage of being complex and time-consuming to edit, often requiring users to possess certain artistic skills. Examples of raster graphics packages include Microsoft Paint, PhotoFiltre, Adobe Photoshop, and others.

In vector graphics software, the definition of geometric objects is based not only on pixel coordinates but also on mathematical equations. This allows for easy manipulation of geometric primitives (points, lines, text, etc.), changing their shape, size, position, and such attributes as color or line style. These operations are performed with high precision. Vector programs form the foundation of graphics in modern Computer-Aided Design (CAD) systems, one of which is AutoCAD, whose application is the subject of this textbook.

1.5. General Description of Computer Geometric Modeling

The representation of objects and processes by graphical means, that is, *geometric modeling*, usually occupies a central position in modern Computer-Aided Design (CAD) systems. This is because the parameters of shape,

dimensions, and position are extensively used in the design, manufacture, and operation of various types of technical products.

For example, geometric data are required for finite element strength analysis, preparation of engineering drawings, and manufacturing on computer numerical control (CNC) machines, among many other applications. In what follows, the main approaches, terminology, methods, and techniques of modern computer geometric modeling are considered.

1.5.1. Fundamental Concepts

In almost all modern Computer-Aided Design (CAD) systems, it is possible to work with *wireframe*, *surface*, and *solid geometries*. Historically, the first to appear were *wireframe models*, which consist of points and lines. The main disadvantage of these models lies in the ambiguity of the constructed objects between the elements of the wireframe. As a result, calculations of such characteristics as area, volume, mass, and center of gravity are often inaccurate. However, their advantage is the simplicity of the underlying mathematical framework and the low computational requirements (processor performance, computer memory, etc.).

Surface models are more advanced, since, in addition to points and lines, they also include various surfaces. The main shortcoming of these models is their inability to effectively simulate real physical bodies and their numerous properties.

Solid models are currently the most advanced, because, on the one hand, they partially generalize the two approaches discussed above, and on the other, they allow sufficiently realistic representation of objects and phenomena of the surrounding world. These achievements are the result of modern advances in applied geometry and computer technology.

At present, the most popular area of application of computer solid modeling is the design, manufacturing, and operation processes in many branches of industry.

1.5.2. Geometric Objects

In Computer-Aided Design (CAD) systems that operate with solid models, the basic geometric figures are conventionally divided into four groups:

- *Points*, which are the smallest elements, have no dimensions, and are characterized only by their location in space.

- *Lines*, which are considered as the trajectories of moving points. The defining property of these components is their length. Lines lying in the same plane are called planar, otherwise they are spatial.

- *Surfaces*, which are generated by the movement of certain lines, often changing shape during motion. The main characteristic of surfaces is their area/

- *Solids*, whose principal distinction from the geometric figures analyzed above is the presence of volume.

1.5.3. Construction Methods

This section provides a brief overview of the most common methods used in modern Computer-Aided Design (CAD) systems to define the geometric objects described above.

Points can be defined by their coordinates (Cartesian, spherical, cylindrical, etc.), by their location parameters on lines or surfaces, or by characteristic positions on geometric objects (for example, the center of a figure or the intersection of certain primitives).

Lines are created using the corresponding equations, through interpolation, approximation, or extrapolation of point sets; by moving points along trajectories; by calculating surface intersections; by joining curves, and by other similar techniques.

Surfaces are defined through their analytical expressions, by interpolation, approximation, or extrapolation of source points and lines, or by kinematic methods – that is, by changing the position, shape, and size of curves in space.

Solids are modeled using predefined primitives (prisms, cylinders, cones, spheres, tori, etc.), by applying set-theoretic operations (union, subtract, intersect), by generating thickness for selected surfaces, by extruding or moving them in space, and through other methods.

A universal technique for constructing elements of all four groups is the modification (change in shape, size, or position) of already existing geometric objects. It should be noted that using this approach, simpler components can be derived from higher-level entities – for example, surfaces and curves from a solid, or points from a line.

At present, the most common practice in CAD systems is the use of geometric objects in parametric form. This applies not only to individual

primitives such as lines, surfaces, and solids, but also to more complex structures, for instance, machine parts. In this case, the engineer defines the shape and dimensions of the created objects using parameters and constraints. Purposeful variation of these parameters allows the designer to automatically generate multiple versions of the studied objects and to perform their optimization. This approach is known as *parametric modeling*.

1.6. Emerging Trends and Directions of Computer Graphics Application in Modern Computer-Aided Design Systems

Several decades ago, *engineering drawings* were practically the only effective means of conveying geometric information about technical products. Spatial data were represented through the projections of objects onto specific planes. Thus, the designer, having formed a three-dimensional concept of the object in their mind, would create its projections, sections, and cross-sections on paper. Only in this way could the engineer's intent be understood by others. This was a rather labor-intensive process – from the mental spatial model to its two-dimensional representation. Accurate interpretation of drawings, especially of complex technical objects, also required significant effort and professional training.

The emergence of the first two-dimensional (2D) Computer-Aided Design (CAD) programs, the most well-known of which was AutoCAD, allowed designers to abandon the drawing board and to develop technical documentation efficiently with the help of a computer. However, the need to depict the three-dimensional world in a flat form still remained. Modern three-dimensional (3D) CAD systems – such as SolidWorks, CATIA, and others – now enable users to work with solid geometric models that realistically reproduce the actual technical objects under development.

In addition to CAD packages, Computer-Aided Design systems also include CAE (Computer-Aided Engineering) software for engineering analysis, such as ANSYS, NASTRAN, and others. These products are considered global standards for strength analysis, thermophysical studies, electromagnetism, and the dynamics of fluids and gases. Like CAD programs, CAE systems also rely on geometric models, which can be created using primitives such as points, lines, surfaces, and solids, or imported from other environments. There is a consistent modern trend toward the integration of CAD and CAE systems.

Due to their high-quality visualization capabilities, Computer-Aided Design systems allow designers to create realistic virtual prototypes of industrial products instead of much more expensive physical prototypes.

A natural development is the integration of CAD/CAE packages with CAM (Computer-Aided Manufacturing) systems for automating the production of technical objects. In this case, computer-based geometric models serve as the foundation for manufacturing processes, enabling the use of numerically controlled machines, robots, and other automated tools.

The further development of CAD/CAM/CAE tools is closely connected with PDM (Product Data Management) and PLM (Product Life-Cycle Management) technologies, which ensure optimal automated management of the entire product life cycle – including design, production, and operation.

The associative relationship between three-dimensional models and drawings makes it possible to generate a significant portion of design, technological, and other documentation automatically.

A characteristic feature of modern Computer-Aided Design systems is their user-friendly graphical interface, comprehensive documentation, and openness, meaning that users are provided with powerful software tools for implementing their own data-processing algorithms. This includes the Visual Basic for Applications environment, whose practical use is demonstrated in this textbook.

Thus, computer-based geometric models occupy a special place in modern Computer-Aided Design systems, as they form the basis for solving a wide range of problems associated with the development, manufacturing, and operation of various industrial products.

1.7. Conclusions

This chapter of the textbook provides general information on computer-aided engineering graphics, including the relevant hardware tools, types of images, color reproduction models, and application software used in this field. Basic information is presented concerning the principal methods for constructing various geometric figures, as well as promising directions in the development of computer graphics within modern Computer-Aided Design (CAD) systems. The issues analyzed in this chapter make it possible to perceive the subsequent learning material comprehensively, that is, from the standpoint of a system-based approach.

Control Questions

1. Describe the purpose of computer-aided engineering graphics and provide several examples of its effective practical applications.
2. List the main technical tools of modern computer graphics and explain their basic operational functions.
3. Explain the principal difference between raster and vector graphics, as well as their advantages and disadvantages.
4. Discuss the mathematical models of color reproduction and the areas of their rational application.
5. Give a brief overview of several computer graphics software packages.
6. Define the fundamental theoretical principles of modern automated geometric modeling.
7. Classify geometric figures by dimensionality and describe the main construction methods used in current Computer-Aided Design systems.
8. Characterize wireframe, surface, and solid models, outlining their advantages and disadvantages.
9. Substantiate the special role of geometry in the integration of CAD/CAM/CAE systems.
10. Explain, using specific examples, the essence of parametric computer models.
11. Prove the importance of providing CAD users with efficient software tools for applying their own algorithms in practice.
12. Identify some prospects for further development of computer-aided engineering graphics and justify your statements.

2. COMPUTER-BASED SHAPE MODELING IN THE AUTOCAD SYSTEM

2.1. Getting Started

As noted earlier, this textbook considers the AutoCAD 2023 version, which is installed from the official website of the software developer – Autodesk. Students and educators can obtain free educational access to Autodesk products and services for a certain period, with the option to renew it. Instructors can add students to their license if the students have their own Autodesk accounts registered on the specified website.

2.1.1. Computer Requirements

Prior to installing AutoCAD 2023, users should consult the official Autodesk website for up-to-date hardware and software requirements. The listed specifications define the necessary operating system, CPU type, memory size, display resolution, graphics adapter performance, and input devices to ensure optimal program operation.

2.1.2. AutoCAD Files

In AutoCAD, a model is stored in a DWG file, which not only contains geometric data but also a set of parameters that define measurement units, object properties, and other settings. By default, AutoCAD uses the .dwg (drawing) file format. File names may include letters (uppercase or lowercase), numbers, and specific symbols such as –, _, and !. Template drawings are saved with the .dwt extension, while specialized exchange formats are available for compatibility with other CAD systems.

2.1.3. Starting the Program

In Windows, AutoCAD 2023 can be launched from the desktop shortcut or the Start menu.

Upon startup, the Start window (Fig. 2.1) appears, providing quick access to recent files, templates, and learning resources.

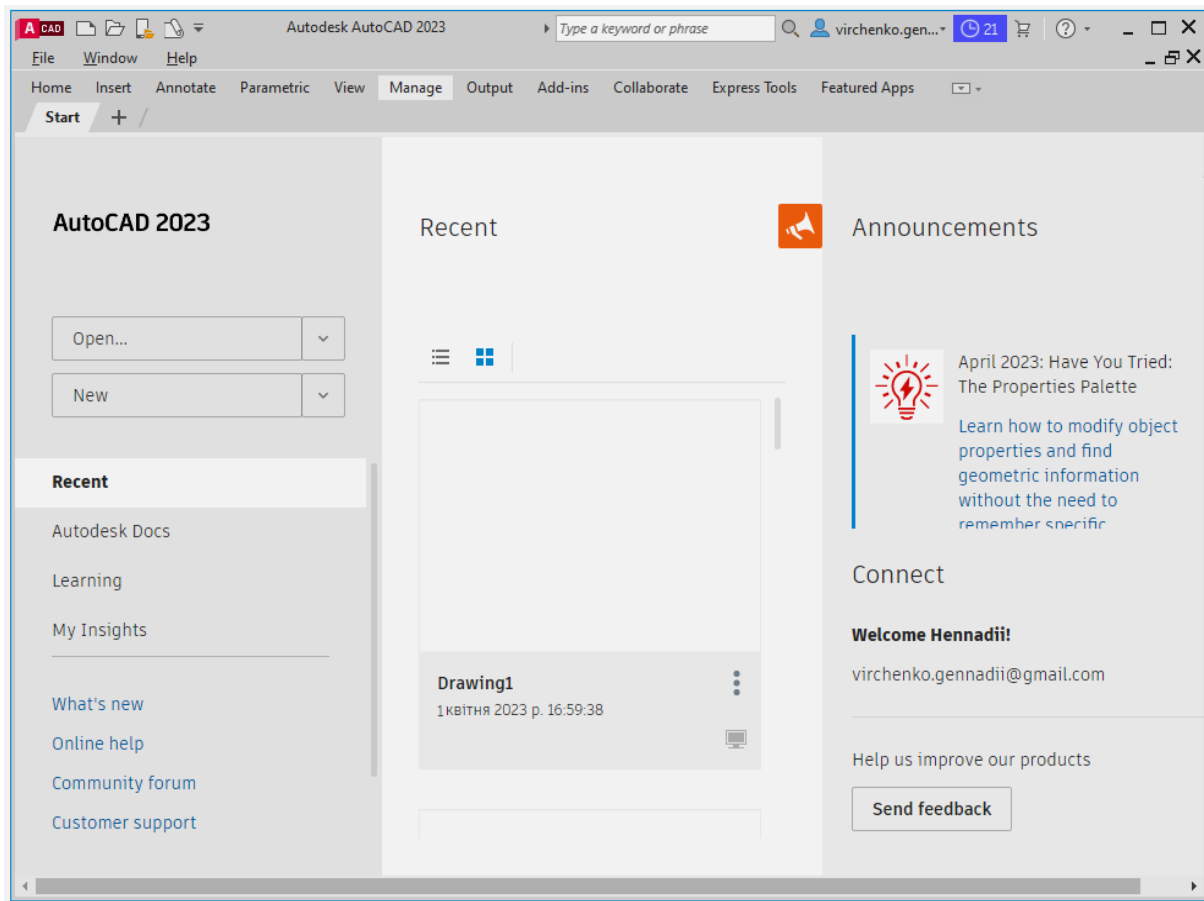


Fig. 2.1. AutoCAD 2023 Start window

The Start tab (Fig. 2.2) displays the Recent area with the most recently opened drawings.

Using the Open menu (Fig. 2.3), users can open existing projects from the file system, load a sheet set, or explore sample drawings provided by AutoCAD.

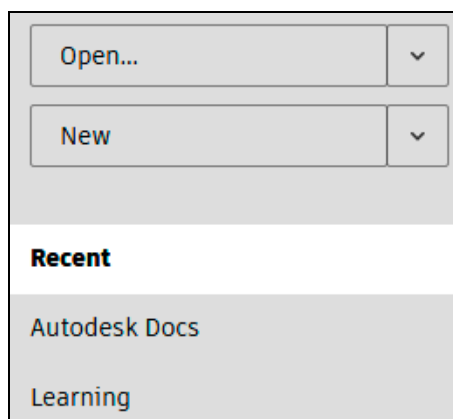


Fig. 2.2. Start tab

Using the Open menu (Fig. 2.3), users can open existing projects from the file system, load a sheet set, or explore sample drawings provided by AutoCAD.

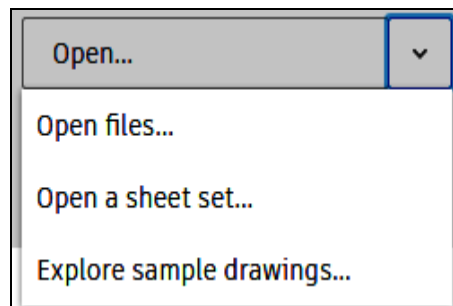


Fig. 2.3. Open command options

The New menu (Fig. 2.4) is used to create a new drawing based on a selected template. For first-time users, the acadiso.dwt template is recommended – it complies with ISO (International Organization for Standardization) standards, applies millimeters as default units, and uses the A3 sheet format. Additional templates can be browsed or downloaded online, and users may also create sheet sets to manage multiple drawings within a project.

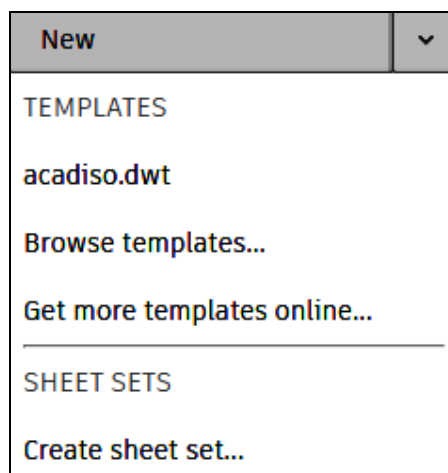


Fig. 2.4. The New menu for creating a new drawing

The Learning section (Fig. 2.5) provides users with access to a variety of educational resources aimed at improving their proficiency in AutoCAD. It includes short tips offering practical advice and examples of commonly used features, instructional videos demonstrating step-by-step procedures for performing key operations, and interactive tutorials that guide users through the interface and fundamental drawing techniques.

This section serves as a convenient starting point for both beginners and experienced users seeking to enhance their skills or explore new tools introduced in the latest version of the program.

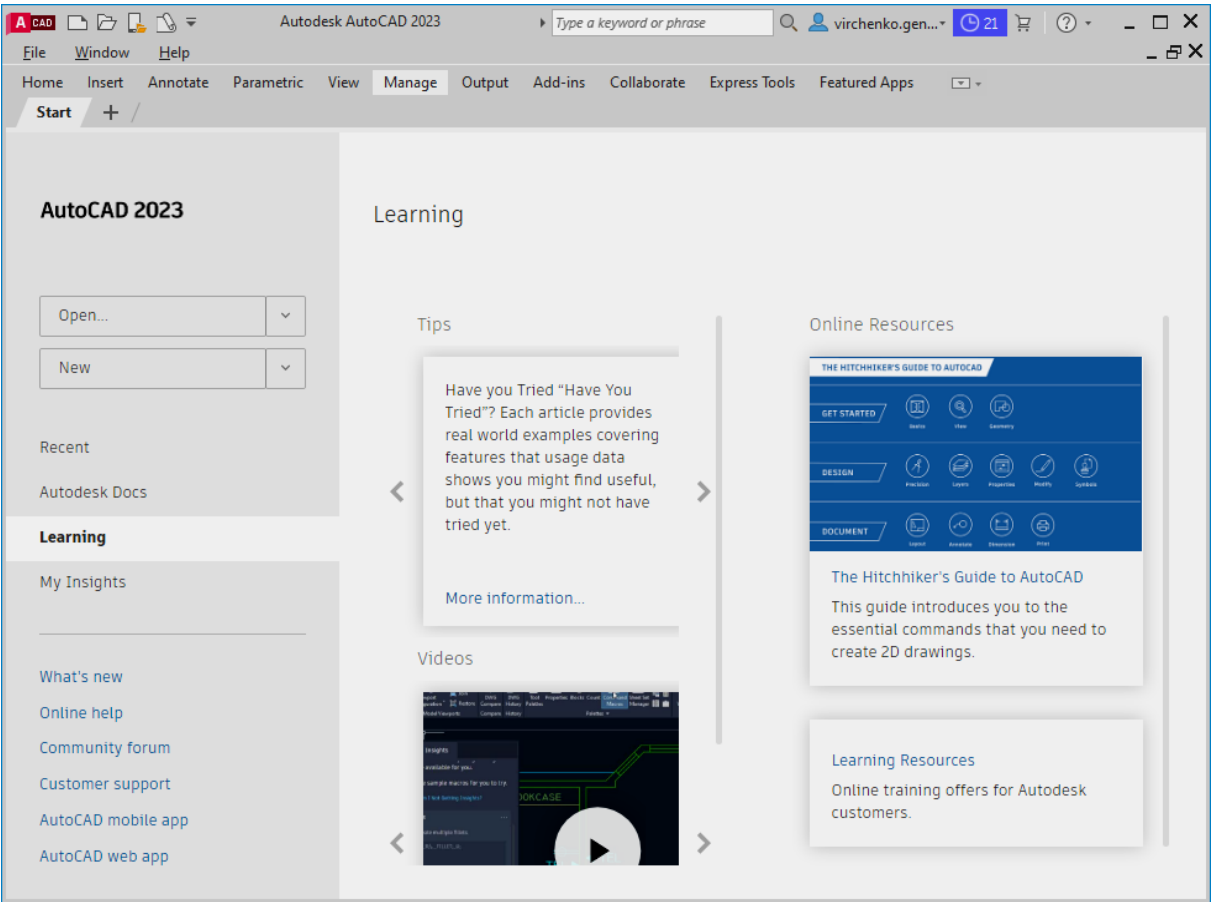


Fig. 2.5. Learning window with educational resources in AutoCAD 2023

Links in the information block (Fig. 2.6) lead to official Autodesk online resources: What’s New, Online Help, Community Forum, Customer Support, and AutoCAD’s mobile and web applications.

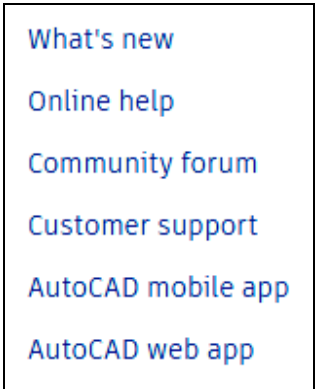


Fig. 2.6. Information and resource links block

2.2. AutoCAD Interface

Once a new drawing is created based on the acadiso.dwt template, AutoCAD opens the main graphic workspace of the editor.

The default layout and interface elements of this workspace are illustrated in Fig. 2.7.

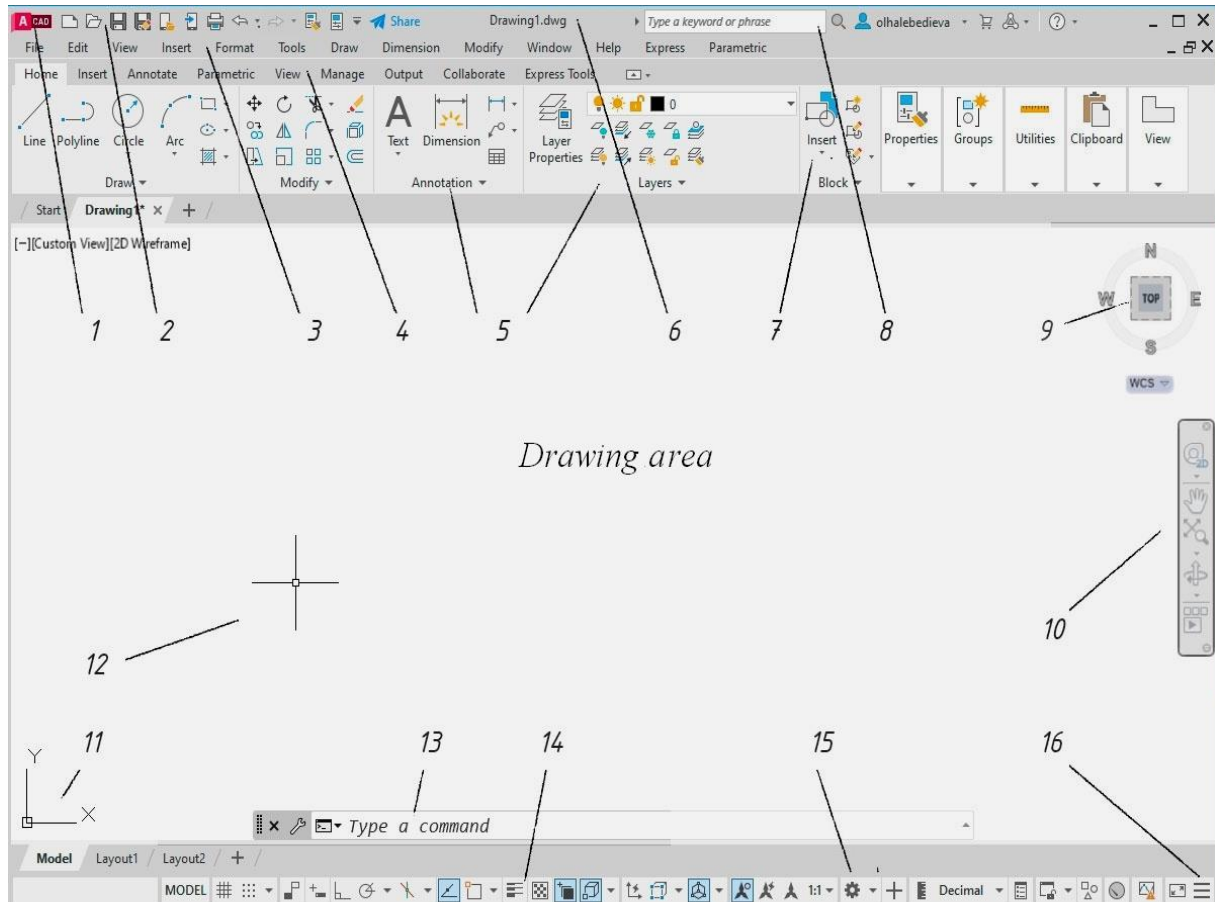


Fig. 2.7. Main interface elements of the AutoCAD 2023 graphic editor window:

- 1 – Application Menu; 2 – Quick Access Toolbar; 3 – Drop-down Menu;
- 4 – Ribbon Tabs; 5 – Tool Panels; 6 – Window Title Bar; 7 – Command Icons;
- 8 – Quick Search; 9 – ViewCube; 10 – Navigation Bar; 11 – Coordinate System;
- 12 – Crosshair Cursor; 13 – Command Line; 14 – Status Bar;
- 15 – Workspace Switching Menu; 16 – Status Bar Customization

The window displays the interface elements provided by the Drafting & Annotation workspace, which is set as the default environment. AutoCAD provides several predefined workspaces:

- Drafting & Annotation – for two-dimensional modeling;
- 3D Basics – for basic three-dimensional modeling;
- 3D Modeling – for advanced three-dimensional design.

2.2.1. Drawing area

The central region of the interface illustrated in Fig. 2.7 is known as the drawing area, where all created and edited objects are displayed.

The mouse cursor in this workspace appears as a crosshair with a small square at the center, allowing precise positioning of elements during drawing and editing. In the lower-left corner of the area, the User Coordinate System (UCS) icon indicates the current orientation of the coordinate axes.

Along the right margin of the workspace are located the ViewCube, which enables interactive 3D view manipulation, and the Navigation Bar, providing quick access to panning, zooming, and orbiting tools (see Fig. 2.8).

In the ViewCube interface, the label WCS (World Coordinate System) appears to indicate that the model is oriented according to the global coordinate system, which serves as the fixed reference for all objects within the drawing. The WCS establishes the fundamental X, Y, and Z axes and the origin point (0, 0, 0) that remain constant throughout the project.

In addition to this global system, users can create User Coordinate Systems (UCS) that allow flexible manipulation of the drawing plane – useful for constructing geometry on inclined or rotated surfaces. When a UCS is activated, the ViewCube dynamically updates its orientation, while the WCS indicator continues to provide a visual reference to the original global coordinates..

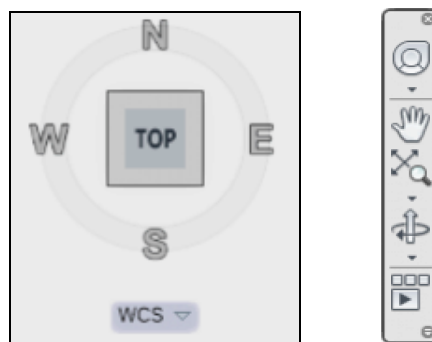


Fig. 2.8. ViewCube and Navigation Bar

In Fig. 2.8, the Navigation Bar contains several tools for managing the view of the drawing and navigating within the 3D workspace:



– Full Navigation Wheel – a multifunctional interface that integrates several 3D navigation tools.



– Pan – shifts the current view parallel to the drawing plane.



– Zoom Extents– adjusts the zoom level so that all visible objects fit within the window.



– Orbit – enables free rotation of the model in 3D space.



– Show Motion – controls camera animation and playback.

The Full Navigation Wheel groups several 3D navigation tools and is primarily intended for advanced users. The Show Motion tool allows control over stored camera views (shots) and enables the creation of animations that provide a comprehensive view of the existing geometric model.

2.2.2. Application Menu

Previously, several methods for creating a new drawing  upon starting AutoCAD were described. During an active session, users can perform various file operations – such as creating, opening, saving, exporting, publishing, or printing drawings – through the Application Menu (see Fig. 2.9). This menu is accessed by clicking the Application button located in the upper-left corner of the AutoCAD window and provides a convenient central hub for managing project files.

The tools available in the Application Menu allow users to perform essential file operations such as creating a new drawing (New), opening an existing one (Open), saving the current project under its existing name (Save) or a new name (Save As), importing data from other formats (Import), exporting drawings to different file types (Export), publishing drawing sets (Publish), printing documents (Print), and closing the current file (Close).

In addition to these commands, the menu contains a search field for finding specific commands or documentation topics, as well as buttons to open the Options dialog box and to exit AutoCAD.

The Options dialog box (Fig. 2.10) provides a comprehensive environment for customizing and managing the program's behavior. It consists of ten tabs – Files, Display, Open and Save, Plot and Publish, System, User Preferences, Drafting, 3D Modeling, Selection, and Profiles. Through these tabs, users can precisely configure how AutoCAD appears and performs.

The Files tab defines search paths for template files, support folders, and automatic save locations, ensuring correct file management within projects.

The Display tab controls the visual appearance of the workspace, allowing users to select between light and dark color themes, specify toolbar and ribbon display options, and adjust the smoothness of curves and rendered objects.

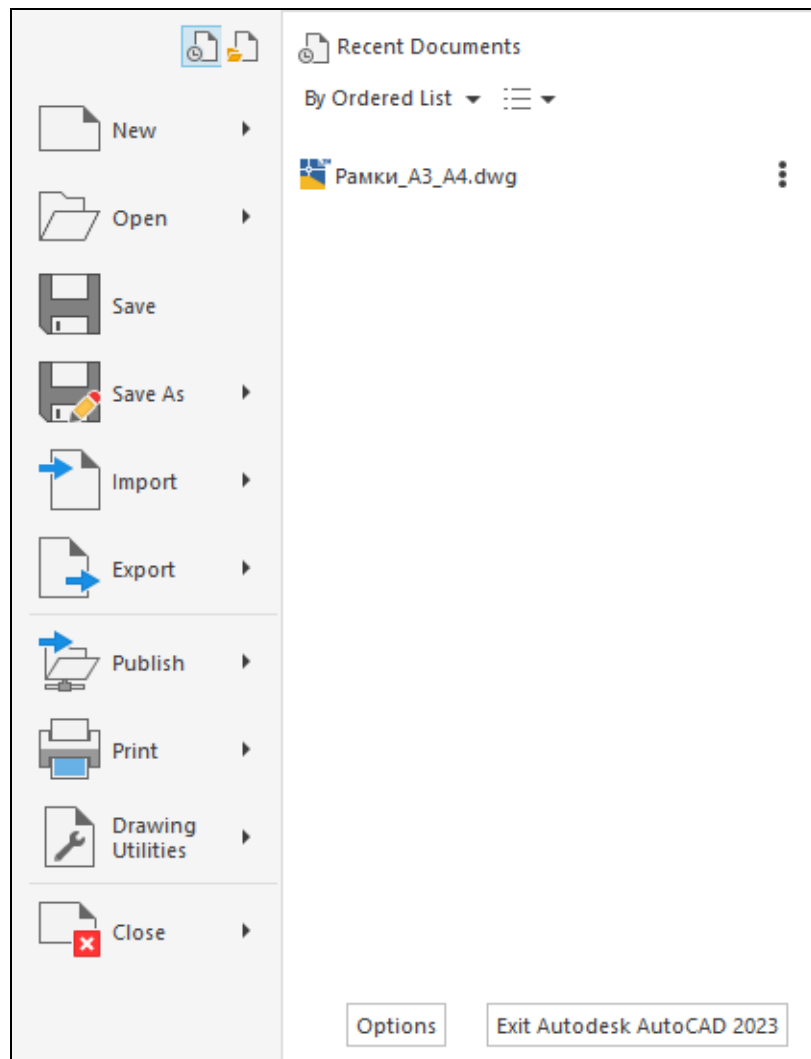


Fig. 2.9. Application Menu for file management operations

The Open and Save tab manages file handling parameters, such as the default drawing format, automatic save interval, and number of backup files retained by the system.

The Plot and Publish tab contains settings for plotter configurations, page setups, and publishing options, making it possible to manage multi-sheet output or generate PDF documentation.

Within the System tab, users can enable or disable hardware acceleration, control display performance, and specify general system behaviors to improve program responsiveness.

The User Preferences tab allows customization of the crosshair size, grip appearance, command input behavior, and other interactive interface elements.

The Drafting tab defines drawing aids such as object snaps, tracking modes, and polar alignment, which enhance precision when creating or editing geometry.

The 3D Modeling tab provides access to visual styles, rendering quality, and lighting effects, which influence how three-dimensional objects are displayed and manipulated.

The Selection tab regulates how objects are highlighted, selected, and previewed before modification, improving control during editing.

Finally, the Profiles tab allows users to create, save, and switch between different workspace configurations, making it easy to maintain personalized AutoCAD environments for various design tasks.

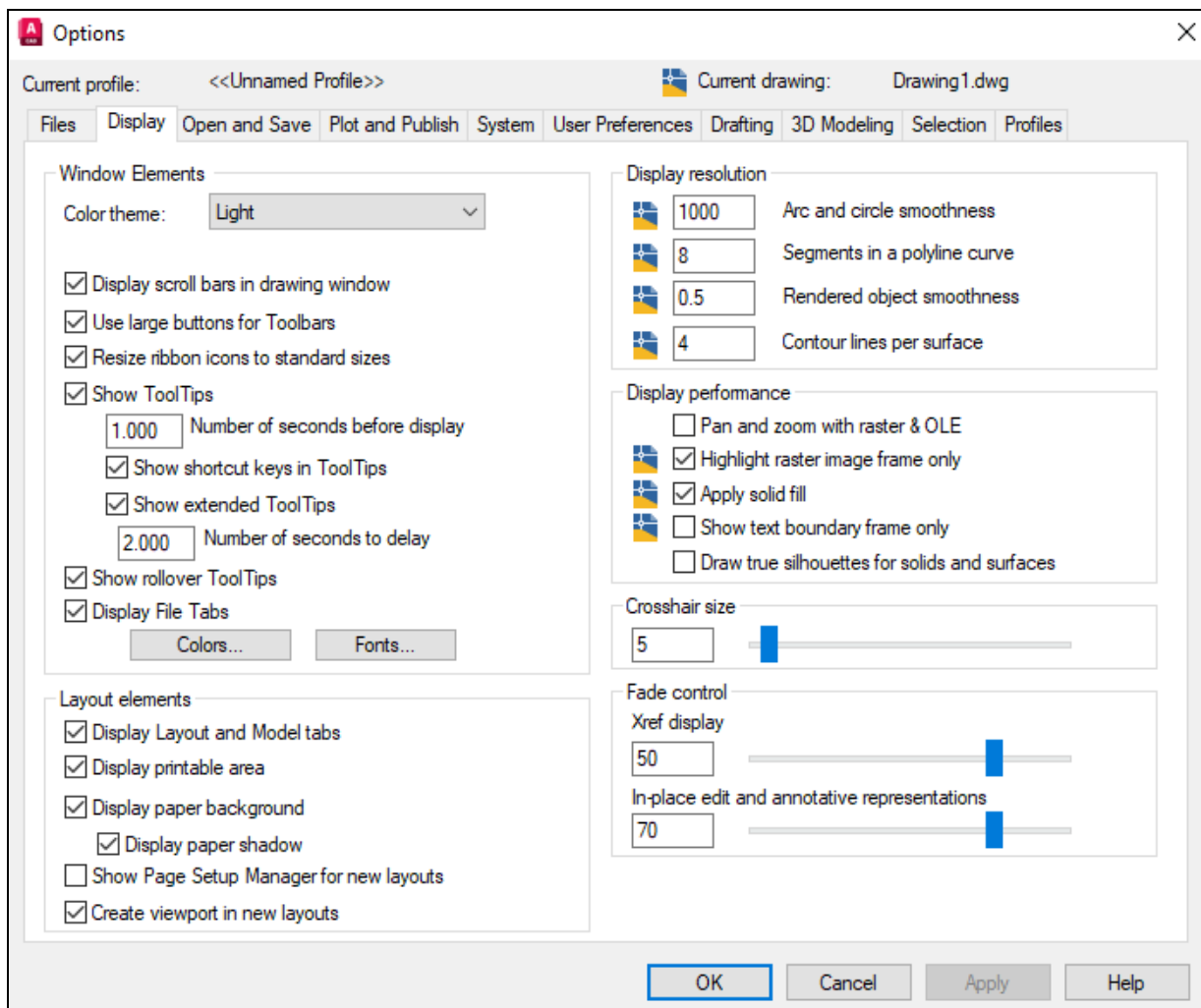


Fig. 2.10. Options dialog box

2.2.3. Quick Access Toolbar

At the top of the AutoCAD window is the Quick Access Toolbar (see Fig. 2.11), which contains general file management commands:

 – New,  – Save,  – Save As,  – Open,
as well as editing commands such as:
 – Undo,  – Redo,

which reverse or repeat the most recent actions.

This toolbar can be customized according to the user's preferences by adding or removing commands as needed.

The Customize Quick Access Toolbar button allows the user to display or hide specific tool menus available in AutoCAD (see Fig. 2.11).

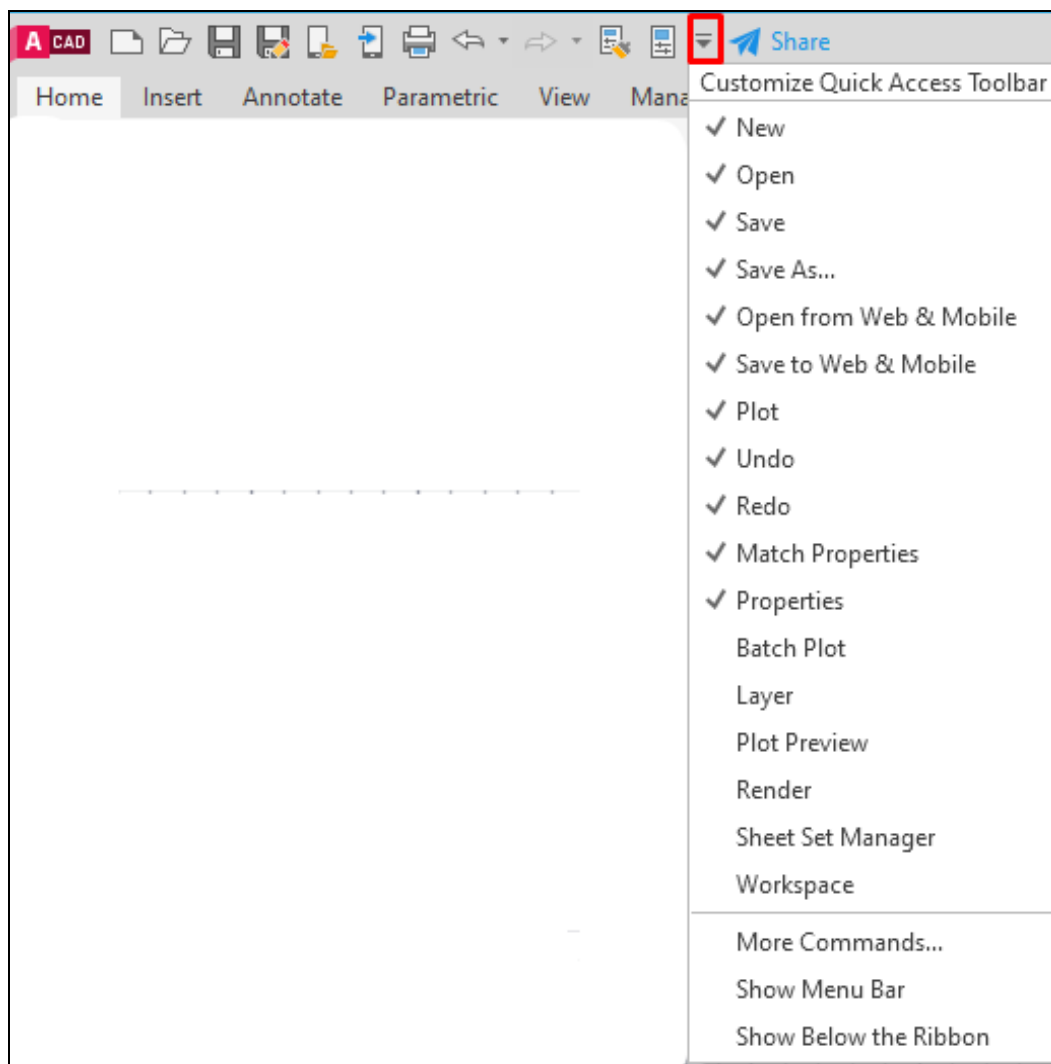


Fig. 2.11. Quick Access Toolbar with the customization menu open

The drop-down menu is displayed below the Quick Access Toolbar (see Fig. 2.12), i.e., above the Ribbon tabs.

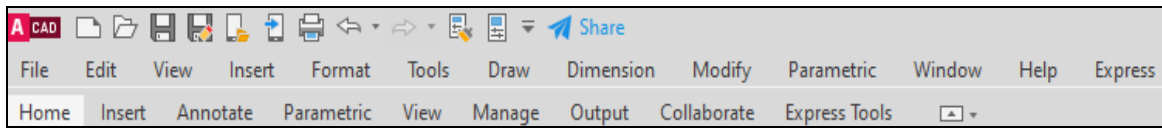


Fig. 2.12. Middle row of the drop-down menu

This menu is often referred to as the traditional menu, since its structure corresponds to that of earlier AutoCAD versions. It provides access to a more comprehensive list of commands and parameters than the Ribbon interface. The MENUBAR system variable is used to control its visibility: setting it to (0) hides the menu bar, while setting it to (1) displays it.

2.2.4. Ribbon

Directly above the drawing area is the Ribbon (see Fig. 2.13), which contains tools and commands related to the current workspace. The Ribbon eliminates the need to display multiple floating toolbars, thereby maximizing the available drawing space. It provides a structured and visually organized interface that groups commands into tabs and panels, making it easier to access frequently used tools. The Ribbon is displayed automatically when creating a new drawing or opening an existing one in any of the following workspaces: Drafting & Annotation, 3D Basics, or 3D Modeling.

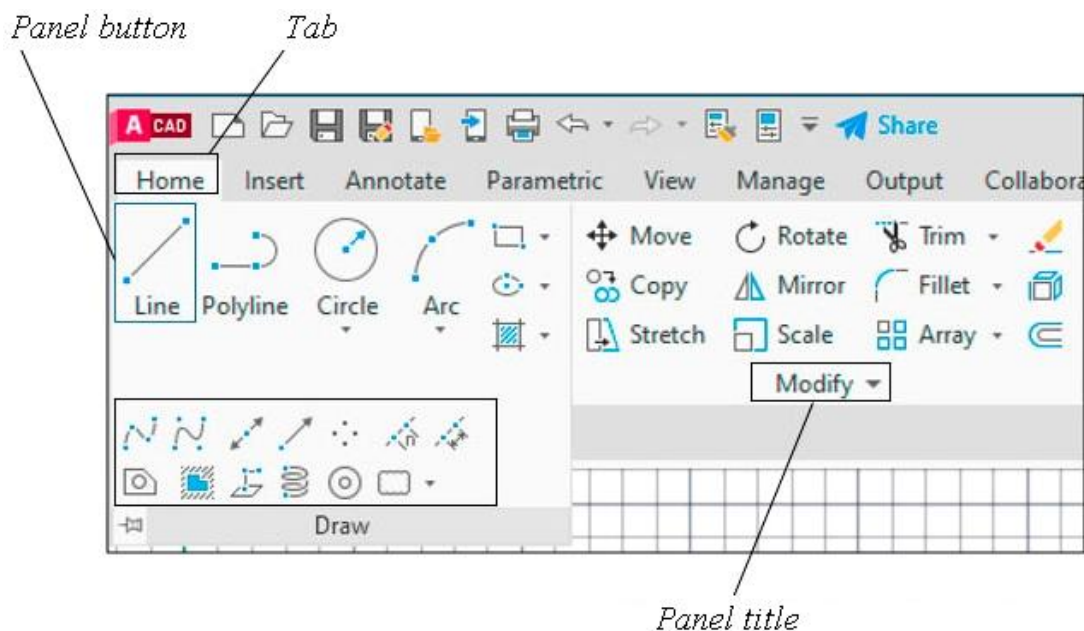


Fig. 2.13. Ribbon elements

The Ribbon consists of panels organized into tabs such as Home, Insert, Annotate, and others. Each panel contains related tools and control elements. The name of each panel is displayed at the bottom, and a small arrow (triangle) in the panel title indicates that it can be expanded to show additional tools (see Fig. 2.13). If the mouse cursor is moved away from the expanded panel, it automatically collapses back to its minimized state. The pin  icon allows users to keep the panel expanded for continuous access to its tools.

The appearance of the Ribbon can be adjusted by minimizing it to tabs, panel titles, or panel buttons, as shown in Fig. 2.14. The control button for this feature is located at the far-right end of the Ribbon tab bar.

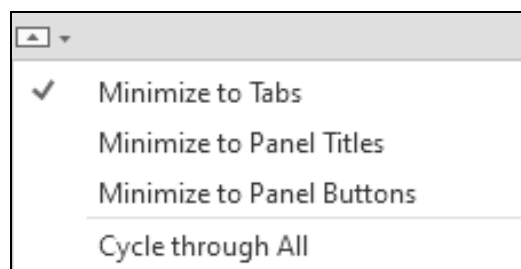


Fig. 2.14. Ribbon display control options

The Ribbon can also be hidden or restored by pressing the keyboard shortcut Ctrl + 0.

2.2.5. Command Window

At the bottom of the AutoCAD window is the Command Line area (Fig. 2.15), which is used to enter commands and specify parameters during interaction with the system. The command line can be toggled on or off using the Ctrl + 9 keyboard shortcut. Its length can be adjusted by dragging the right edge, and the entire window can be moved by dragging the grip icon located on its left side .

By dragging the upper edge of the command window upward, the Command History expands to display a list of previously executed actions (see Fig. 2.16). The same result can be achieved by pressing F2 or clicking the arrow icon at the right side of the command line (refer to Fig. 2.15).



Fig. 2.15. Command Line window

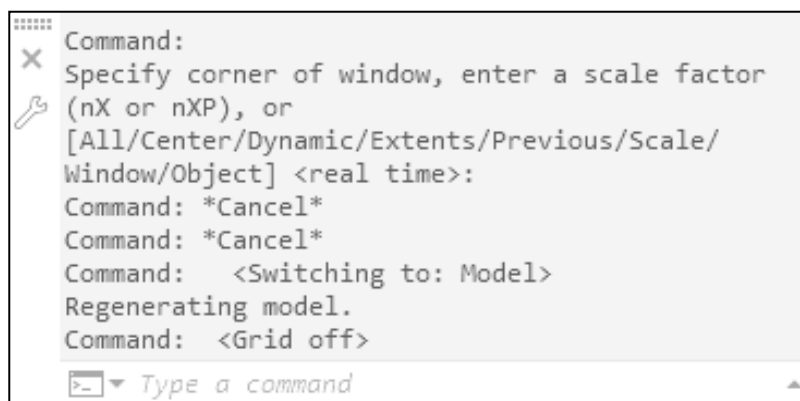


Fig. 2.16. Text window displaying the command history

Commands and their options can be entered directly in the Command Line, without using the Ribbon, toolbars, or menus. It is sufficient to start typing the name of a command and select it from the suggested list that appears automatically. When working through the Command Line, default values are displayed in angle brackets. To accept a default value, press Enter.

To choose an option from a list, move the cursor over the desired item and click the left mouse button. Pressing Enter confirms the selected parameter or completes the current command. To repeat the last executed command, simply press Enter at the prompt Type a command.

2.2.6. Model Space and Layout Tabs

AutoCAD provides two primary working environments: Model Space and Layout Space. Model Space is an unlimited 3D workspace where the geometric modeling of objects is performed. The Model tab is active by default.

Layout Space, on the other hand, is a 2D environment intended for creating drawing sheets. Within this space, users prepare scaled viewports of the model for printing on a specific sheet size. It is also used to add title blocks, specifications, tables, and other required documentation elements.

The Model and Layout tabs bar (see Fig. 2.17) is located below the drawing area, on the left side. A new drawing contains three tabs by default – Model, Layout1, and Layout2 – along with a plus (+) button that allows users to create additional layout sheets.

All geometric construction and modeling are carried out in the Model environment.



Fig. 2.17. Model and Layout tabs

2.2.7. Status Bar

Below the Command Line is located the Application Status Bar (see Fig. 2.18).



Fig. 2.18. Status Bar (default component layout)

The Status Bar contains a set of buttons that enable or disable specific drawing modes, activate auxiliary tools, and display relevant information about the current workspace.

By default, the Status Bar includes the following components:

MODEL – Workspace indicator showing the name of the active environment (Model or Layout).

– Grid Display – toggles the display of the drawing grid on and off.

– Snap Mode – turns cursor snapping to the grid on or off (defines the step of cursor movement).

– Drawing Modes Group, which includes several essential toggles:

– Infer Constraints – automatically applies geometric constraints when creating or editing objects.

– Ortho Mode – restricts cursor movement to horizontal or vertical directions along the Cartesian axes.

– Polar Tracking – limits cursor movement to specified angular increments.

– Isometric Drafting – enables isometric 3D object representation aligned with one of the three primary isometric axes. The drop-down list allows the user to select the isoplane for axonometric projection (see Fig. 2.19).



Fig. 2.19. Isoplane selection list for isometric projections

 – Tracking (Object Snap Tracking) – enables or disables cursor tracking along alignment paths based on object snap points.

 – Object Snap (OSNAP) – turns on or off precise snapping to key points of geometry, such as endpoints, midpoints, or centers.

 – Lineweight – displays or hides line thicknesses on the screen.

 – Selection Cycling – enables or disables the ability to select overlapping objects. When multiple objects are found under the cursor, an icon  appears indicating that several selectable entities are available. Right-clicking opens the Selection window with the list of overlapping objects (see Fig. 2.20).

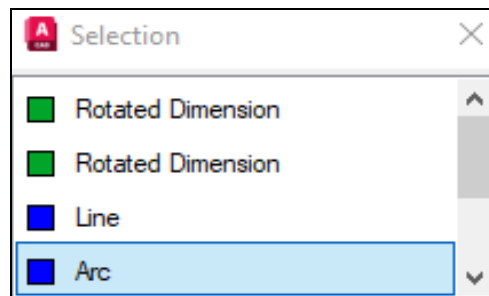


Fig. 2.20. Selection window listing overlapping objects

To choose the desired object in the Selection window, hover the cursor over its name and click the left mouse button.

 – The group of buttons is intended for 3D modeling, where:

 – 3D Object Snap (3DOSNAP) – enables or disables snapping to 3D reference points.

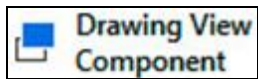
 – 3D Object Snap Filter – activates or deactivates filtering of 3D elements and viewports:

 Vertex – Vertex points.

 Edge – Edges.

 Face – Faces

 – controls the display of history for composite solids

 – manages components of drawing views.

The filtering function can be disabled using the Shift+F1 key combination.

 – Gizmo – provides graphical dynamic controls for moving, rotating, and scaling 3D objects relative to coordinate axes and planes.

 – Annotation Scale and Visibility – a group of buttons that control annotation display and scale settings.

 – Workspace Switching – allows changing the current workspace.

 – Annotation Monitor – enables or disables the display of non-associative dimensions.

 – Units – opens the list of measurement units used in the drawing.

 – Quick Properties – toggles the display of the Quick Properties window, which appears automatically next to a selected object, showing its main parameters (see Fig. 2.21).

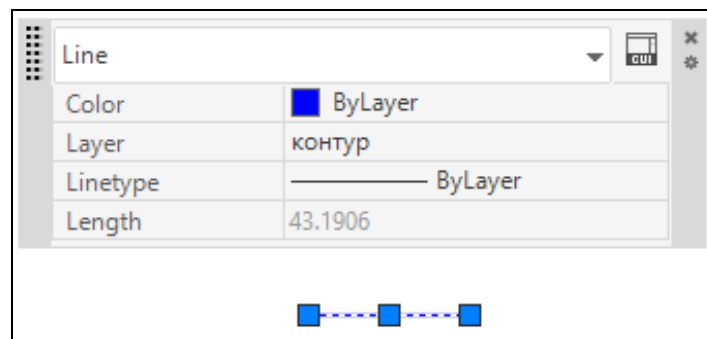


Fig. 2.21. Quick Properties window

 – Lock/Unlock UI – locks or unlocks interface elements to prevent accidental repositioning.

 – Isolate Objects – temporarily hides unselected objects or restores previously hidden ones when toggled.

 – Clean Screen – hides interface elements to maximize the drawing area.

 – Customization – opens a drop-down list of all available Status Bar buttons; active ones are marked with checkboxes.

To add or remove specific functions, simply click the left mouse button on the corresponding name in the Customization menu of the Status Bar. Active modes are highlighted with a frame and a light blue background. In addition to the previously mentioned options, the Status Bar allows users to include the following modes and functions:

 – Coordinates – displays the current cursor coordinates, which can be static or dynamic, showing either linear or angular values. In the static mode, the coordinates are updated only when a specific point is selected, while in the dynamic mode they change continuously as the cursor moves.

 – Transparency – adjusts the transparency level of selected objects.

 – Dynamic Input – provides a dynamic interface for entering data, consisting of value fields and contextual prompts. Any field can be activated by pressing the Tab key.

 – Dynamic UCS (User Coordinate System) – temporarily aligns the XY plane of the UCS with the face of a solid, grid cell, or other surface when the cursor hovers over it. After the current command is completed, the UCS automatically returns to its previous settings.

 – Graphics Performance – controls hardware acceleration using the computer's graphics processor (GPU). Settings for this function can be adjusted in the Graphics Performance dialog box, which opens by right-clicking the icon on the Status Bar.

Some of these parameters can also be toggled using function keys:

- F3 – Object Snap (OSNAP) ;
- F4 – 3D Object Snap (3DOSNAP) ;
- F6 – Dynamic UCS ;
- F7 – Grid Display ;
- F8 – Ortho Mode ;
- F9 – Snap Mode ;
- F10 – Polar Tracking ;
- F11 – Object Snap Tracking .

2.2.8. Context menu

AutoCAD provides a convenient and intuitive tool for performing context-sensitive actions – the Context Menu. This menu offers quick access to the most relevant commands without the need to search through the Ribbon or toolbar, thereby significantly accelerating the workflow. Its content changes dynamically depending on the active command, the selected objects, or the area of the interface where the cursor is positioned – for example, within the drawing area, a viewport, or the command line. The menu is opened by right-clicking the mouse in the desired part of the drawing window, which makes it an essential part of contextual interaction between the user and the system.

If no command is active and the cursor is placed within the drawing area, the standard context menu appears (see Fig. 2.22). The first item in this menu allows the user to repeat the previously executed command, providing a fast way to perform repetitive operations. The figure shows the context menu for a new drawing, where the recommended initial option is Help, which gives access to AutoCAD’s built-in documentation. The Recent Input list displays the most recently entered commands, while the Clipboard submenu provides various options for cutting, copying, and pasting objects, including advanced features such as Copy with Base Point and Paste to Original Coordinates (see Fig. 2.23).

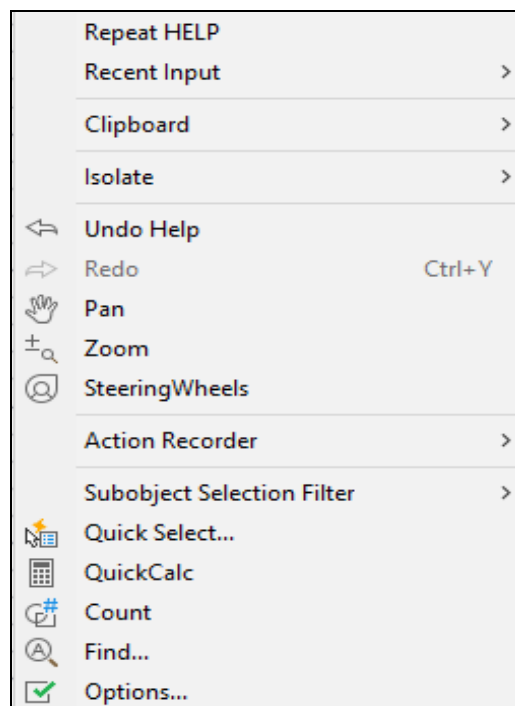


Fig. 2.22. Standard context menu

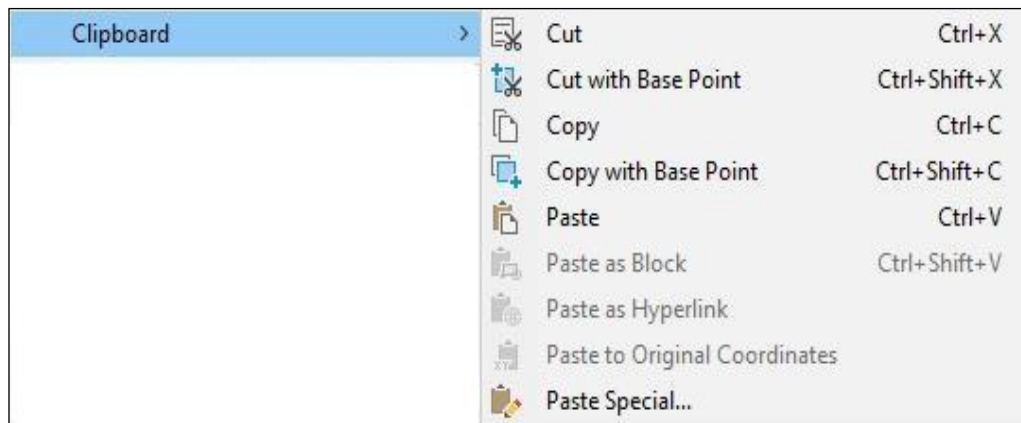


Fig. 2.23. Clipboard drop-down menu

The Isolate command in the context menu offers the following actions: Isolate Objects, Hide Objects, and End Object Isolation (see Fig. 2.24).

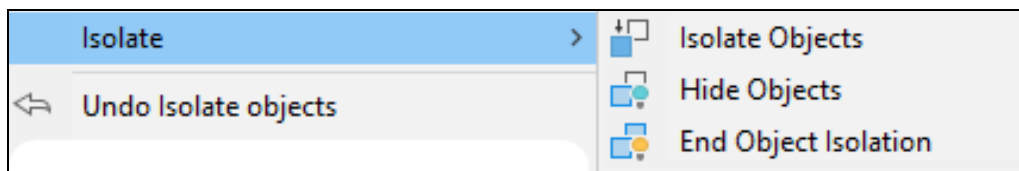


Fig. 2.24. Isolate command options in the context menu

The difference between Isolate Objects and Hide Objects is that in the first case all other objects become invisible except the selected ones, whereas in the second case only the selected objects are hidden. The End Object Isolation option restores the display of all previously hidden objects.

In addition, the context menu provides quick access to view control commands such as Pan and Zoom, the Steering Wheel navigation tool, Action Recorder, selection filters, Quick Select, QuickCalc, Find, and Options.

2.3. Two-Dimensional Modeling

This section introduces two-dimensional geometric modeling in AutoCAD. The concepts presented here form the foundation for effective work in three-dimensional modeling later in the course.

2.3.1. Coordinate System

The position of any drawing element in AutoCAD is defined by its coordinates. By default, AutoCAD uses a three-dimensional rectangular coordinate system consisting of three mutually perpendicular axes – X, Y,

and Z – known as the World Coordinate System (WCS). This fixed global system establishes the location of all objects in the drawing.

In 2D mode, points are defined within the XY plane, also referred to as the drawing's working plane. The X value specifies the horizontal distance from the origin (0,0,0), while the Y value defines the vertical distance. A positive X direction extends to the right, negative X to the left; similarly, positive Y extends upward, and negative Y downward.

In the polar coordinate system, a point is defined by a distance and an angle measured counterclockwise from the horizontal axis.

In both Cartesian and polar systems, coordinates can be entered in either absolute or relative form.

Absolute coordinates are referenced to the origin (0,0), while relative coordinates (preceded by the symbol @) are specified with respect to the previous point.

The User Coordinate System (UCS) is a movable rectangular coordinate system that allows users to define custom orientations for drawing or modeling.

It is created and controlled using the UCS command, entered in the Command Line:

Command: UCS

*Current UCS name: *WORLD* – indicates the current coordinate system*

Specify origin of UCS or [Face/NAmed/OBject/Previous/View/World/X/Y/Z /ZAxis] <World>: – options are available for defining the following User Coordinate System:

Face – aligns the UCS with the face of a solid, surface, or mesh segment;

NAmed – bselects from a list of previously saved coordinate systems;

OBject – selects from a list of previously saved coordinate systems;

Previous – restores the previously used UCS;

View – aligns the UCS to the current view;

World – restores the World Coordinate System (WCS);

X/Y/Z – rotates the UCS by a specified angle about the corresponding axis;

ZAxis – defines a new Z-axis direction, adjusting the UCS orientation accordingly.

Access to UCS management commands is also available from the Tools tab in the Classic Menu.

The list of options in the New UCS component is shown in Fig. 2.25, which presents the available parameters for defining or modifying the User Coordinate System (UCS).

These include alignment by face, object, or view, restoring the world coordinates, and rotation around the X, Y, or Z axes.

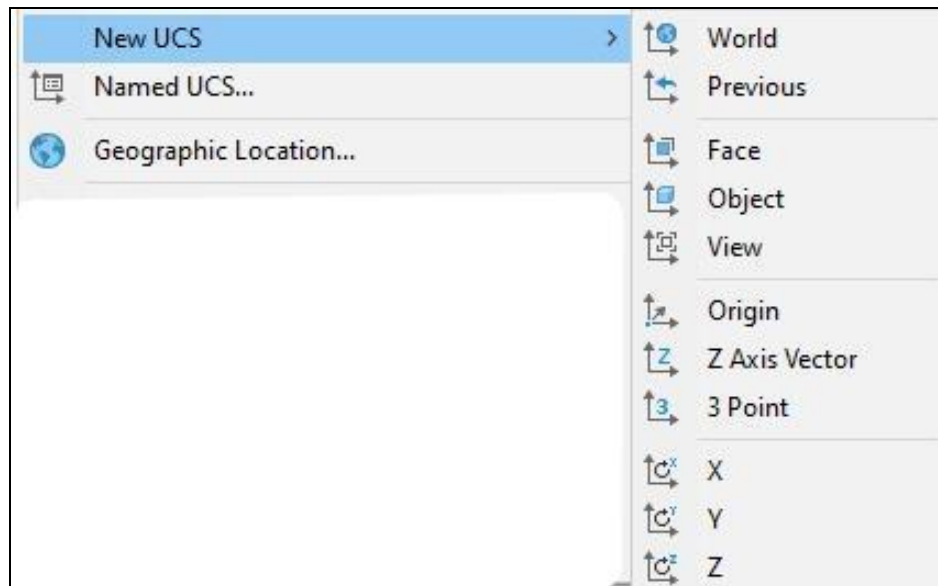


Fig. 2.25. UCS command options

The main options of the UCS command provide various ways to define or modify the User Coordinate System:

-  **World** – restores the World Coordinate System (WCS).
-  **Previous** – reverts to the previously used UCS.
-  **Face** – aligns the UCS with the face of a solid object.
-  **Object** – sets the UCS based on an existing 3D object.
-  **View** – aligns the UCS with the current view direction.
-  **Origin** – defines a new origin point and orientation for the coordinate axes.
-  **Z Axis Vector** – changes the orientation of the coordinate system according to the direction of the Z-axis.
-  **3 Point** – redefines the UCS using three points: the first specifies the origin, the second defines the positive X-axis, and the third sets the positive Y-axis direction.
-  **X**,  **Y**,  **Z** – rotates the UCS around the selected coordinate axis.

By default, in a new drawing, the User Coordinate System (UCS) coincides with the World Coordinate System (WCS). The orientation of the X and Y axes is shown by the UCS icon located in the lower-left corner of the drawing area.

According to the right-hand rule, the Z-axis is directed outward toward the user. The small square at the intersection of the axes (see Fig. 2.26) indicates that the UCS is aligned with the WCS.

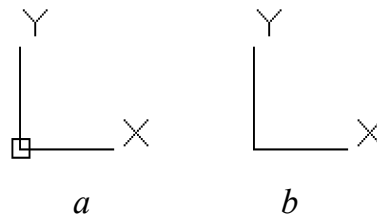


Fig. 2.26. Coordinate system icons:

a — World Coordinate System (WCS); *b* — User Coordinate System (UCS)

In 2D mode, the UCS defines the XY plane, also referred to as the working or construction plane, where objects are created and edited. In layout space, the UCS is displayed as a triangular icon, indicating its orientation within the sheet's workspace.

2.3.2. Units of Measurement

In AutoCAD, lengths and angles are measured in units defined for the current drawing, ensuring that all geometric models are created at their true scale. When printing, the drawing is scaled by specifying the ratio between the drawing units and the paper units.

AutoCAD supports 22 linear units, ranging from micrometers to parsecs. By default, Decimal units are used for linear dimensions. Alternative formats include Engineering, Architectural, Fractional, and Scientific units.

For angular measurements, the default format is Decimal Degrees, though users can also select Degrees/Minutes/Seconds (DMS), Grads, Radians, or Surveyor's Units. The choice depends on the specific requirements of the project. By default, angles increase counterclockwise from the horizontal axis.

The display precision is determined by the number of decimal places shown after the point. By default, linear values are displayed to four decimal places, while angular values are shown as whole numbers without fractional parts.

Units and precision are set using the UNITS command, which opens the Drawing Units dialog box (see Fig. 2.27).

This window allows the user to configure the length and angle formats, precision, and insertion scale for imported content. To reverse the direction of angle measurement, enable the Clockwise checkbox.

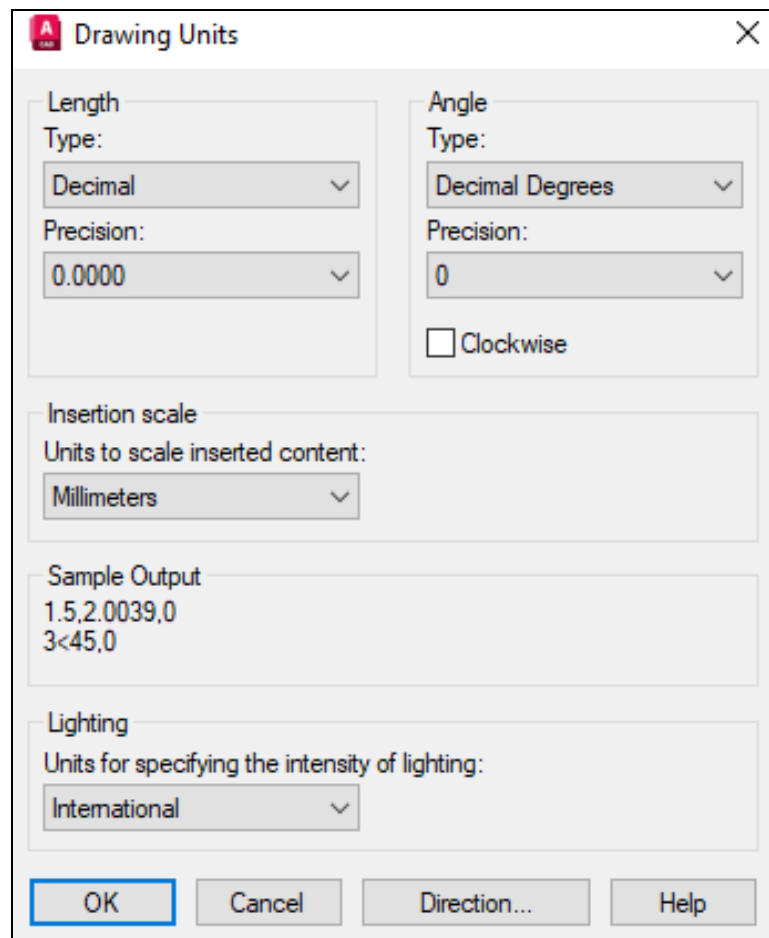


Fig. 2.27. Drawing Units dialog box

For existing dimensions, measurement units can be modified through the Properties panel of the context menu. Under the Primary Units tab, the Dim Units drop-down list allows the selection of the required unit type, while the Precision field defines the numerical accuracy of displayed values.

This flexibility enables users to adjust drawings for various regional standards or project documentation requirements without redefining geometry.

Linear measurement units can also be set via the Units button located on the Status Bar, providing quick access to commonly used unit formats.

The corresponding drop-down menu is shown in Fig. 2.28, where Decimal is displayed by default. Changing the format here instantly updates the measurement display across the drawing, ensuring consistency and simplifying the preparation of technical documentation.

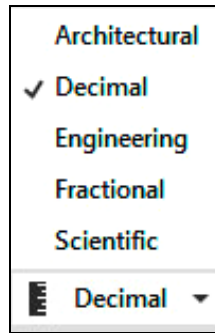


Fig. 2.28. Drop-down list of units on the Status Bar

The corresponding drop-down menu is shown in Fig. 2.28, where Decimal is displayed by default.

2.3.3. Drawing Limits

If a new drawing is created without using the standard acadiso.dwt template, it becomes necessary to define its drawing limits manually.

When this template is used, AutoCAD automatically applies the metric system of measurement and sets the default drawing area to 420×297 mm (A3 format).

These values can be modified by entering the LIMITS command in the Command Line or by selecting Drawing Limits from the Format tab in the Classic Menu.

After invoking the command, the following prompt appears in the Command Line:

Specify lower left corner or [ON/OFF] <0.0000,0.0000>:

At this prompt, specify the lower-left corner of the rectangular drawing boundary, or choose one of the listed options. Once the first point is defined, AutoCAD displays the next prompt:

Specify upper right corner<420.0000, 297.0000>:

Here, the user must define the upper-right corner of the drawing limits to replace the default values shown in the angle brackets.

Command options:

ON – enables automatic control to prevent the creation of objects outside the defined drawing limits.

OFF – disables this control, allowing objects to be drawn beyond the specified area.

2.3.4. Display Control

AutoCAD provides two primary commands for managing the view and display of a drawing on the screen: ZOOM and PAN. The ZOOM command controls the magnification of the drawing view, while PAN enables movement of the visible area within the drawing plane without altering the scale.

The ZOOM command can be activated by typing its name in the Command Line or by using the Navigation Bar.

It includes several options, as shown in Fig. 2.29.

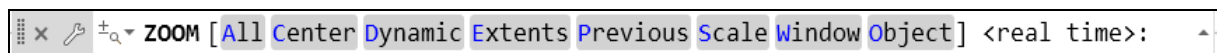


Fig. 2.29. Zoom command options

The option displayed in angle brackets is the default one. To select it, simply press Enter.

Real time – increases or decreases the view interactively by moving the cursor while holding down the left mouse button.

All – displays the entire drawing within the defined drawing limits.

Center – displays the entire drawing within the defined drawing limits.

Dynamic – uses a movable frame to define and pan a visible portion of the drawing interactively.

Extents – zooms to show all visible objects in the drawing, fitting them to the screen.

Previous – restores the previous view before the last zoom operation.

Scale – enlarges or reduces the view by a specified factor. If the number is followed by x, scaling is relative to the current view; if not, it is relative to the drawing limits.

Window – zooms in on a rectangular area defined by two opposite corners.

Object – zooms in on one or more selected objects, fitting them to the current view.

All of these options are also available on the Navigation Bar, where the Extents mode is set by default. Zoom control can conveniently be performed with the mouse wheel: scrolling forward zooms in, scrolling backward zooms out, and double-clicking the wheel performs a Zoom Extents operation to display the entire drawing.

The PAN command can be accessed from the Command Line or directly from the Navigation Bar. Holding down the mouse wheel while moving the mouse also performs panning, allowing the user to navigate smoothly across the drawing area without changing the zoom level.

2.3.5. Object Properties

Object properties define the appearance and behavior of elements in a drawing. Each object in AutoCAD possesses a set of general properties, which include Color, Layer, Linetype, Linetype Scale, Lineweight, Transparency, and Thickness. In addition to these, every object also has geometric properties (Geometry), such as the coordinates of characteristic points, length, area, elevation, and other parameters. All these properties can be viewed and modified in the Properties palette, an example of which is shown in Fig. 2.30.

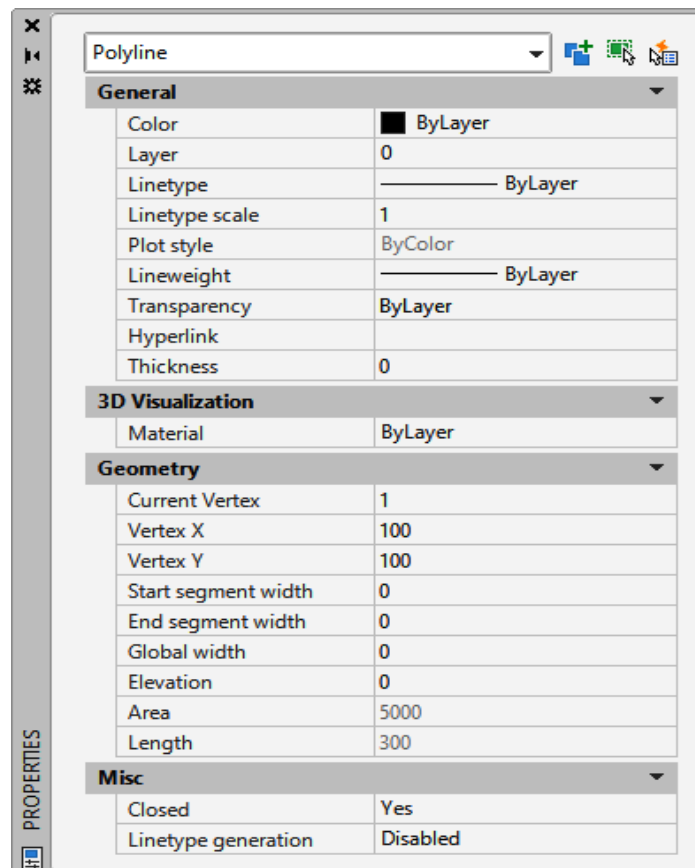


Fig. 2.30. Properties palette for a Polyline

The content of the Properties window changes dynamically depending on the selected object. If no object is selected, as illustrated in Fig. 2.31, the window displays default settings applicable to the drawing as a whole.

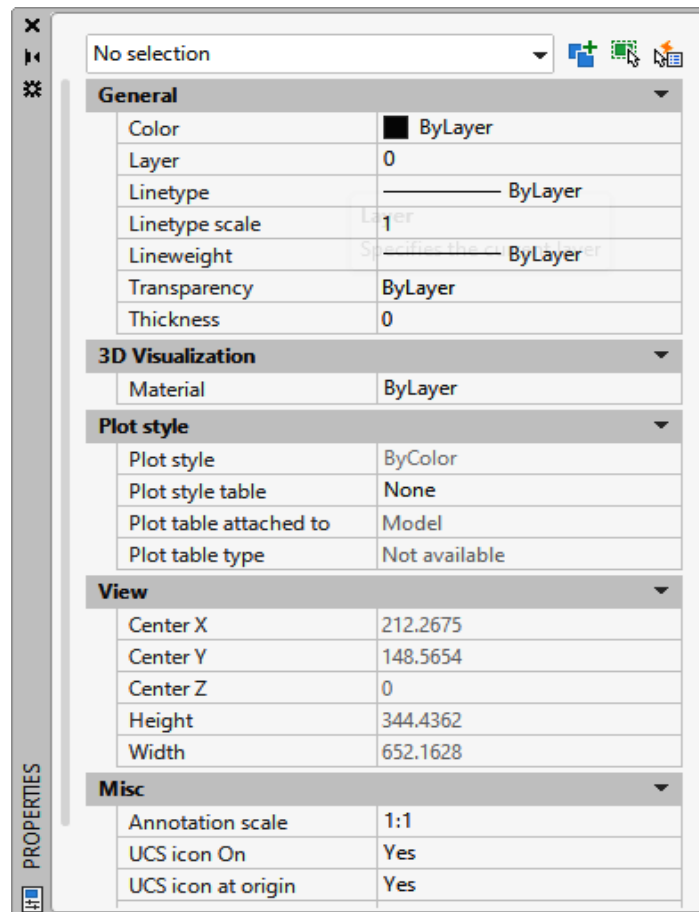


Fig. 2.31. Properties palette with default settings

By default, general properties such as Color, Linetype, and Lineweight are assigned the value ByLayer, meaning that these attributes are inherited from the active layer settings.

The management and customization of layers, as well as their influence on object appearance, are discussed in the following section of this textbook.

2.3.6. Layer Management

In AutoCAD, layers can be compared to transparent sheets, each containing objects with shared properties. Together, these layers form the complete drawing, while individual layers can be displayed or hidden selectively as needed.

By default, every new drawing contains only one layer – Layer 0 (zero layer). It is not recommended to place drawing objects on this layer, as it serves as a system default used for creating blocks and inheriting object properties.

To organize objects by their characteristics, AutoCAD provides the Layer Properties Manager tool (see Fig. 2.32).

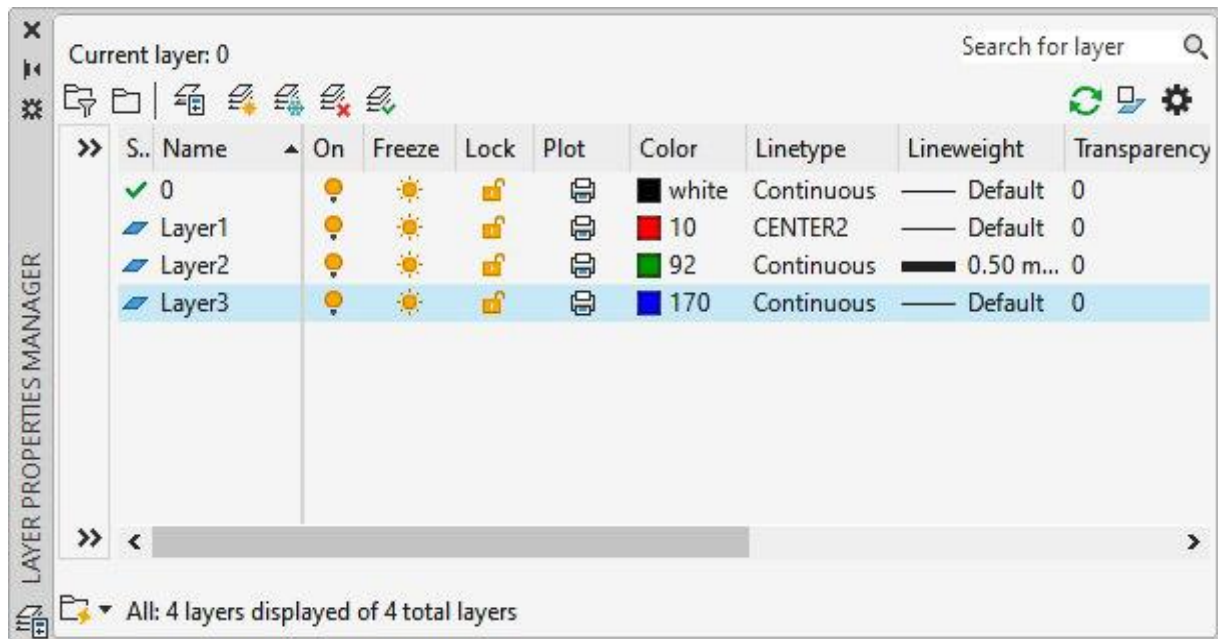


Fig. 2.32. Layer Properties Manager dialog box

This window can be opened from the Format tab of the Classic Menu using the Layer command. It allows users to create, delete, rename, and modify layers, as well as manage their properties and status.

The central area of the window contains a table listing all layers and their parameters, while the toolbar above includes buttons for creating a new layer, deleting unused ones, or setting the current layer.

The current and zero layers cannot be deleted. The columns in the layer list represent the following properties:

On – toggles layer visibility; turned-off layers are not displayed or plotted.

Freeze – freezes or unfreezes layers; frozen layers are hidden from view and excluded from plotting.

Lock – locks or unlocks layers; locked layers are visible and selectable but cannot be edited.

Plot – controls whether the layer is included in printing.

Color – sets the color for objects on the layer. By right-clicking the color square icon, the Select Color dialog box appears (see Fig. 2.33), allowing the user to choose the desired color.

Linetype – specifies the line pattern used for layer objects. By default, all layers use the Continuous linetype. To assign another pattern, use the Select Linetype dialog box (see Fig. 2.34).

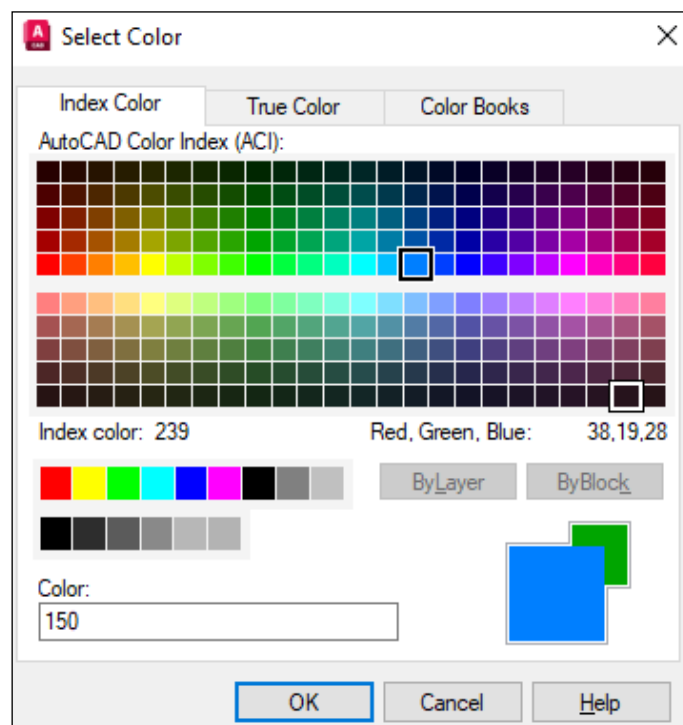


Fig. 2.33. Select Color dialog box

This window is opened by clicking the left mouse button on the linetype name of the desired layer.

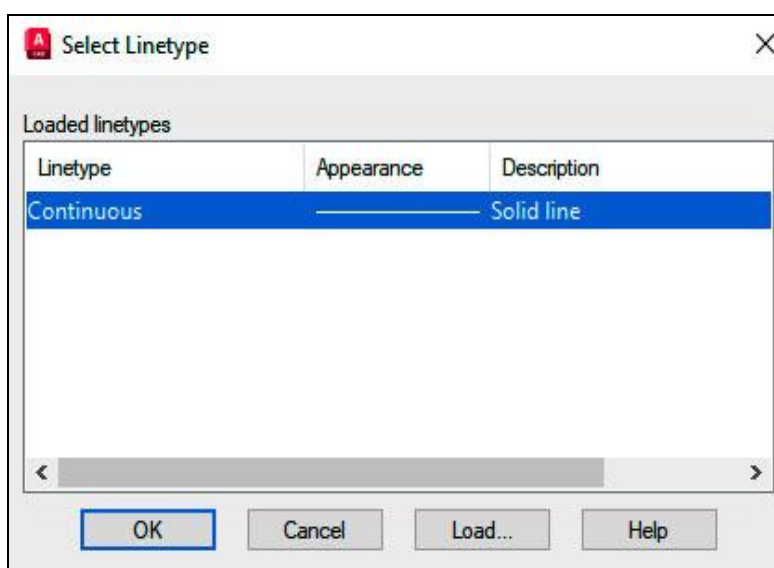


Fig. 2.34. Select Linetype dialog box

If the desired linetype has not been loaded previously, only Continuous will appear in the list.

Clicking Load opens the Load or Reload Linetypes dialog box, which accesses the predefined linetypes stored in the acadiso.lin file (see Fig. 2.35).

If the required linetypes have not been used previously, the window will display only the default Continuous linetype. The Load button allows access to the Load or Reload Linetypes dialog box, which contains the standard linetypes supplied with AutoCAD in the acadiso.lin file (see Fig. 2.35).

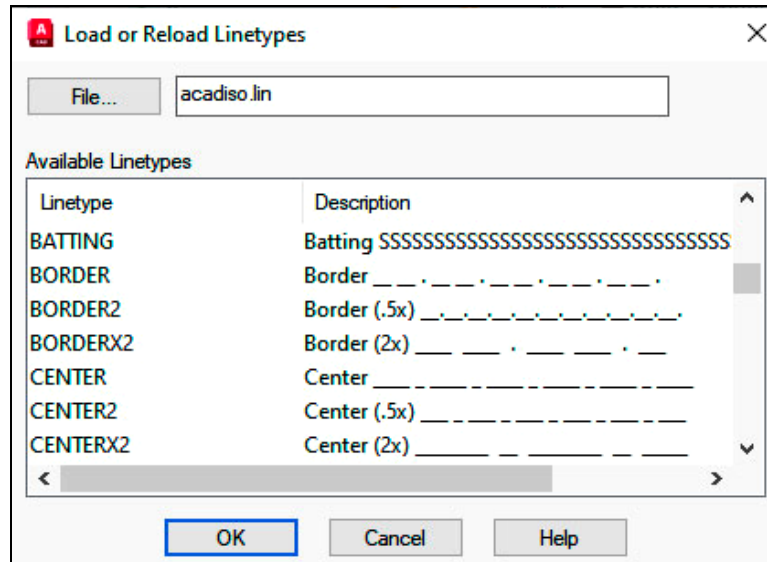


Fig. 2.35. Load or Reload Linetypes dialog box

Lineweight – the line thickness is assigned by clicking the current lineweight value in the Layer Properties Manager table with the left mouse button.

This action opens the Lineweight dialog box (see Fig. 2.36), where the required lineweight can be selected.

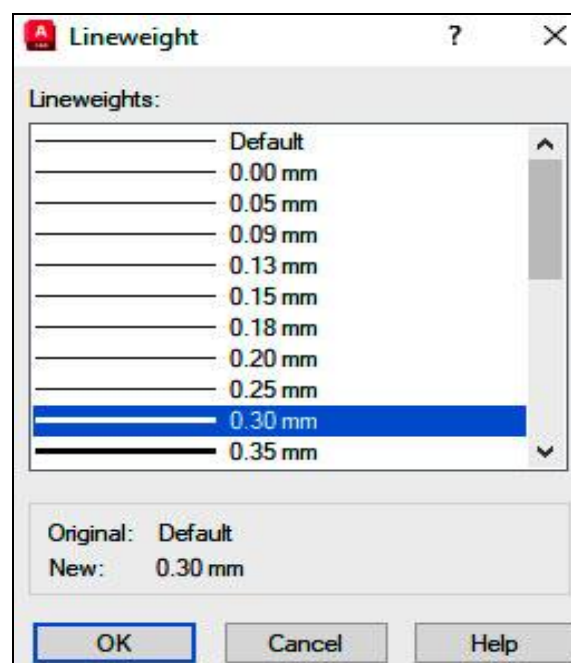


Fig. 2.36. Lineweight dialog box

Transparency – defines the transparency level of layer objects. To open the Layer Transparency dialog box (see Fig. 2.37), click the current transparency value in the Layer Properties table. In the dialog field, specify the required transparency percentage.

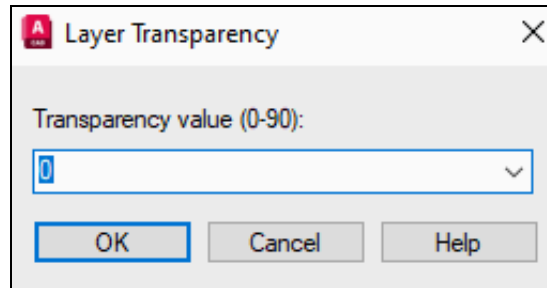


Fig. 2.37. Layer Transparency dialog box

By default, new layers receive names such as Layer1, Layer2, Layer3, and so on. However, it is advisable to assign each layer a descriptive name that reflects its properties or intended purpose. A layer name may contain from 1 to 31 characters.

New layers are created using the New Layer button, unnecessary layers are removed with Delete Layer, and the Set Current button designates the active layer for drawing operations.

2.3.7. Object Selection

Many AutoCAD commands require the selection of objects, which is indicated by the prompt Select objects in the Command Line. When this prompt appears, the mouse cursor turns into a selection square, allowing the user to select objects one by one. Selected objects are highlighted with a dashed outline. Press Enter to complete the selection process.

Auto Mode. Place the selection square on an empty area of the drawing so that it does not touch any objects. Click the left mouse button to set the first corner of the selection window. By moving the cursor to the right, AutoCAD creates a rectangular selection window whose opposite corner is defined with another left-click. All objects that are completely enclosed by this window become selected (see Fig. 2.38). If the cursor is moved to the left, AutoCAD creates a crossing window (see Fig. 2.39). In this case, all objects that are crossed by the window boundary are selected. Additionally, AutoCAD provides free-form selection, which is activated by moving the mouse while holding down the left button to create an arbitrary closed shape (see Fig. 2.40).

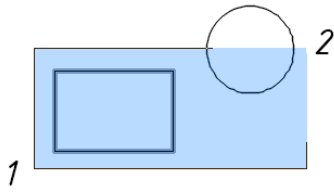


Fig. 2.38. Selection window

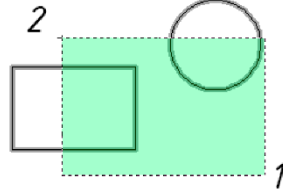


Fig. 2.39. Crossing window

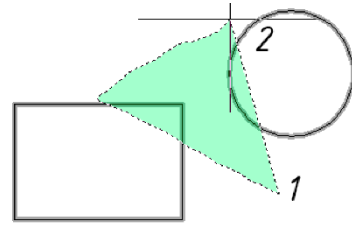


Fig. 2.40. Free-form selection

ALL Mode. This mode is activated by typing ALL in the Command Line or by pressing Ctrl + A. It selects all objects in the drawing except those located on locked or frozen layers.

Object Snaps (Osnaps). Additional tools for precise selection of characteristic points of existing objects during drawing are the Object Snaps (see Table 2.1). Access to object snap settings is available through the Status Bar.

Table 2.1

Object Snaps

Icon	Name	Snap Function
	Endpoint	snaps to the endpoint of an object
	Midpoint	snaps to the midpoint
	Center	snaps to the center of an arc or circle
	Geometric Center	snaps to the centroid of a closed polyline or spline
	Node	snaps to a point object
	Quadrant	snaps to quadrant points of a circle, arc, or ellipse
	Intersection	snaps to the intersection of two objects
	Extension	snaps to the extension of an object beyond its endpoint
	Insertion	snaps to the insertion point of a block or text
	Perpendicular	snaps to a point where a line from your previous point would be perpendicular to the selected object
	Tangent	snaps to a point where a line from your previous point would be tangent to an arc or circle
	Nearest	snaps to the nearest point on an object
	Apparent Intersection	snaps to the visual intersection of two objects that do not physically intersect
	Parallel	constrains the next line segment to be parallel to an existing line

3.3.8. Coordinate Entry Methods

AutoCAD provides five methods for specifying coordinates when defining points in the drawing.

Interactive Method. The point is located directly on the XY plane by selecting its position with the mouse.

Absolute Coordinates. The X and Y values are entered relative to the origin (0,0) using the keyboard. Format: X, Y.

Relative Rectangular Coordinates. The point is defined by its displacement along the coordinate axes relative to the last specified point. The offset values (ΔX , ΔY) are entered as: Format: @ ΔX , ΔY .

Relative Polar Coordinates. The location is defined by a polar distance R and an angle ω , both measured from the last point. Format: @R< ω .

Direction and Distance Method. The point is defined by specifying a direction with the mouse (using Ortho or Polar Tracking mode) and then entering the distance from the keyboard. The cursor movement only sets the angle, while the numeric entry determines the exact distance.

Object Snap Method. The point is specified using Object Snaps, which place points precisely at characteristic locations on existing geometry.

2.3.9. Construction of Geometric Primitives

This section presents the methods for constructing basic geometric figures, or primitives, in the AutoCAD environment. These include points, line segments, circles and ellipses, their arcs, polygons, splines, and other objects. Access to these tools can be achieved through the Command Line by entering the required command name, through the DRAW tab of the Classic Menu, or via the Ribbon. In the examples that follow, the Ribbon method is used.

2.3.9.1. Points

The creation of these objects is performed using the POINT command, which can be accessed as follows: Ribbon → HOME tab → DRAW panel →  POINT command.

After launching the command, the following prompt appears:

Specify a point: – the point is defined by entering coordinates or using another method.

This prompt repeats until you exit the command, for example by pressing the Esc key.

2.3.9.2. Line Segments

The LINE command is used to create a sequence of straight-line segments, where each segment is treated as an individual object and can be edited independently. Access to this command is provided as follows: Ribbon → HOME tab → DRAW panel →  LINE command.

Options:

Undo – cancels the last action.

Close – closes the contour (becomes available after the third point is entered).

Let us examine the use of the LINE command (Fig. 2.41), specifying points using different methods.

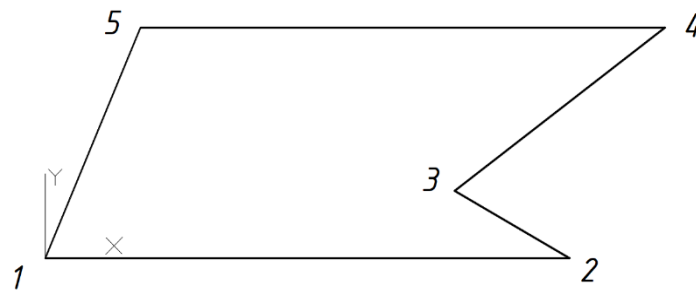


Fig. 2.41. Contour created using the LINE command

Command: LINE

Specify first point: point 1 is defined using absolute coordinates.

Specify next point or [Undo]: <Ortho on> 100 – point 2 is created using the Direction and Distance method with Ortho mode enabled.

Specify next point or [Undo]: <Polar on> 25 – point 3 is defined using the Direction and Distance method in Polar mode with an angle of 30°.

Specify next point or [Close/Undo]: @40<30 – point 4 is defined using relative polar coordinates.

Specify next point or [Close/Undo]: @-100,0 – point 5 is created using relative rectangular coordinates.

Specify next point or [Close /Undo]: C – connects back to point 1.

2.3.9.3. Circles and Arcs

The CIRCLE command is used to construct circles and offers several options for creating them in different ways. Access to this command is provided as follows: Ribbon → HOME tab → DRAW panel →  CIRCLE command.

Options:

-  *Center, Radius* – specify the center and radius of the circle.
-  *Center, Diameter* – specify the center and diameter.
-  *2 Point* – select two points that define the diameter of the circle

(Fig. 2.42, a).

-  *3 Point* – select three points that lie on the circle (Fig. 2.42, b);

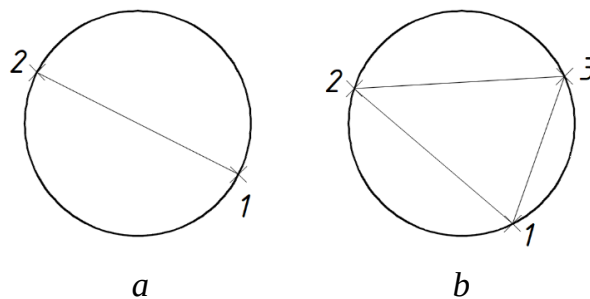


Fig. 2.42. Constructing a circle:

a – using two points; b – using three points

 *Tan, Tan, Radius* – specify two tangent objects and the radius (Fig. 2.43, a).

-  *Tan, Tan, Tan* – specify three tangent objects (Fig. 2.43, b).

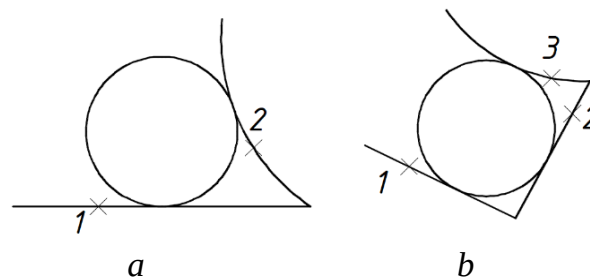


Fig. 2.43. Constructing a circle:

a – using two tangent objects; b – using three tangent objects

Arcs are created using the ARC command, accessed as follows: Ribbon → HOME tab → DRAW panel →  ARC command.

AutoCAD provides 11 methods for constructing arcs.

 *3-Point* – defined by three points (Fig. 2.44), where the selection order determines the arc shape.

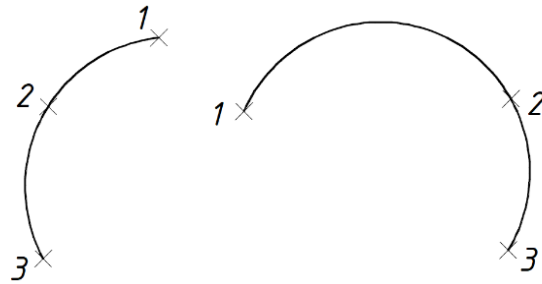


Fig. 2.44. Arc defined by three points

For all other methods, note that arcs are created counterclockwise (Fig. 2.45, a). Holding CTRL allows creating arcs clockwise (Fig. 2.45, b).

This behavior is illustrated using the Start–Center–End method.

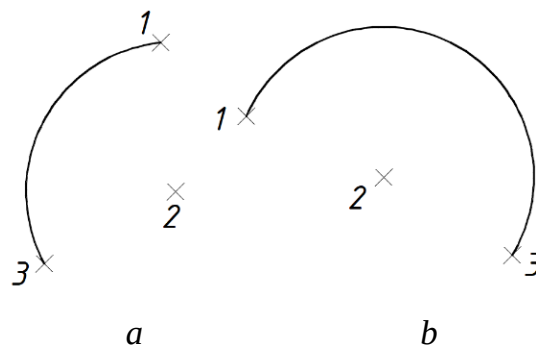


Fig. 2.45. Arc construction:

a, b – counterclockwise and clockwise

-  Start, Center, End – specify the start point, center, and end point.
-  Start, Center, Angle – specify the start point, center, and arc angle.
-  Start, Center, Length – specify the start point, center, and chord length.
-  Start, End, Angle – specify the start and end points and the angle.
-  Start, End, Direction – specify the start and end points and the tangent direction (Fig. 2.46). The tangent direction is defined by selecting a point that determines the direction vector from the start point.

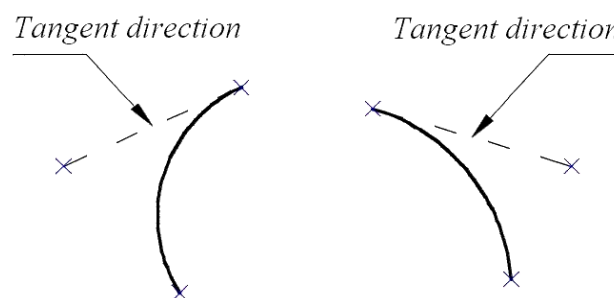


Fig. 2.46. Arcs defined by two points and a tangent

-  Start, End, Radius – specify the start and end points and the radius.
-  Center, Start, End – specify the center, start, and end.
-  Center, Start, Angle – specify the center, start, and arc angle.
-  Center, Start, Length – specify the start and end points and the chord length.
-  Continue – creates an arc tangent to the last created object (either an arc or a line), see Fig. 2.47.

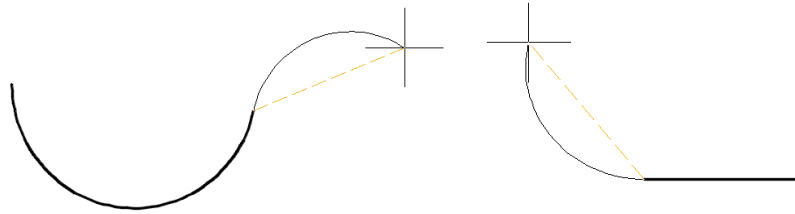


Fig. 2.47. Arc tangent to the previously created object

The next subsection discusses the construction of second-order curves.

2.3.9.4. *Ellipses and Elliptical Arcs*

The ELLIPSE command is used to construct ellipses and elliptical arcs. It is accessed as follows: Ribbon → HOME tab → DRAW panel →  ELLIPSE command.

Depending on the selected option, AutoCAD provides three methods for creating ellipses.

Center – specify the center (point 1) and the endpoint of one axis (point 2), see Fig. 2.48. Then the following prompt appears:

Specify distance to other axis or [Rotation]: – specify the endpoint of the second axis (point 3).

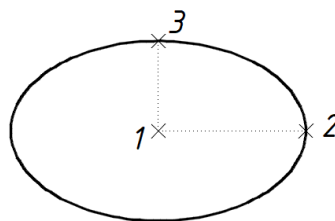


Fig. 2.48. Constructing an ellipse using the center and semi-axes

The *Rotation* option allows constructing an ellipse as a projection of a circle on a plane rotated about its diameter.

 **Axis, End** – specify the first axis of the ellipse by selecting its two endpoints. Then define the second axis.

To construct an ellipse as an axonometric projection of a circle, enable the Isodraft  function on the Status Bar. From the drop-down menu (Fig. 2.49), select the appropriate projection plane.

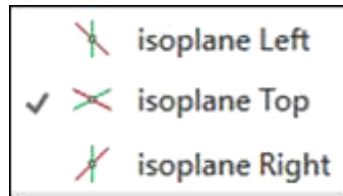


Fig. 2.49. Selecting an isometric plane

When Isodraft mode is enabled, the ELLIPSE command includes the *Isocircle* option::

Command: ELLIPSE

Specify axis endpoint of ellipse or [Arc/Center/Isocircle]: I

Specify center of isocircle: – specify the center point;

Specify radius of isocircle or [Diameter]: – define the radius (or diameter).

This produces a projection of a circle onto the selected plane (Fig. 2.50).

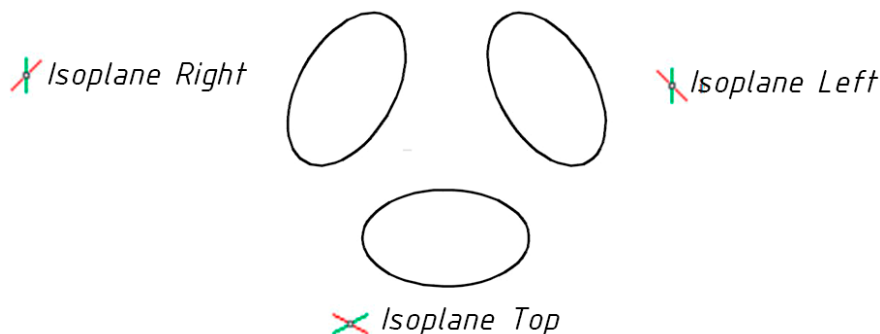


Fig. 2. 50. Ellipses as circle projections depending on the isometric plane

To construct an elliptical arc, select the Arc option:

Command: ELLIPSE

Specify axis endpoint of ellipse or [Arc/Center/Isocircle]: a

Specify axis endpoint of elliptical arc or [Center/Isocircle]: I

Specify center of isocircle:

Specify radius of isocircle or [Diameter]:

Specify start angle or [Parameter]: – specify the start angle of the arc;

Specify end angle or [Parameter/Included angle]: – specify the end angle of the arc. The start and end angles define the portion of the ellipse that forms the arc.

 The Elliptical Arc option allows creating an elliptical arc directly. First, AutoCAD prompts for the construction of the full ellipse, and then for the start and end angles of the arc (Fig. 2.51).

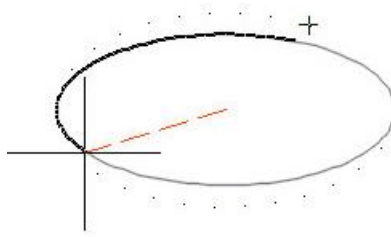


Fig. 2.51. Constructing an elliptical arc

Specify start angle or [Parameter]: – specify the starting point of the arc;

Specify end angle or [Parameter/Included angle]: – specify the endpoint of the elliptical arc.

The *Included angle* option allows specifying the central angle of the arc measured from the start angle.

2.3.9.5. Polylines

A polyline is a composite object consisting of one or more connected linear or arc segments. Each segment may have its own width (Fig. 2.52).

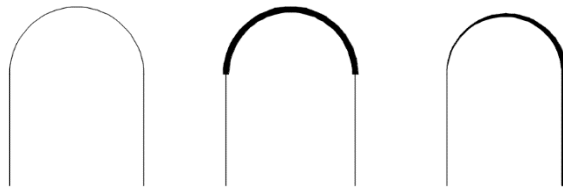


Fig. 2.52. Examples of polyline segment width variations

Regardless of the number of segments, AutoCAD treats the entire polyline as a single object.

Access to this command is provided as follows: Ribbon → HOME tab → DRAW panel →  PLINE command. After launching the command, the following prompts appear:

Command: PLINE

Specify start point: – specifies the starting point of the polyline.

Current linewidth is 0.0000 – displays the current segment width.

Specify next point or Arc/Halfwidth/Length/Undo/Width/Close]: – specifies the next point (for a linear segment) or selects one of the options:

Arc – switches to arc-segment creation mode:

Arc – перехід у режим побудови дуг.

Specify next point or [Arc/Halfwidth/Length/Undo/Width]: A

Specify endpoint of arc (hold Ctrl to switch direction) or Specify endpoint of arc (hold Ctrl to switch direction) or

[Angle/CEnter/Direction/Halfwidth/Line/Radius/Second pt/Undo/Width]:

Halfwidth – sets the half-width of the polyline segment.

Length – defines the length of the next straight segment.

Undo – cancels the creation of the last segment.

Width – sets the width of the next segment.

Close – connects the endpoint of the last segment to the starting point of the polyline and ends the command.

2.3.9.6. Rectangles

Execution of the RECTANGLE command is performed as follows: Ribbon → HOME tab → DRAW panel →  RECTANGLE command.

It constructs a polyline in the shape of a rectangle based on its parameters (length, width, rotation angle) and the type of corner treatment (chamfer, fillet, or right angle).

Command: RECTANG

Specify first corner point or [Chamfer/Elevation/Fillet/Thickness/Width:

Options:

Specify first corner point – defines the first vertex.

Chamfer – creates a rectangle with chamfered corners of specified sizes (Fig. 2.53, c).

Elevation – sets the elevation of the rectangle in the Z direction (Fig. 2.53, f).

Fillet – rounds the corners of the rectangle with a user-defined radius (Fig. 2.53, d).

Thickness – creates a 3D prism of a specified height with the rectangle as its base (Fig. 2.53, g).

Width – sets the polyline width (Fig. 2.53, e).

If no option is selected and the first corner point has been specified, the command provides additional options for various rectangle construction methods and property settings:

Specify other corner point or [Area/Dimensions/Rotation]:

Specify other corner point – defines the opposite corner (Fig. 2.53, a).

Area – creates a rectangle with a specified area and either a fixed length or width.

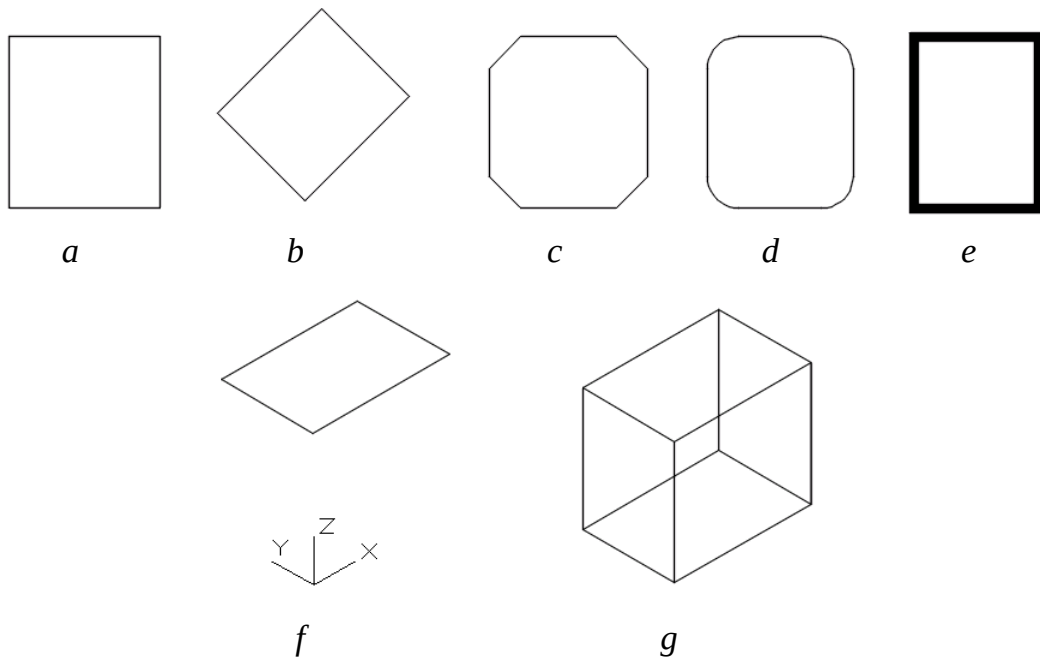


Fig. 2.53. Capabilities of the RECTANGLE command

Dimensions – creates a rectangle with a specified area and either a fixed length or width.

Rotation – sets the rotation angle relative to the first specified corner (Fig. 2.53, b).

2.3.9.7. Polygons

Access to the corresponding command is provided as follows: Ribbon → HOME tab → DRAW panel →  POLYGON command.

The POLYGON command creates a regular polygon with the number of sides ranging from 3 to 1024.

Three construction methods are available: a polygon inscribed in a circle, circumscribed about a circle, or defined by one of its sides.

The following prompts appear in the Command Line:

Command: *POLYGON*

Enter number of sides <>: – specify the number of sides.

Specify center of polygon or [Edge]: – specify the center of the polygon.

Command prompts:

Radius of circle – defines the radius of the inscribed or circumscribed circle.

Options:

Inscribed in circle – constructs a polygon inscribed in a circle (Fig. 2.54, a).

Circumscribed about circle – constructs a polygon circumscribed about a circle (Fig. 2.54, b).

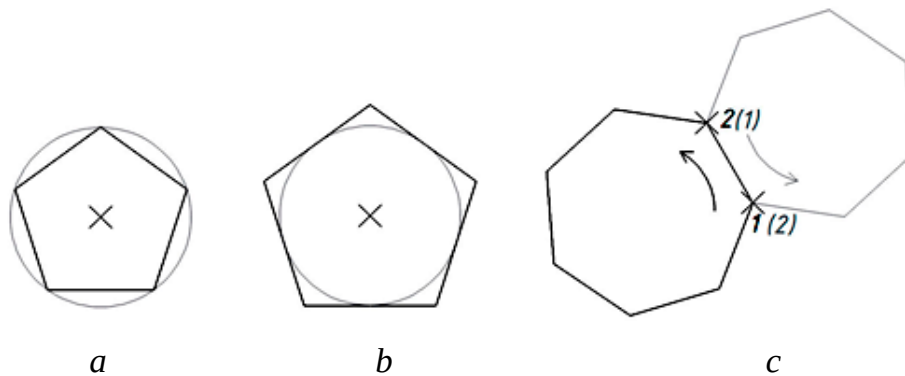


Fig. 2.54. Methods for constructing a polygon

Edge – Edge – creates a polygon by specifying one of its sides through its start and end points. The polygon is constructed counterclockwise (Fig. 2.54, c).

It should be noted that when prompted for the radius of the inscribed or circumscribed circle, a preview of the resulting polygon is also displayed.

3.3.9.8. Donuts

Execution of this operation is performed as follows: Ribbon → HOME tab → DRAW panel →  DONUT command.

This tool creates filled rings with a specified color. The width of the figure equals the difference between the outer and inner diameters, as shown in Fig. 2.55.

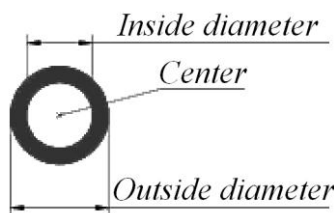


Fig. 2.55. Donut parameters

Command: DONUT

Specify inside diameter of donut <>: – specifies the inner diameter (if set to 0, a solid filled circle is created).

Specify outside diameter of donut <>: – specifies the outer diameter.

Specify center of donut or <exit>: – specifies the center of the donut.

Specify center of donut or <exit>: – optionally specify a new center to create additional donuts with the same parameters, or press Enter to exit the command.

2.3.9.9. Splines

In AutoCAD, splines can be constructed in two ways: through a set of points that the curve must pass through or using control vertices that define the shape of the curve.

The SPLINE FIT command creates a spline using the first method. It is accessed as follows: Ribbon → HOME tab → DRAW panel →  SPLINE FIT command.

The resulting spline passes through the points specified by the user (Fig. 2.56).

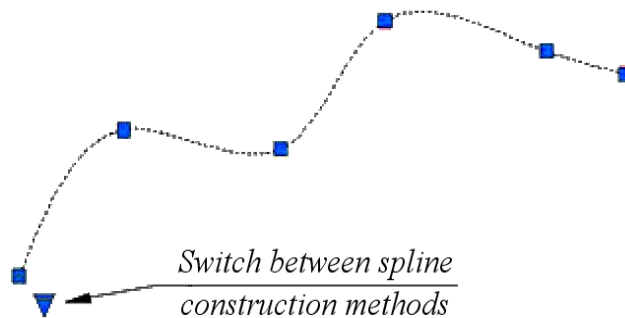


Fig. 2.56. Spline created through fit points

The construction method can be switched to the alternative mode—using control vertices. This may be done by selecting the corresponding option in the Command Line during creation, or by selecting the spline afterward and using the toggle located near the first point.

The command that constructs a spline based on control vertices (Fig. 2.57) is accessed as follows: Ribbon → HOME tab → DRAW panel →  SPLINE CV command.

In this mode, the characteristic control polygon is also displayed.

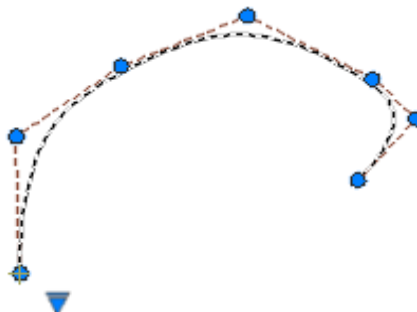


Fig. 2.57. Spline created using control vertices

This method allows the spline to be modified by adjusting the positions of its control vertices.

2.3.9.10. Regions

The REGION command creates a single object from a set of adjoining figures that form a closed contour. These components may include line segments, polylines, arcs, splines, and other objects.

The command is executed as follows: Ribbon → HOME tab → DRAW panel →  REGION command.

After launching the command, the prompt *Select objects* appears, and you must select the objects that form the closed contour (Fig. 2.58).

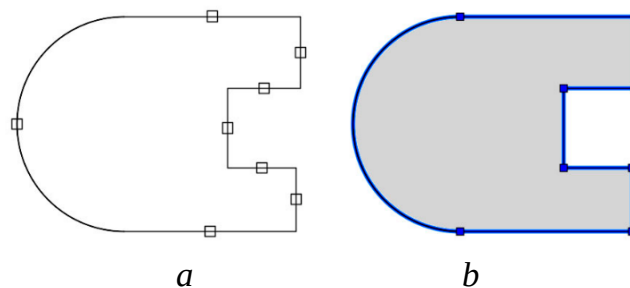


Fig. 2.58. Creating a region:

a – selecting individual objects; b – resulting single region object

Regions support hatching, filling, centroid and area calculation, as well as Boolean operations (Fig. 2.59): Union, Subtract, and Intersect.

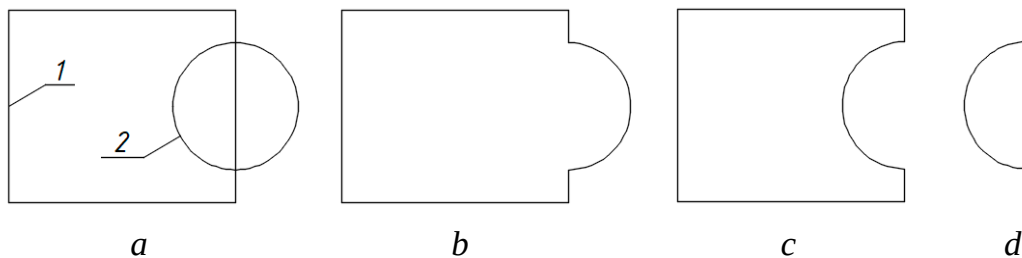


Fig. 2.59. Boolean operations on regions:

a – source shapes; b – union; c – subtraction; d – intersection

These tools significantly enhance the efficiency of geometric modeling.

2.3.10. Editing Objects

AutoCAD allows objects to be edited by modifying their position, size, and shape. There are two fundamental approaches to editing:

- First select the objects and then apply the editing operation.
- Start the desired editing command and then select the objects to be modified.

When selecting objects, all standard selection techniques may be used (see Section 2.3.7).

2.3.10.1. Moving Objects

The MOVE command shifts one or several objects without changing their orientation (Fig. 2.60). It is accessed as follows: Ribbon → HOME tab → MODIFY panel →  MOVE command.

Command: MOVE

Select objects: – select the objects.

Select objects: – press Enter to complete the selection.

Specify base point or [Displacement] <Displacement>: – specify the base point; using the Displacement option sets the origin as the base point.

Specify second point or <use first point as displacement>: – specify the second point to complete the displacement.

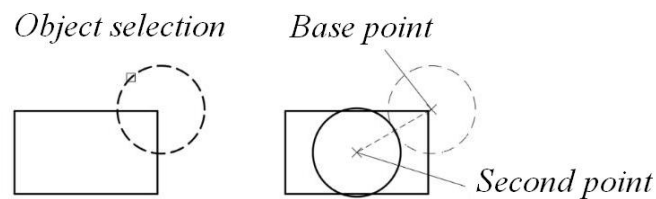


Fig. 2.60. Moving an object

Objects can also be repositioned using grips. This method is described in Section 2.3.10.7.

2.3.10.2. Rotating Objects

The ROTATE command rotates one or several objects around a specified point. Rotation may be performed by a relative angle (the objects rotate from their current orientation) or by an absolute angle (the current rotation angle is set to the specified value).

The operation is executed as follows: Ribbon → HOME tab → MODIFY panel →  ROTATE command.

Command: ROTATE

Current positive angle in UCS: ANGDIR=counterclockwise ANGBASE=0 – message indicating current settings.

Select objects: – select the objects.

Select objects: – press Enter to finish selection.

Specify base point: – specify the base point.

Specify rotation angle or [Copy/Reference] <>: – enter the rotation angle or choose an option.

Options:

Copy

Rotating a copy of the selected objects – rotates a copy of the selected objects.

Specify rotation angle or [Copy/Reference] <>: – enter the rotation angle for the copied geometry relative to the selected base point.

Reference

This option is useful when the absolute value of the rotation angle is unknown. An example is shown in Fig. 2.61.

First, select the objects and specify the base point (point 1 in Fig. 2.61, a). Then, after choosing the Reference option, define the reference angle.

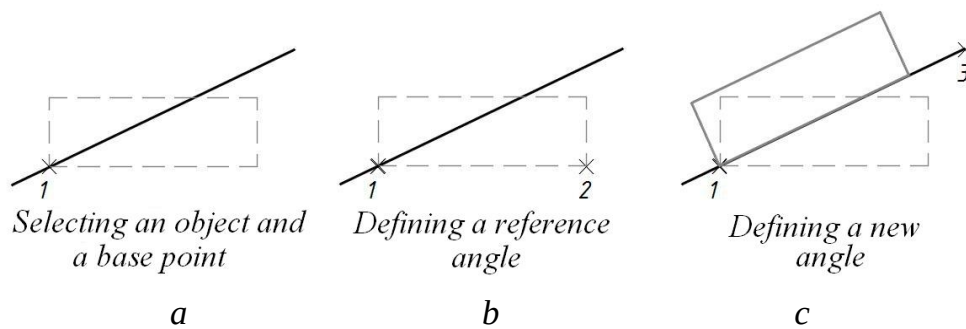


Fig. 2.61. Rotating an object using a reference angle

If the reference angle is unknown, two points on the screen may be specified (points 1 and 2 in Fig. 2.61b) to define the angle geometrically.

When prompted for the new angle, enter its numeric value or specify a point (point 3 in Fig. 2.61c).

2.3.10.3. Mirror

The MIRROR command creates a mirrored copy of selected objects relative to a specified axis of symmetry. The original objects may either be retained or deleted.

Access to the command is provided as follows: Ribbon → HOME tab → MODIFY panel →  MIRROR command.

After selecting the objects with a selection window (points 1–2 in Fig. 2.62), the system prompts for the first and second points (points 3 and 4) that define the mirror axis. Next, AutoCAD asks whether the original objects should be deleted.

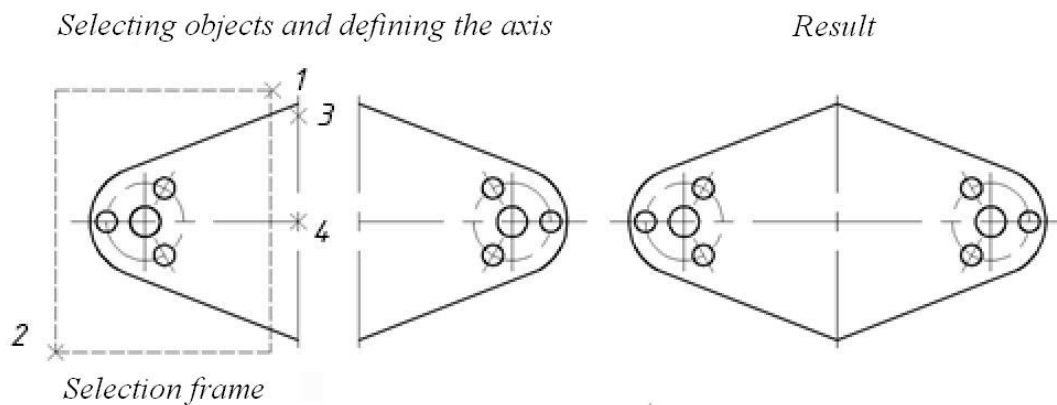


Fig. 2.62. Mirroring objects

To delete the original objects, choose Yes; otherwise, choose No.

2.3.10.4. Copying

The COPY command creates one or multiple copies of selected objects while preserving their properties and orientation. It is accessed as follows: Ribbon → HOME tab → MODIFY panel →  COPY command.

Command: COPY

Select objects: – select the objects.

Select objects: – the prompt repeats until Enter is pressed.

Current settings: Copy mode = Multiple – message indicating the current mode.

Specify base point or [Displacement/mOde] <Displacement>: – specify the base point or choose an option.

Options:

Displacement – sets the base point at the origin.

mOde – switches between single and multiple copy modes.

Enter a copy mode option [Single/Multiple] <Multiple>:

Single – creates one copy.

Multiple – creates multiple copies.

It is commonly used to replicate repeating elements efficiently, maintain consistent spacing, or generate construction patterns without redrawing individual components. An example of using the COPY command is shown in Fig. 2.63.

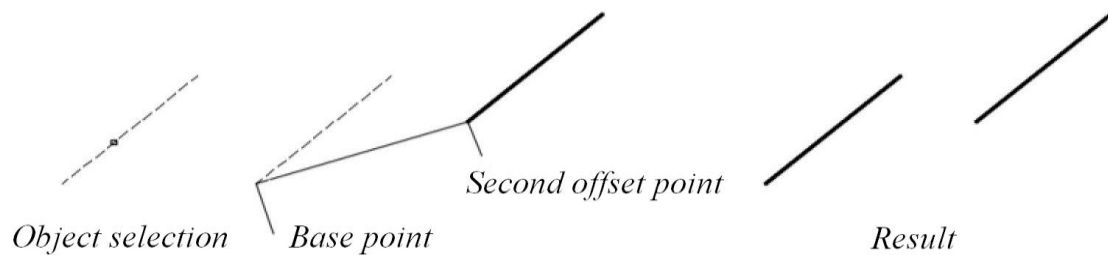


Fig. 2.63. Copying objects

If the Array option is selected (Fig. 2.64), prompts appear for constructing an array of copies:

Specify base point or [Displacement/mOde/Multiple] <Displacement>: – specify the base point.

Specify second point or [Array] <use first point as displacement>: – choose the Array option.

Enter number of items to array: – specify the total number of copies, including the original.

Specify second point or [Fit]: – specify the spacing and direction between copies, or choose an option..



Fig. 2.64. Creating an array of copies

Fit option:

Specify second point or [Array]: – specify the second point; AutoCAD places the chosen number of copies evenly between the base point and the second point.

Specify second point or [Array/Exit/Undo] <Exit>: – press Enter to finish the command.

2.3.10.5. Deleting Objects

The ERASE command is used to remove one or several objects from the drawing. It is executed as follows: Ribbon → HOME tab → MODIFY panel →  ERASE command.

Command: ERASE

Select objects: – select the objects.

Select objects: – press Enter to complete the selection.

Objects may also be deleted using the Delete key, provided the required objects are selected beforehand.

This command is one of the most frequently used editing tools, allowing users to quickly remove unnecessary geometry and maintain a clean, organized drawing environment.

2.3.10.6. *Scaling*

The SCALE command is used to resize objects proportionally and is accessed as follows: Ribbon → HOME tab → MODIFY panel →  SCALE command.

After selecting the required objects, the system prompts for a base point. Then, either a scale factor (Fig. 2.65a) is entered, or one of the following options is chosen: Copy – creates a scaled copy of the original object (Fig. 2.65, b); Reference – scales objects based on the length of a reference segment (Fig. 2.65, c).

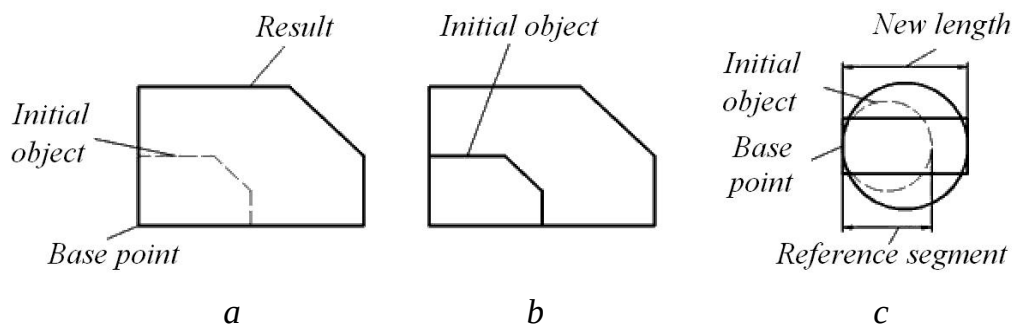


Fig. 2.65. Scaling objects

The scale factor defines the multiplier by which the selected geometry will be enlarged or reduced.

This command is essential when adjusting object proportions, preparing drawings for layout, or matching the size of imported geometry to the current project.

2.3.10.7. *Grips*

Grips are key reference elements that determine the position, size, and shape of geometric objects. They appear as small square markers – typically blue – when objects are selected while no command is active (Fig. 2.66).

Using grips enables quick, intuitive editing without launching separate commands, allowing the user to move, rotate, copy, scale, stretch, or otherwise modify geometry directly through interactive manipulation.

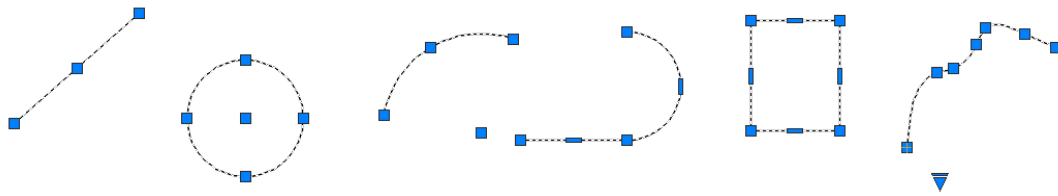


Fig. 2.66. Grips for different objects

The available actions can be accessed through the shortcut menu that appears when a grip is activated; an example of this menu is shown in Fig. 2.67.

Because grips provide immediate visual feedback and require minimal command input, they significantly speed up the editing process and offer a highly efficient way to refine or adjust model geometry.

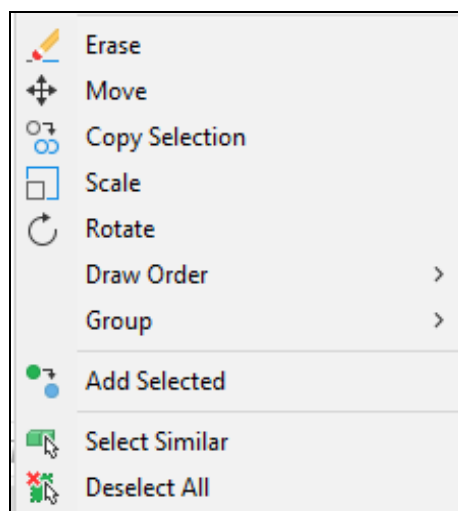


Fig. 2.67. Context menu for editing with grips

2.3.10.8. Creating Similar Objects

The OFFSET command creates a new object that is geometrically similar to an existing one. It is accessed as follows: Ribbon → HOME tab → MODIFY panel →  OFFSET command.

Command: OFFSET

Specify offset distance or [Through/Erase/Layer] <0.0000>: – specify the offset distance. If the Through option is selected, the new object will pass through a point specified on the screen (Fig. 2.68, a).

Erase – deletes the original object after generating the offset copy (Fig. 2.68, b).

Layer – allows choosing the layer on which the offset object will be created. The *Current* option places the new object on the current layer (Fig. 2.68c), while *Source* places it on the source object's layer.

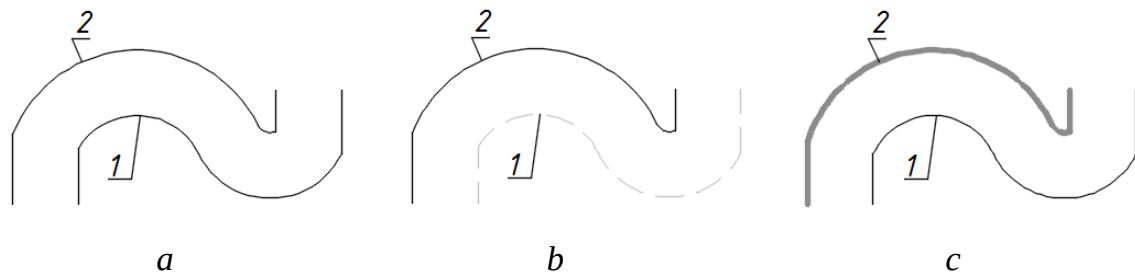


Fig. 2.68. Options for locating offset objects:
1 – source object; 2 – offset object

Select object to offset or [Exit/Undo] <Exit>: – select the source object;

Specify point on side to offset or [Exit/Multiple/Undo] <Exit>: – specify the side to which the object is offset.

The Multiple option allows repeated creation of offset objects without reselecting the original geometry.

This command is especially useful for generating parallel lines, concentric curves, and layered geometric structures, making it indispensable for drafting architectural, mechanical, and engineering drawings.

2.3.10.9. Creating Chamfers

The CHAMFER command constructs chamfers defined by either:

- The lengths of the two offset distances (legs).
- A linear distance and an angle.

This tool is used to replace sharp corners with beveled edges, improving both the geometric clarity and manufacturability of a design. Chamfers are widely applied in mechanical drafting, architectural layouts, and any situation where angled transitions between lines are required.

The command is executed as follows: Ribbon → HOME tab → MODIFY panel →  CHAMFER command.

Command: CHAMFER

Select first line or [Undo/Polyline/Distance/Angle/Trim/mEthod/Multiple]: – select the first object or choose an option.

Options:

Trim – trims the intersecting objects to accommodate the chamfer.

No Trim – leaves the objects unchanged.

Current chamfer Dist1 = 0.0000, Dist2 = 0.0000 – when Trim mode is enabled, the external portions of the lines are removed, as shown in Fig. 2.69.

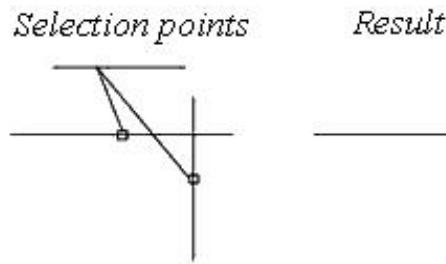


Fig. 2.69. Trimming excess segments when both offsets are 0

Distance – specifies the two offset distances that define the chamfer (Fig. 2.70).

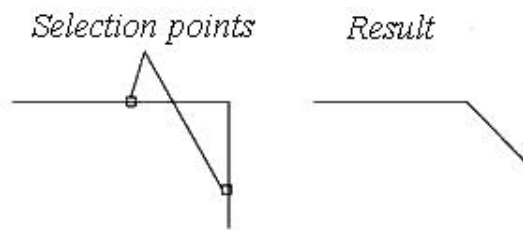


Fig. 2.70. Trimming excess segments when offsets > 0 in Trim mode

Angle – used when the chamfer must be defined by a known first offset distance and an angle relative to the first selected line.

mEthod – switches between the Distance and Angle input methods.

Multiple – allows multiple chamfers to be created during a single command session without restarting the command.

2.3.10.10. Filleting

The FILLET command creates a rounded transition (fillet) between two objects using an arc of a specified radius.

It is accessed as follows: Ribbon → HOME tab → MODIFY panel →  FILLET command.

Command: FILLET

Current settings: Mode = TRIM, Radius = 0.0000

Select first object or [Undo/Polyline/Radius/Trim/Multiple]: – select the first object or choose an option.

Options:

Radius – specifies the radius of the fillet arc.

Trim – trims the selected objects to their intersection point before constructing the fillet. Proper use of this option is shown in Fig. 2.71. If the object is closed (e.g., a circle), trimming is not applied.

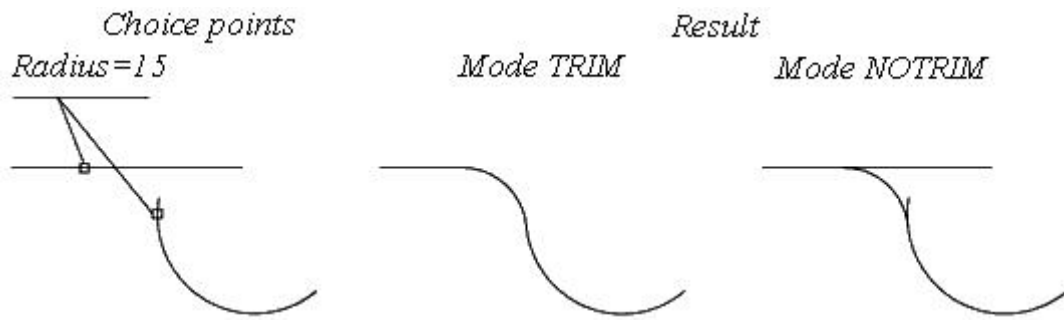


Fig. 2.71. Using the Trim option

Multiple – allows several fillets to be created during a single command session (Fig. 2.72).

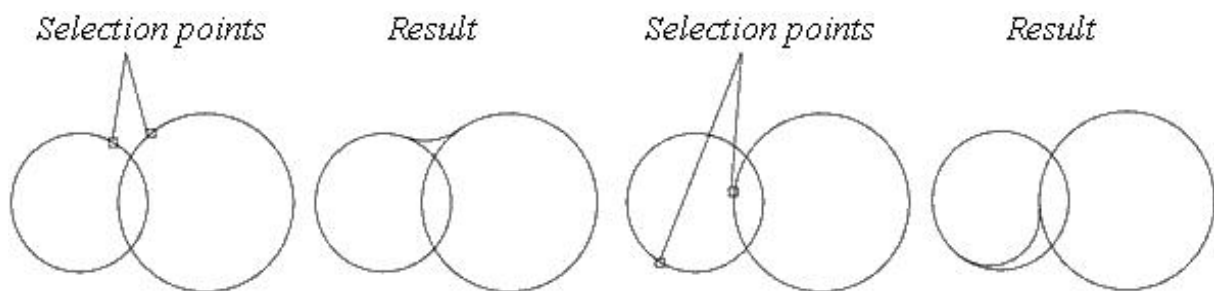


Fig. 2.72. Using the Multiple option

Fillets between arcs and circles can be constructed in different ways. The arc of the fillet is created so that its endpoints lie as close as possible to the points selected on the objects, as illustrated in Fig. 2.72.

This command is widely used to create smooth transitions between segments, eliminate sharp corners, and produce geometrically consistent rounded edges. Fillets are essential in mechanical design, architectural drafting, and any application requiring continuous tangential connections.

3.3.10.11. *Trimming and Extending*

The TRIM and EXTEND commands perform cutting and lengthening operations on geometric objects. They are accessed as follows: Ribbon → HOME tab → MODIFY panel →  TRIM or  EXTEND command.

These commands are among the most frequently used editing tools in AutoCAD, allowing users to clean intersections, adjust geometry to meet design constraints, and quickly refine linework in both 2D drafting and more complex layouts. They greatly enhance precision and productivity when modifying connected or overlapping elements.

Command: TRIM

Current settings: Projection=None, Edge=Extend

Select cutting edges – specify the objects that will act as cutting boundaries.

Select object to trim or shift-select to extend or[Fence/Crossing/Project/Edge/eRase/Undo]: – specify the objects to be trimmed or choose an option.

To extend objects without leaving the TRIM command, hold Shift while selecting the geometry to be lengthened (Fig. 2.73).

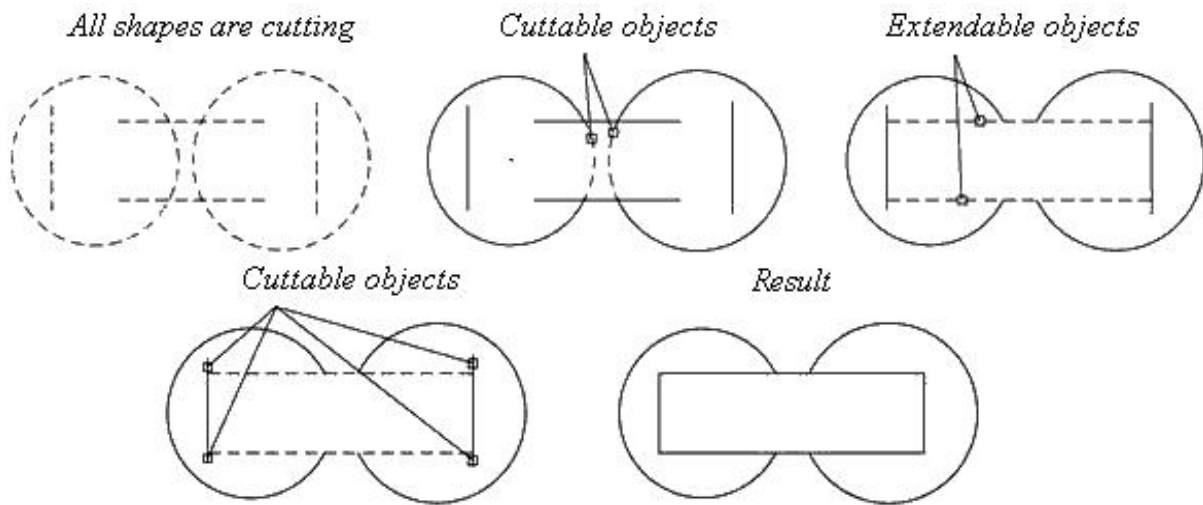


Fig. 2.73. Examples of Trim and Extend operations

Options:

Fence/Crossing – allow trimming objects by using a temporary polyline (Fence) or a crossing selection window.

Edge – enables or disables extension to imaginary intersections.

eRase – deletes selected objects without interrupting the trimming process.

The EXTEND command includes the same options as TRIM. When selecting boundary edges with Shift, the command switches temporarily into trimming mode.

3.3.10.12. Breaking Objects

The BREAK command allows a portion of an object to be removed by specifying the first and second break points. It is accessed as follows: Ribbon → HOME tab → MODIFY panel →  BREAK command.

If it is necessary to split an object into two parts without deleting any segment, the  BREAK AT POINT command is used. In this case, the user selects the object first and then specifies the break point.

Both tools provide precise control over segmenting geometry, making it easier to adjust linework, clean intersections, or prepare objects for further modification.

2.3.10.13. *Stretching*

The STRETCH command is used to stretch a geometric object by moving a portion of it. The operation is performed as follows: Ribbon → HOME tab → MODIFY panel →  STRETCH command.

After the command is launched, the system prompts for object selection, which must be carried out using a crossing window. It is important to note that only the objects partially enclosed by this window will be stretched. Objects fully enclosed by the window are moved without deformation, and the positions of endpoints and vertices outside the window remain unchanged (Fig. 2.74).

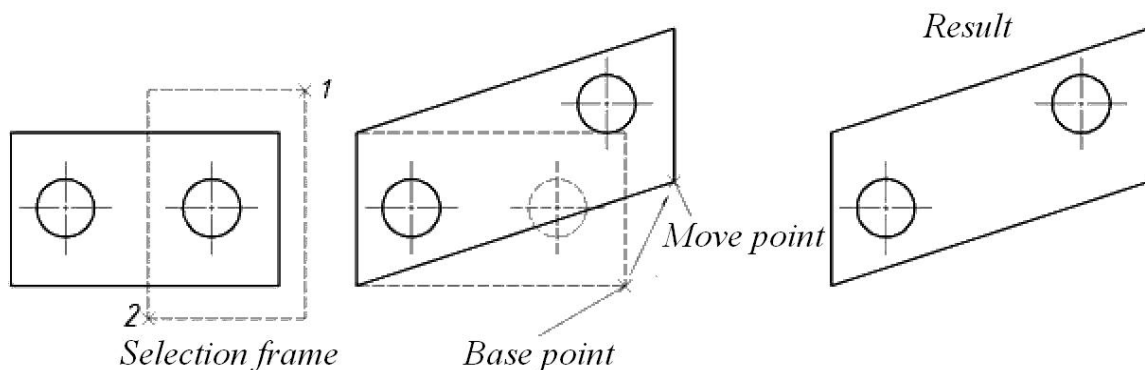


Fig. 2.74. Stretching objects

It should be noted that circles and ellipses cannot be stretched using this command. The STRETCH command is particularly effective for modifying linear or polyline-based geometry, adjusting proportions, and reshaping structural elements while preserving connections to the rest of the drawing.

3.3.10.14. *Arrays of Geometric Objects*

The ARRAY command creates multiple copies of selected objects arranged in a structured pattern. After choosing the source objects, the user must specify how these copies will be positioned.

Arrays may be:

Associative (Fig. 2.75, a) – their properties, such as the number of items, spacing, and source-object modifications, remain dynamically editable.

Non-associative (Fig. 2.75, b) – all items are independent; modifying one element does not affect the others.

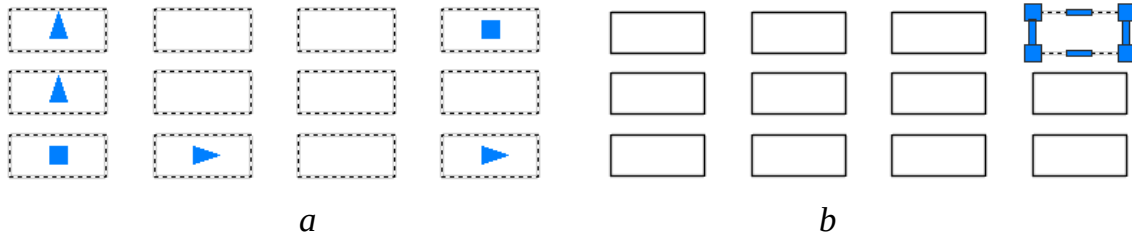


Fig. 2.75. Associative and non-associative arrays

The command is accessed as follows: Ribbon → HOME tab → MODIFY panel →  ARRAY command.

Three types of array layouts are available:

 Rectangular ARRAY – distributes items in rows and columns (Fig. 2.75).

 Path ARRAY – arranges items evenly along a specified path (Fig. 2.76).

Options:

Method – selects how the elements are distributed along the path.

Divide – divides the path into equal segments and places the items at the division points (Fig. 2.76, a). If the array is associative, the spacing adjusts automatically when the path length is modified.

Measure – places items along the path with a specified fixed interval (Fig. 2.76, b). If the path becomes too short to accommodate all items, their number is automatically reduced.

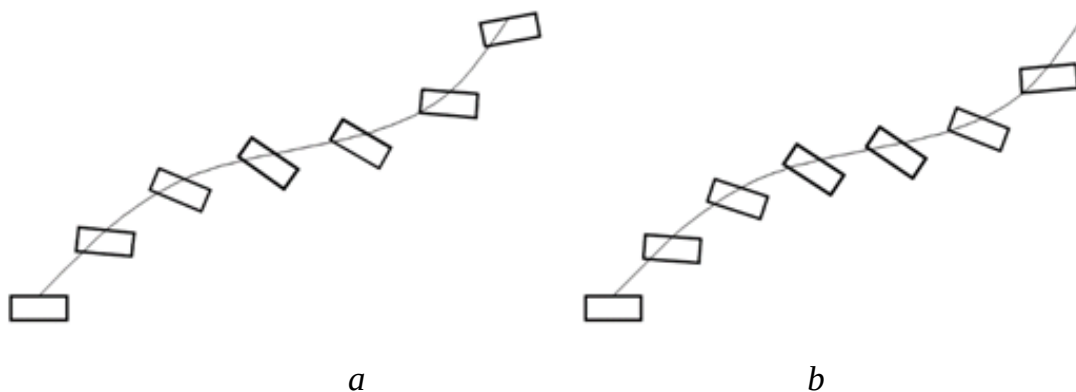


Fig. 2.76. Array adjustment after path modification

 Polar ARRAY – distributes items evenly along an arc or full circle around a defined center (Fig. 2.77).

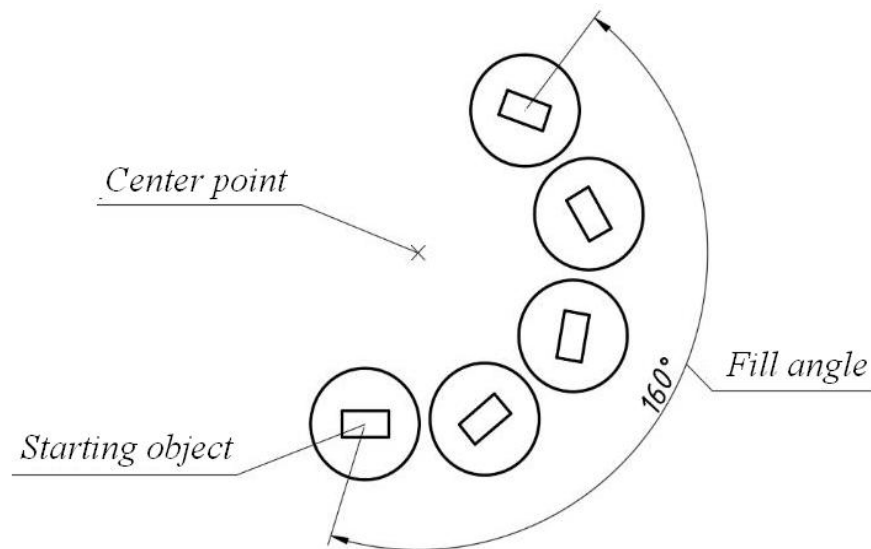


Fig. 2.77. Polar array

By default, the array is constructed counterclockwise. To reverse the direction, a negative fill angle is entered. The Rotate items as copied option enables rotation of individual objects as they are placed in the array.

The ARRAY command is a powerful tool for creating repetitive patterns, structural layouts, and design elements, greatly enhancing efficiency when working with modular or symmetrically arranged geometry.

2.3.11. Text and Annotations

In AutoCAD terminology, text or annotations refer to a group of tools used to add explanatory and supplementary information to a drawing. These include notes, table data, dimensions, callouts with leaders, tolerances, and other annotation elements.

Annotations play a key role in documenting a design, ensuring that the geometry is accompanied by clear technical information required for manufacturing, construction, or further analysis.

2.3.11.1. Text Styles

A text style in AutoCAD is a set of parameters that define the appearance of text elements in a drawing. These parameters include the font, text height, obliquing angle, color, and other visual characteristics. Each text style has its own name and is stored within the drawing file, ensuring consistent formatting across all annotations.

The tool used to manage text styles is the Style command, which opens the Text Style dialog box (Fig. 2.78). Here, users can create, modify, and configure the styles that will be applied to text objects. This command is available both from the command line and from the Annotate tab on the ribbon.

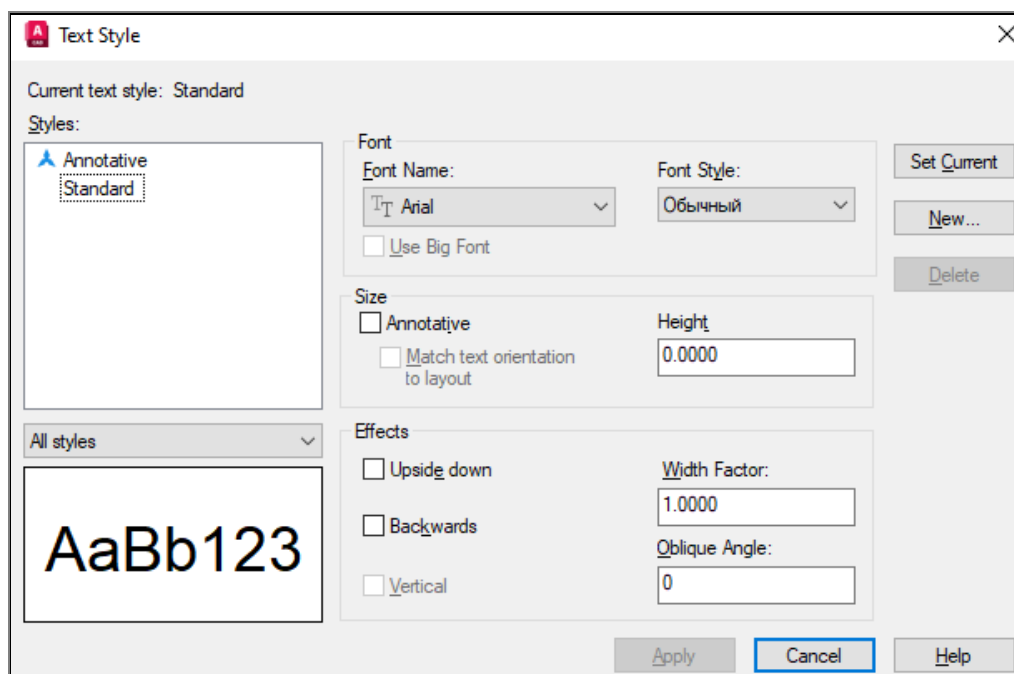


Fig. 2.78. Text Style dialog box

By default, AutoCAD provides two predefined styles: Annotative and Standard. To create a new style with customized settings, the New button is pressed. This opens the New Text Style dialog box, where a style named Style1 is generated automatically. It is advisable to assign it a meaningful name.

The Font section of the dialog contains a drop-down list of all available typefaces under Font Name. For domestic (Ukrainian) technical drawings, the recommended font is Isocpeur. The Font Style list provides variants such as Italic or Regular. The Height field specifies the default text height; if left at zero, the height can be assigned later during text creation.

2.3.11.2. Creating Text

Text objects are created using the following sequence: Ribbon → ANNOTATE tab → TEXT panel → TEXT drop-down list (Fig. 2.79).

Two commands are available:

- Single Line – creates single-line text.
- Multiline Text – creates multi-line text.

The Single Line text command requires specifying the starting point for text placement and defining its parameters, such as style, height, and justification. One or several lines of text may be entered, each separated by pressing Enter. Every line created in this mode becomes an individual object.

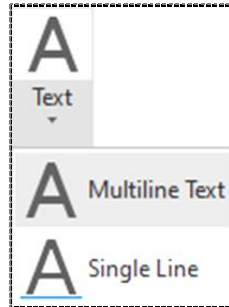


Fig. 2.79. Icons for text creation commands

Command: TEXT

Current text style: "style1" Text height: 2.5000 Annotative: No Justify: Left – displays the current text settings, including the active text style, height, annotative property, and justification mode.

Specify start point of text or [Justify/Style]: – prompts the user to specify the starting point for the text or choose an option to modify the current text parameters.

After defining the first point, the following dialog begins:

Specify height <>: – specifies the required text height or accepts the suggested default value by pressing Enter;

Specify rotation angle of text <>: – defines the rotation angle of the text or accepts the default angle shown in angle brackets by pressing Enter.

When the *Justify* option is selected, the system offers several alignment methods:

Enter an option [Left/Center/Right/Align/Middle/Fit/TL/TC/TR/ML/MC/MR/BL/BC/BR]: – prompts the user to choose the desired justification mode.

The following list provides detailed explanations of the available justification options, each of which specifies how the insertion point relates to the position of the text within its bounding box:

Left – aligns the text so that its starting point corresponds to the leftmost position of the baseline.

Center – places the midpoint of the text at the specified insertion point. *Right* – aligns the end of the text with the chosen point, positioning the text to the left of it.

Align – stretches or compresses the text so that it fits exactly between two specified points.

Middle – centers the geometric midpoint of the text at the indicated location.

Fit – adjusts character spacing so the entire text fits precisely between two points without changing text height.

TL, TC, TR – position the text so that the insertion point corresponds to the top-left, top-center, or top-right of the bounding box.

ML, MC, MR – align the insertion point with the middle-left, middle-center, or middle-right of the bounding box.

BL, BC, BR – align the insertion point to the bottom-left, bottom-center, or bottom-right corner of the bounding box.

The Multiline Text command places text within a rectangular area defined on the screen by selecting two opposite corners. In this mode, all lines of text are treated as a single unified object. The formatting capabilities for multiline text are significantly broader than those available for single-line text. The command-line dialogue proceeds as follows:

Command: MTEXT

Current text style: "style1" Text height: 2.5 Annotative: No – displays the current text parameters.

Specify first corner: – prompts the user to define the first corner of the rectangular text boundary.

Specify opposite corner or [Height/Justify/Linespacing/Rotation/Style/Width/Columns]: – specifies the opposite corner of the text field or allows choosing one of the available options:

Height – defines the text height;

Justify – sets the alignment method;

Line spacing – adjusts the spacing between lines;

Rotation – specifies the rotation angle of the text;

Style – selects the active text style;

Width – determines the width of the text boundary;

Columns – enables the creation of text arranged in multiple columns.

After completing the necessary settings, pressing Enter activates the MTEXT editor, where the required text is entered. To finish the process, click the left mouse button outside the text boundary.

2.3.11.3. Editing Text

To modify multiline text, the MTEDIT command is used, which provides access on the Ribbon to the tools of the Text Editor, as shown in Fig. 2.80. The same editing mode can be activated by double-clicking the left mouse button on the selected text.

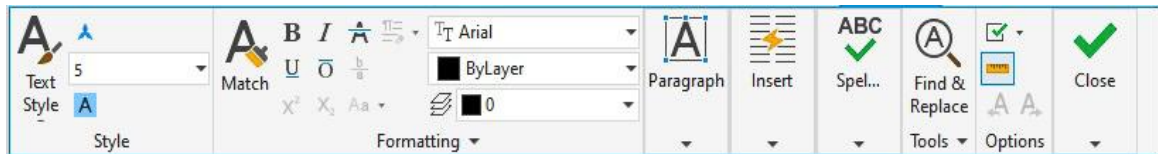


Fig. 2.80. Text Editor toolbar

The available tools allow changing the text style, adjusting its height, copying properties, formatting paragraphs, and applying other text modifications. These editing actions performed with the left mouse button can also be applied to single-line text objects.

2.3.11.4. Dimension Styles

In AutoCAD, the appearance of dimensions can be customized in the Dimensions Style Manager window (see Fig. 2.81), which is opened using the DIMSTYLE command or through the ribbon. Pressing the New button opens the Create New Dimension Style dialog box (Fig. 2.82). In the Start With list, the user must select an existing base dimension style, and in the New Style Name field, enter the name of the new style. The Continue button then opens the New Dimension Style window (Fig. 2.83).

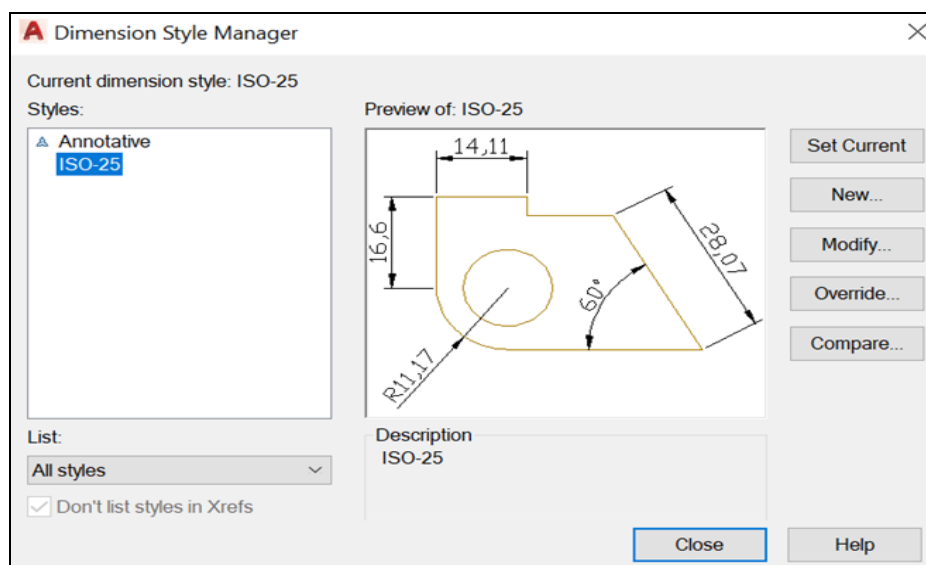


Fig. 2.81. Dimensions Style Manager dialog box

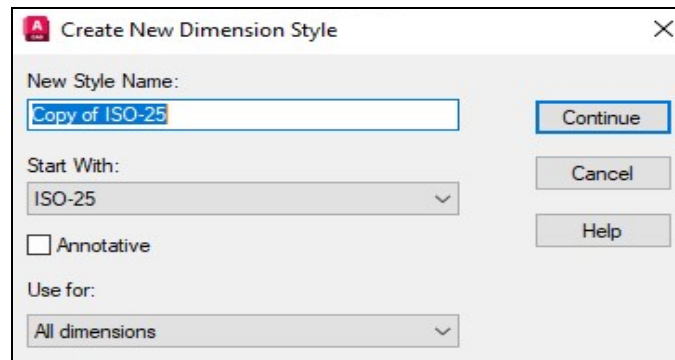


Fig. 2.82. Create New Dimension Style window

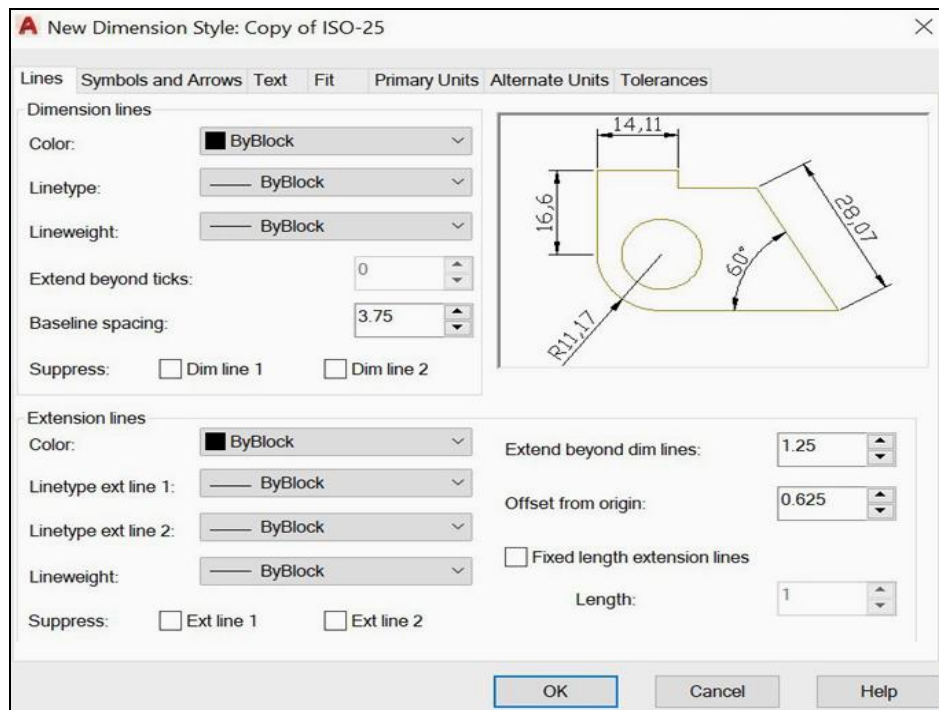


Fig. 2.83. New Dimension Style window

This window contains seven tabs, each allowing users to modify different dimension parameters. The preview area reflects all changes applied during customization. The tabs include:

- Lines – specifies the properties of dimension lines and extension lines.
- Symbols and Arrows – defines arrowheads and related symbols.
- Text – controls formatting of the dimension text.
- Fit – adjusts the appearance of dimensions when the standard placement is not possible.
- Primary Units – specifies the format and precision of primary linear and angular units.
- Alternate Units – sets parameters for alternative measurement units.
- Tolerances – defines the method of displaying tolerances.

3.3.11.5. Dimensioning

Each type of dimension – linear, angular, radius, diameter, and others – has its own dedicated command.

For all linear dimension commands, the workflow begins with specifying the start point of the extension line, followed by the location of the dimension line. For radius and diameter dimensions, a circle or arc must be selected, and for angular dimensions, the sides of the angle are indicated. It is recommended to enable object snaps for higher accuracy.

Access to dimensioning tools is provided as follows: Ribbon → HOME tab → ANNOTATION panel → dimensioning command menu (Fig. 2.84).

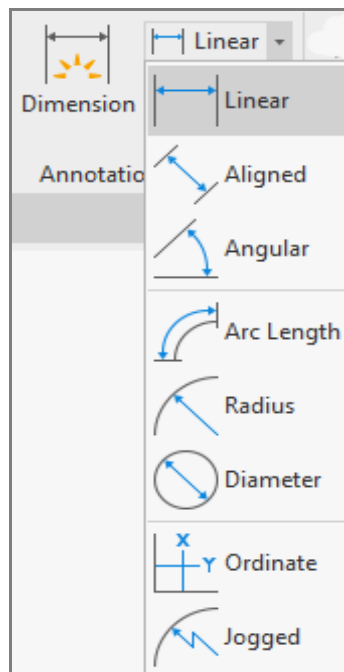


Fig. 2.84. Dimensioning tools

-  Linear – creates horizontal and vertical linear dimensions.
-  Aligned – produces a linear dimension aligned parallel to the referenced segment.
-  Angular – constructs angular dimensions.
-  Arc Length – dimensions the length of an arc.
-  Radius – creates a radius dimension.
-  Diameter – creates a diameter dimension.
-  Jogged – applies a jogged radius dimension when the true center of the arc lies off the drawing.

2.3.11.6. Editing Dimensions

Editing dimensions is an essential part of preparing a clear, accurate, and well-structured technical drawing. AutoCAD provides several tools that allow users to refine the appearance, placement, and content of dimension objects, ensuring compliance with drafting standards and design intent. Depending on the task, dimensions may be adjusted interactively or through dedicated editing commands.

Dimensions in AutoCAD can be modified in several ways. Some of the most commonly used methods are described below.

Editing dimensions using grips is the most convenient approach, as it allows the user to reposition extension lines and dimension lines, as well as adjust the placement of the dimension text. The grips that appear at characteristic points of a dimension object are arranged according to its type (Fig. 2.85).

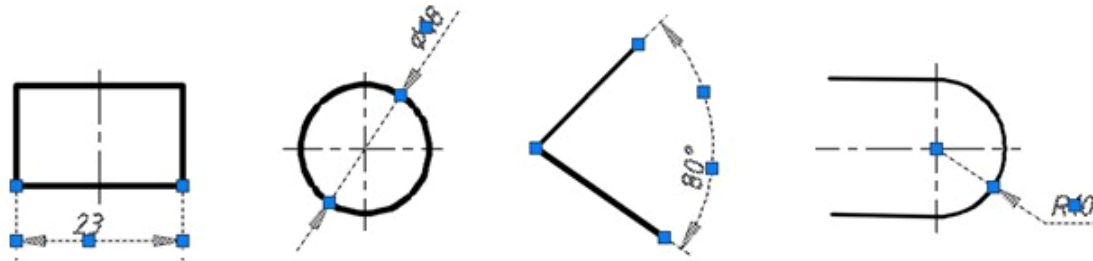


Fig. 2.85. Grips for editing dimensions

To open the text editor for a dimension, double-click the dimension text with the left mouse button. For example, a user may need to specify the number of holes, as illustrated in Fig. 2.86.

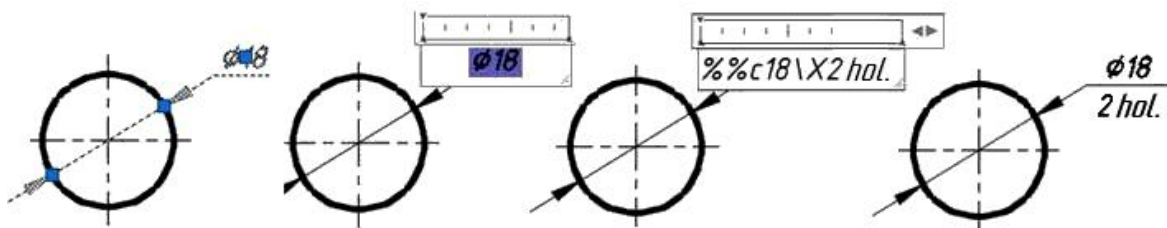


Fig. 2.86. Editing dimension text

The diameter symbol is entered using the code `%%c` when the keyboard is set to the English layout. The notation for the number of holes is created using the sequence `\X`. Editing dimensions with the DIMED (DIMEDIT) command provides additional capabilities, including modifying the text content, rotating the text, and creating extension lines at an oblique angle to the measured feature (Fig. 2.87).

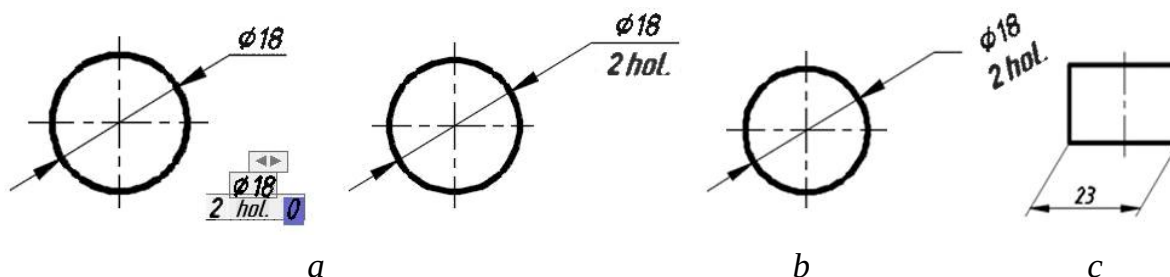


Fig. 2.87. Dimension editing options of the DIMEDIT command:

a – New; *b* – Rotate; *c* – Oblique

Command: DIMED

Enter type of dimension editing [Home/New/Rotate/Oblique] <Home>: – specifies the required editing option and allows its configuration.

Select objects: – selects the dimension object to be modified.

Home – cancels the editing operation and restores the original appearance of the dimension.

New – replaces the existing dimension text with new text; the new text is created in an empty area of the drawing and is applied to the selected dimension once it is chosen.

Rotate – rotates the dimension text by a specified angle.

Oblique – creates extension lines at a user-defined oblique angle relative to the measured segment.

Using the Properties palette is also a convenient way to edit dimensions, providing access to all key dimension parameters in a structured and efficient manner.

2.3.12. Drawing Template Files

Appendix A provides the main requirements of the existing standards for technical drawing preparation. These requirements relate to sheet sizes, internal borders, the content of title blocks, line types, text specifications, scales, and other formatting rules. To ensure convenient and efficient drafting in AutoCAD, it is advisable to use drawing template files. Such templates may include predefined layers, dimension and text styles, a frame of a selected format, a standard title block, and any other elements prepared in advance. Corresponding examples are illustrated in Fig. A.2 of the appendices.

2.3.13. Examples of Constructing Part Contours

The following examples are intended to help develop skills in analyzing the geometric shapes of technical objects and applying practical methods of two-dimensional geometric modeling in AutoCAD. Such analysis involves determining how individual geometric primitives are positioned relative to one another and forming an appropriate sequence of construction steps.

It is recommended to perform the work in a pre-prepared template file containing a title block and border. Typically, the process begins with identifying symmetry axes. Based on them, the necessary elements of the modeled object are constructed. Note that the outlines of many technical parts consist of lines that smoothly transition into one another; these transitions are known as fillets. Mastering fillet construction is essential for accurate modeling. The contour construction is completed by adding dimensions.

2.3.13.1. Support Bracket

The purpose of this example (see Fig. 2.88) is to illustrate the creation of fillets within part contours using AutoCAD tools.

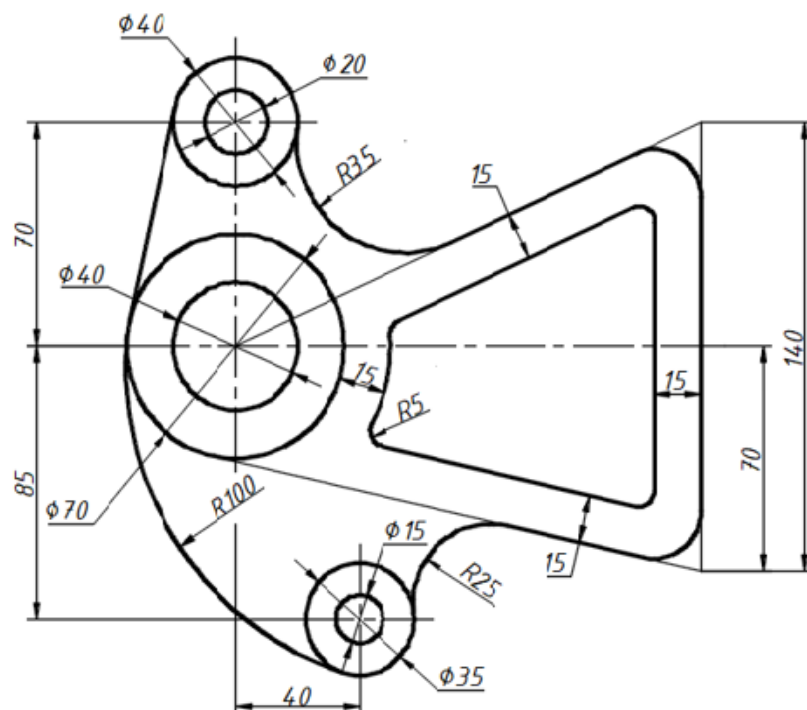


Fig. 2.88. Support bracket

Assignments for independent practice are provided in Appendix B.1.

Construction Sequence:

1. Before starting the work, analyze the geometry of the part, identify its basic shapes, and determine the graphic operations required to construct them.

2. Open the appropriate template file. Distribute the necessary drawing elements – such as axes, contour lines, and dimensions – across the corresponding layers. In this example, the template contains pre-created layers:

- *kontyr* – or the part contour (lineweight 0.5);
- *osi* – axes (default lineweight, linetype Center2);
- *dim* – dimensions;
- *dopom* – auxiliary constructions.

The template also includes predefined dimension and text styles, as well as the title block placed in the Layout space. Save the drawing under the name “*Fillets*”.

3. Begin the construction with the axes of the circles. Then create the left portion of the contour by performing the following steps.

3.1. Set *osi* as the current layer. Enable the ORTHO mode to ensure convenient creation of horizontal and vertical lines.

3.2 Using the XLINE command, draw the centerlines for the circle of diameter 70 (see Fig. 2.89). Select the Offset option and apply the required distances to construct the additional centerlines marked as 3, 4, and 5 in the figure.

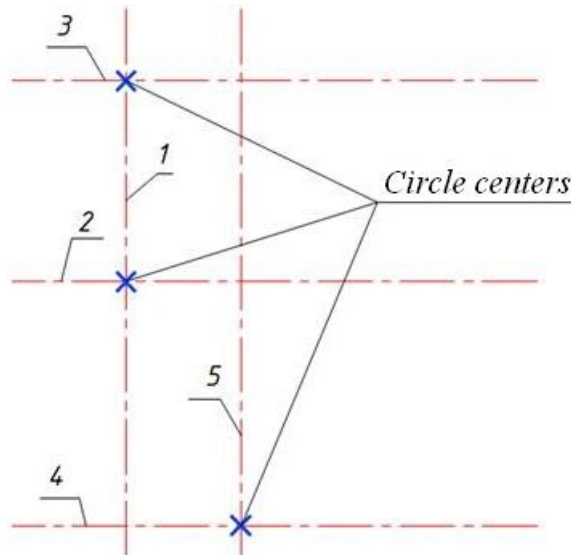


Fig. 2.89. Constructing the centers of the circles

3.3. On the *kontyr* layer, draw the circles (Fig. 2.90) with the required diameters, positioning their centers at the intersection points of the corresponding axes.

3.4. Use the Trim command to edit the existing axes (Fig. 2.91).

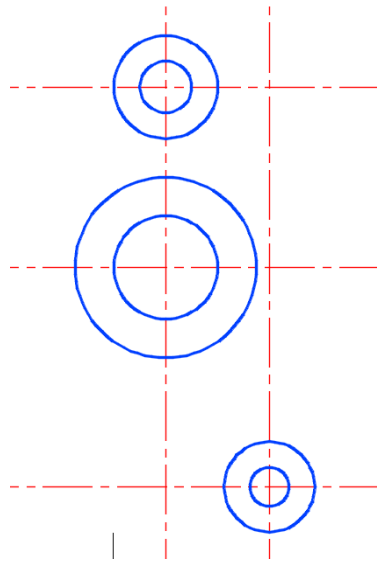


Fig. 2.90. Constructing the circles

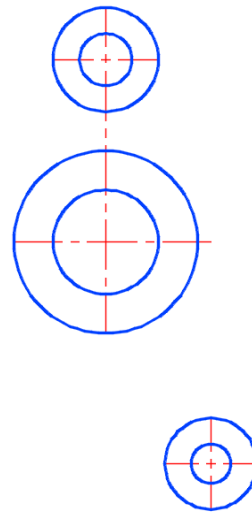


Fig. 2.91. Trimming the axes

3.5. Create the fillets shown in Fig. 2.92 between circle 1 and circle 2 using the LINE command. To do this, specify a point on circle 1 using the Tangent object snap in the expected fillet region, and then specify a tangent point on circle 2.

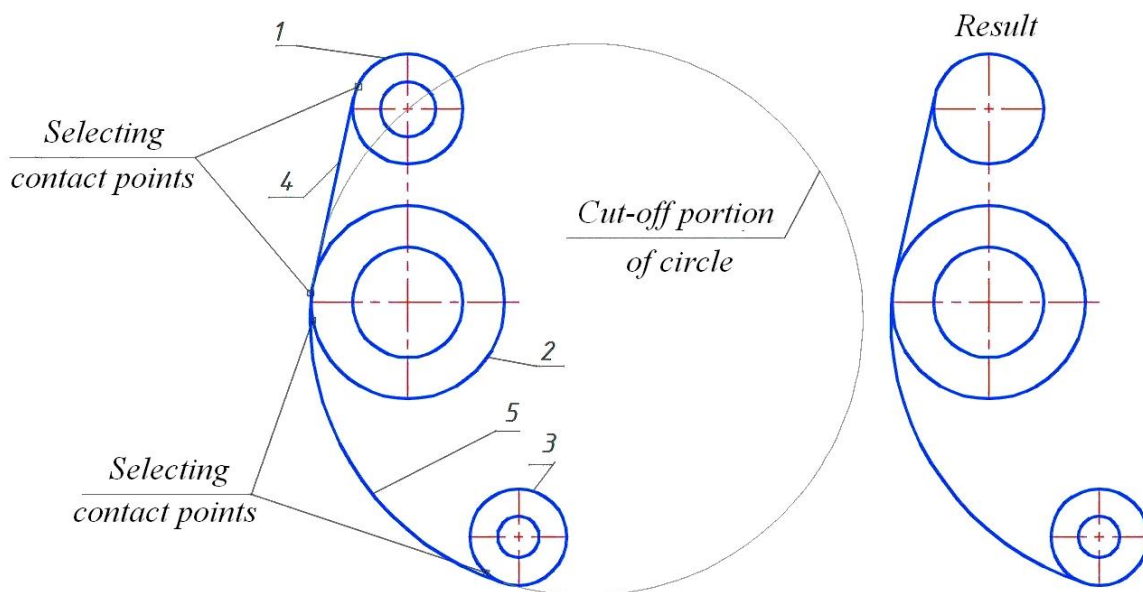


Fig. 2.92. Constructing fillets in the left part of the part

3.6. To construct the fillet arc with radius R100, use the CIRCLE command with the Ttr (tan, tan, radius) option. First specify the tangent point on object 2, then on object 3, and finally enter the radius value 100. Use the Trim command to remove the unnecessary fragment of circle 5 (Fig. 2.92).

4. Construction of the right part of the bracket begins with defining the outer contour.

4.1. Use the Offset option of the XLINE command to determine the overall length dimension. Set the offset distance to 150 and select axis 1 (see Fig. 2.93, a). This will create the infinite line 2. Next, construct the upper line 3 with the starting point 4 (the intersection of the axes) and the endpoint 5, which should be defined using the From object snap: from @0,70 relative to point 6 (the intersection of the horizontal axis and vertical line 2).

4.2. Line 9 is tangent to circle 8 and passes through point 7, which is located at an offset of -70 along the Y-axis relative to point 6. To specify this point, use the From snap: from @0,-70.

4.3. Construct the external fillets using the FILLET command.

When the Trim option is enabled, segments beyond the trimming edges are removed. To keep these segments (see Fig. 2.93, b), choose the No Trim option.

Create the following fillets: R35 between circle 1 and line 2; R25 between circle 5 and line 4; R15 between lines 4 and 3, and between lines 2 and 3.

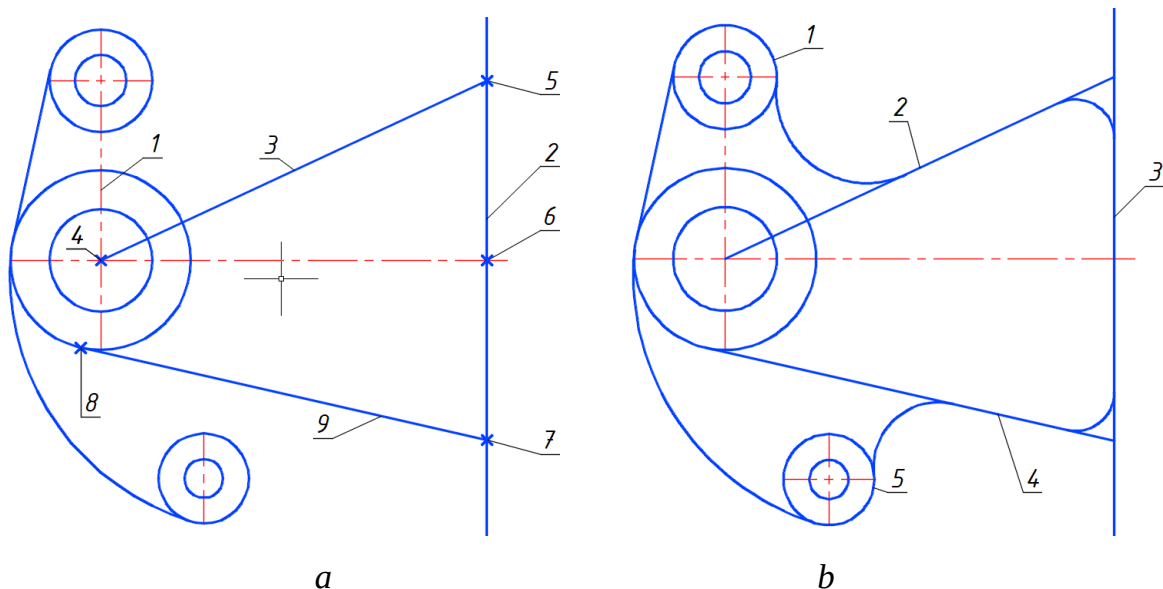


Fig. 2.93. Constructing the right-hand side of the bracket

4.4. Move portions of lines 2, 3, and 4 (see Fig. 2.92, b) to a separate auxiliary layer. To split these elements, use the BREAK AT POINT command. The resulting geometry is shown in Fig. 2.94.

5. Construct the inner contour of the part using the OFFSET command. Set the offset distance to 15, then specify the object and the direction of offset.

For clarity, change the color of the newly created objects (Fig. 2.95). Refine the contour using the FILLET command with the Trim and Multiple options enabled, setting the fillet radius to 5. Modify the axes using the LENGTHEN command with the Delta option, assigning a value of 3.

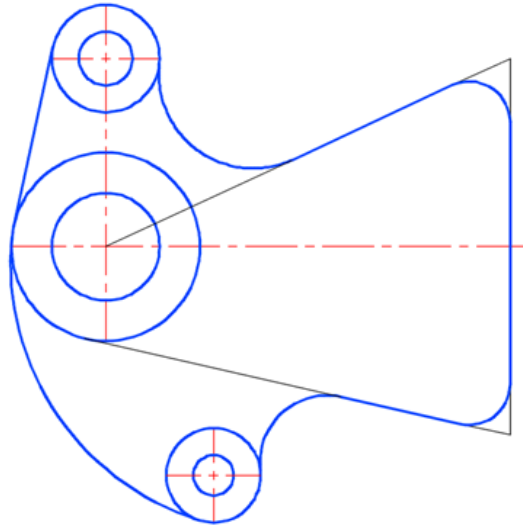


Fig. 2.94. Completed fillets

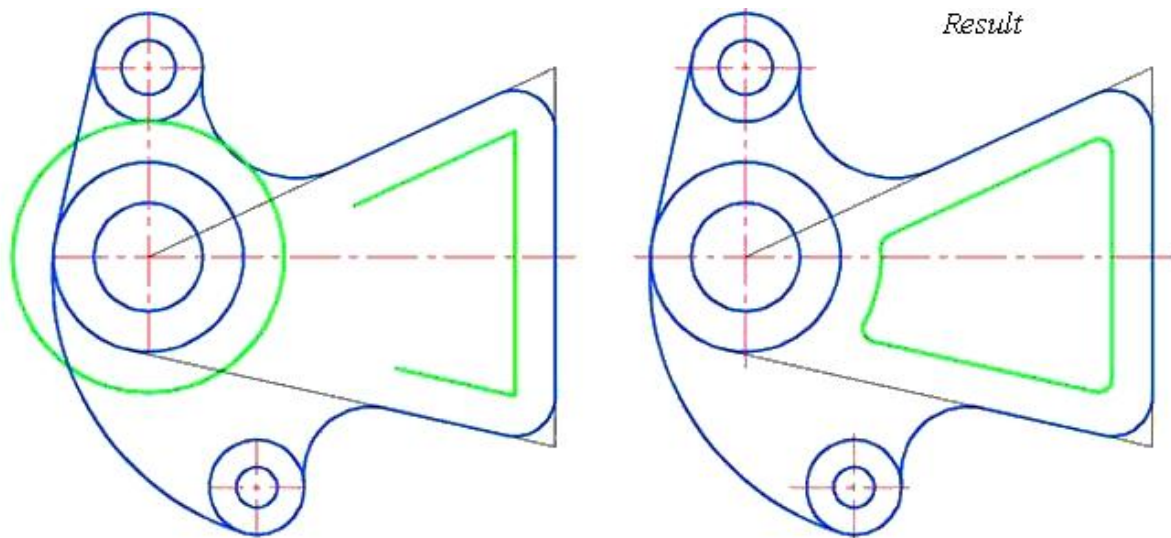


Fig. 2.95. Creating the inner contour

6. Apply the dimensions (see Fig. 2.96), selecting *dim* as the current layer.

7. Finalize the drawing.

7.1. Switch from Model Space to the A3 Layout in AutoCAD.

7.2. Create a viewport by calling the Rectangular option (Fig. 2.97).

Ribbon → LAYOUT tab → LAYOUT VIEWPORTS panel →

 RECTANGULAR command.

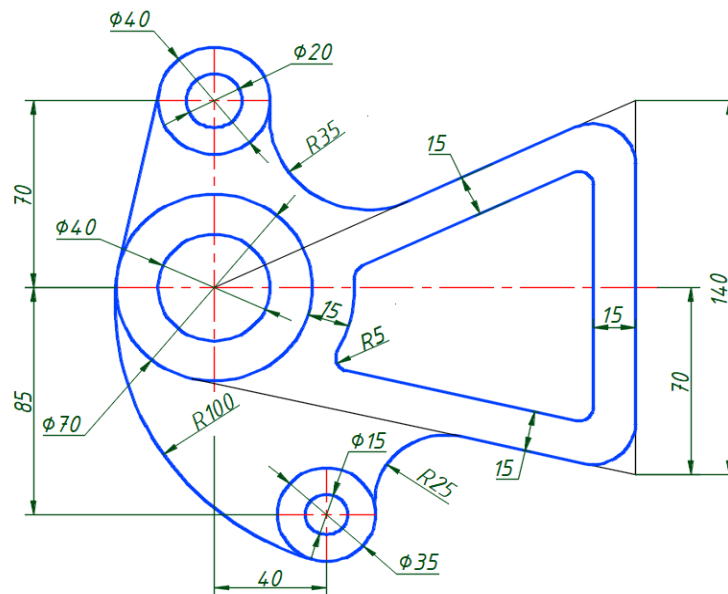


Fig. 2.96. Support bracket with dimensions applied

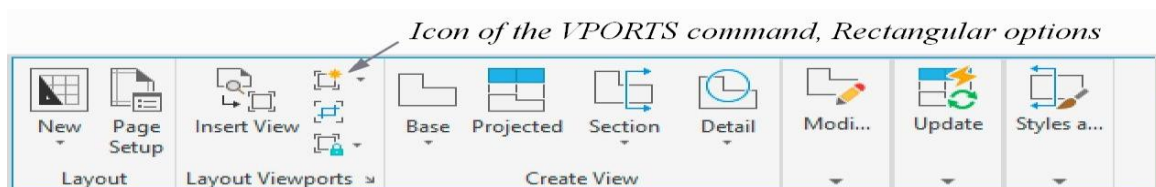


Fig. 2.97. Layout tab

When prompted, specify points 1 and 2 — the opposite corners of the rectangular viewport in which the drawing will be displayed (see Fig. 2.98).

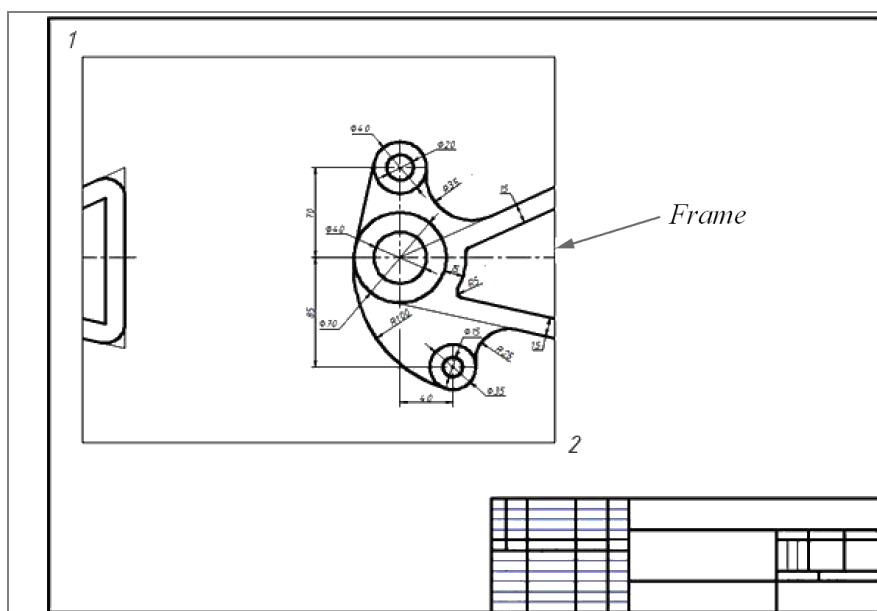


Fig. 2.98. Viewport in the Layout space

Double-clicking with the left mouse button outside the viewport frame allows you to edit the frame itself (move or resize it), while double-clicking inside the frame enables modifications of the displayed model.

7.3. Select the required scale (Fig. 2.99).

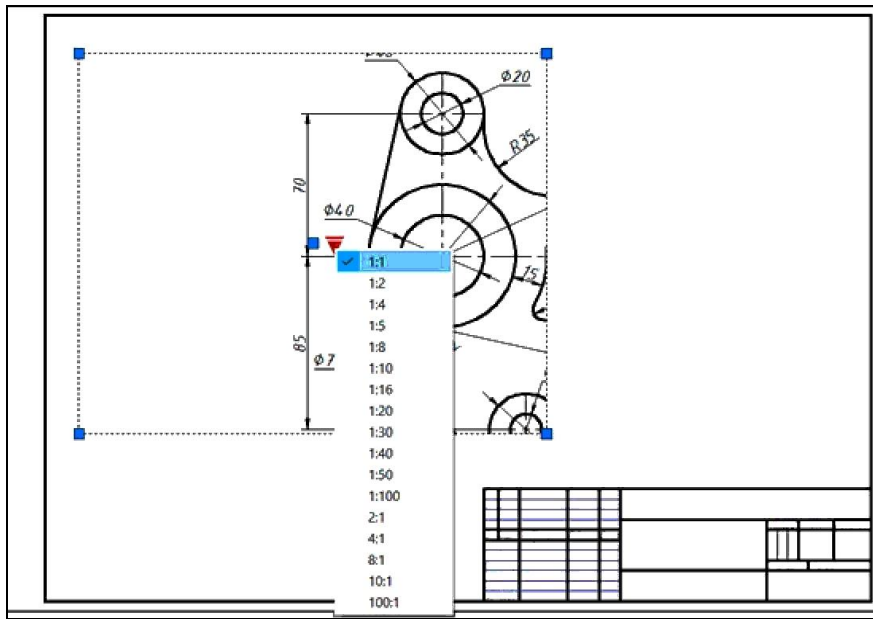


Fig. 2.99. Selecting the viewport scale

To open the scale list, select the viewport to activate its grips. Hover the cursor over the triangular grip at the lower left corner, and click the left mouse button. This action displays the list of available scales.

7.4. Fill in the title block (Fig. 2.100).

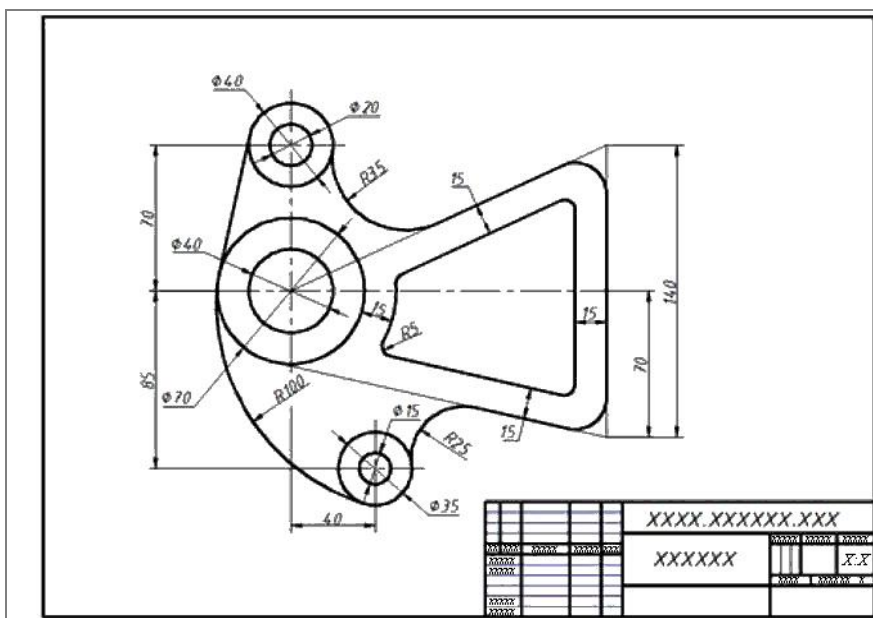


Fig. 2.100. Completed exercise "Fillets"

7.5. Move the viewport boundary to a layer that is turned off for printing.

8. Save the created geometric model file.

2.3.13.2. Support Bracket

This example (see Fig. 2.101) demonstrates how to construct slopes and tapers within a part's profile using AutoCAD's two-dimensional modeling tools.

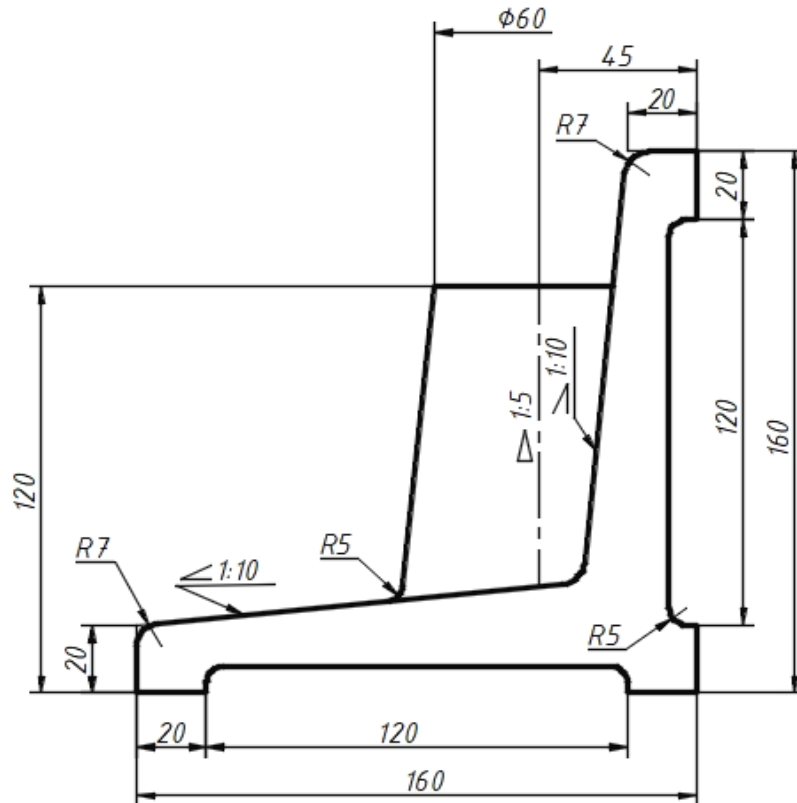


Fig. 2.101. Support Bracket

The tasks provided in Appendix B.1 are intended for independent practice.

Construction sequence

1. Begin by analyzing the geometry of the part, identifying simple constituent shapes and determining which drawing operations are needed to construct them efficiently.
2. Open the required template file, which contains the same predefined layers, dimension and text styles, border, and title block as in the previous example (see Subsection 2.3.13.1).
3. Set the current layer to *kontyr* and enable the ORTHO mode. This ensures that all horizontal and vertical elements are drawn precisely using the Line command (Fig. 2.102, a).
4. Slopes (tapers) on technical drawings are specified as numerical ratios.

For example, 1:10 indicates that a horizontal run of 10 units corresponds to a vertical rise of 1 unit. To construct such slopes using the Line command and relative coordinates:

- For the horizontal sloped segment, specify any starting point, then enter the second point as @10,1.
- For the vertical slope, start at an upper point and specify the second point as @-1,-10. The result is shown in Fig. 2.102, b.

5. Create fillets using the Fillet command with the Trim option enabled. This trims the intersecting segments automatically to form a clean tangent transition (Fig. 2.102, c).

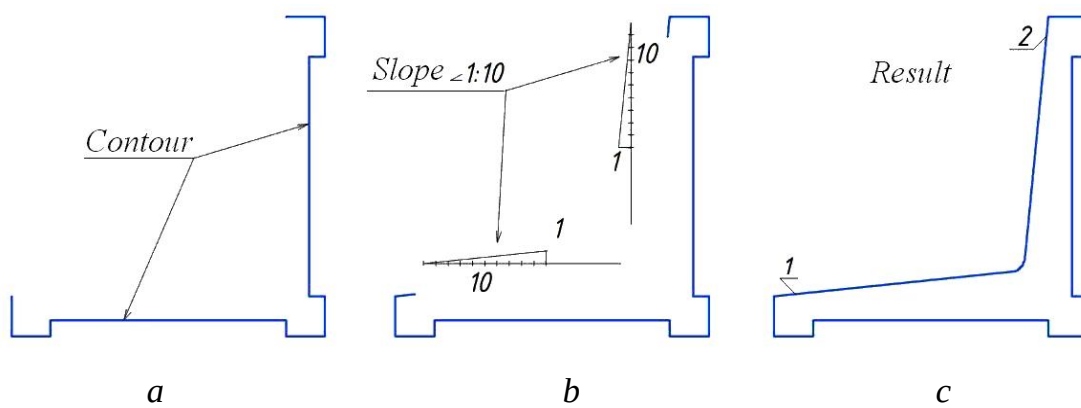


Fig. 2.102. Creating slopes 1:10

6. On the *osi* layer, draw the axis of the cone shown in Fig. 2.101. Use Xline with the Offset option to place this axis 45 mm from line 1 (Fig. 2.103, a).

7. Switch back to the *kontyr* layer. Use Xline → Offset to construct the upper base of the cone at a distance of 120 mm from line 2 (Fig. 2.103, a).

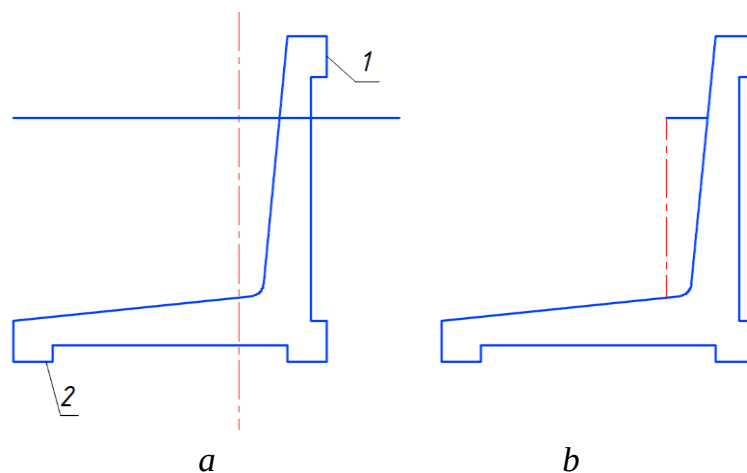


Fig. 2.103. Defining the axis and the upper base of the cone

8. Refine the contour using the Trim command (Fig. 2.103, b).

9. The diameter of the top cone base is 60 mm (radius 30 mm). Extend segment 1 using Lengthen (Delta = 30 mm) (Fig. 2.104, a). Draw line 3 using relative coordinates from point 2: @-1,-10 (Fig. 2.104, b). Use Extend to lengthen line 3 until it intersects line 4 (Fig. 2.104, c). Apply necessary fillets and refine the axis.

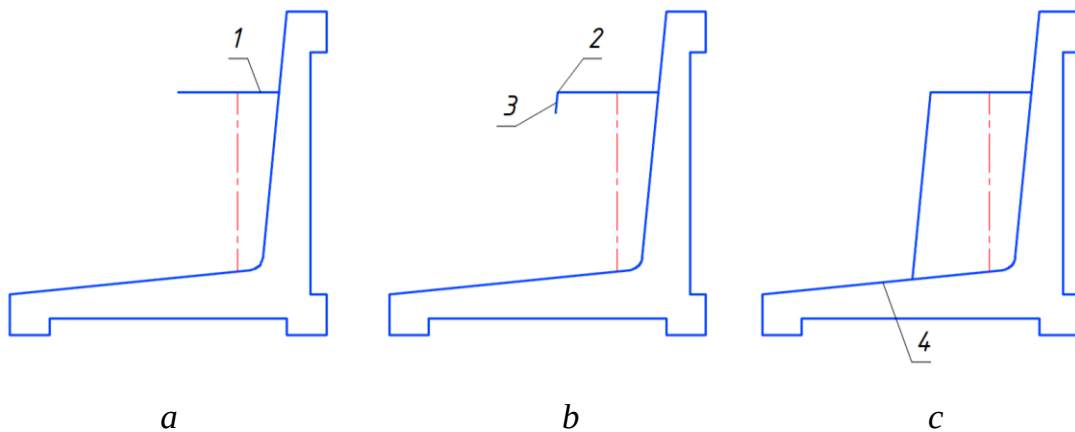


Fig. 2.104. Constructing the cone profile

10. Apply dimensions, selecting the *dim* layer as current (Fig. 2.105).

11. Prepare the drawing for output by switching to the A3 layout (see Fig. 2.105). The remaining steps—creating a viewport, assigning scale, filling the title block—match those in Subsection 2.3.13.1.

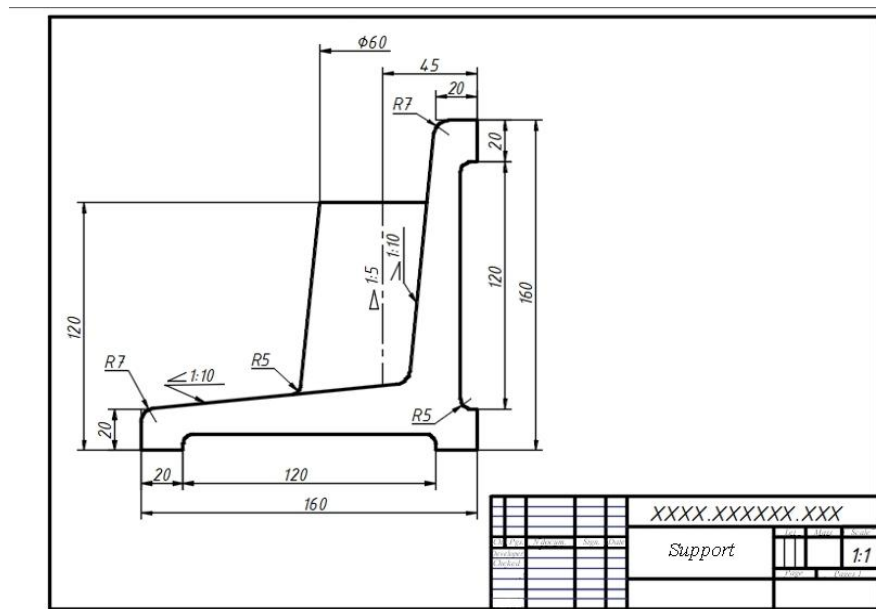


Fig. 2.105. Completed drawing

12. Save the finished file.

The following section discusses how the flexibility of two-dimensional geometric models in AutoCAD can be enhanced through the use of parametric tools.

2.3.14. Parametric Constraints

Geometric objects created in a two-dimensional workspace may be assigned a set of constraints that control their behavior. These include geometric constraints, which manage relationships such as parallelism, perpendicularity, horizontality, verticality, tangency, coincidence, symmetry, and fixation, and dimensional constraints, which specify linear, angular, and radial values.

Once applied, these constraints establish associative relationships among the objects, so that any modification of the shape, size, or position of one element automatically adjusts the others according to the defined parameters. Proper *parametrization* significantly increases efficiency when developing drawings of similar components, as automated updates reduce repetitive manual edits.

The tools for applying geometric and dimensional constraints are available on the Parametric tab of the Ribbon (Fig. 2.106).

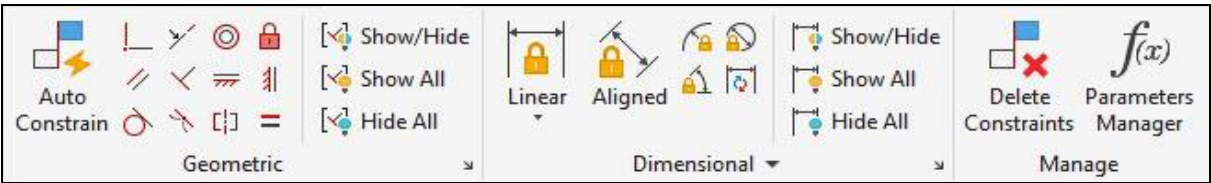


Fig. 2.106. Parametric tab

The subsequent subsections examine these tools in greater detail and provide several examples of their practical application.

2.3.14.1. Geometric Constraints

Geometric constraints in AutoCAD allow users to control the behavior of 2D geometry by enforcing precise relationships between objects, such as parallelism, perpendicularity, tangency, or symmetry. The command icons for geometric constraints are located on the Geometric panel (see Fig. 2.106).

Access path: Ribbon → PARAMETRIC tab → GEOMETRIC panel.

Table 2.2 lists the constraint symbols, their names, and their purposes.

Table 2.2

Geometric Constraints

Icon	Name	Function
	Coincident	Constrains objects to share a common point
	Collinear	Constrains two line segments to lie on the same straight line
	Concentric	Forces the centers of arcs, circles, or ellipses to coincide
	Fix	Locks an object in place, preventing it from being moved
	Parallel	Ensures line segments remain parallel
	Perpendicular	Ensures two line segments remain perpendicular
	Horizontal	Constrains an object to a horizontal orientation
	Vertical	Constrains an object to a vertical orientation
	Tangent	Ensures tangency between two objects
	Smooth	Applies a smooth (G2) continuity condition to a spline
	Symmetric	Ensures objects remain symmetrical with respect to a selected axis
	Equal	Forces selected lines or radii to maintain equal length

Once constraints are applied, the corresponding symbols appear next to the constrained objects. These icons may be repositioned anywhere within the drawing area for better visibility.

The display of constraint symbols is managed through the Show/Hide, Show All, and Hide All buttons located on the Geometric panel of the Parametric tab (see Fig. 2.106).

2.3.14.2. Automatic Parametrization

When creating or editing geometry with the  Constraint Infer mode enabled in the status bar, AutoCAD automatically applies geometric constraints

based on the object snaps used during construction. For example, if two circles are created with the same center, the software will automatically impose a coincident constraint on their centers. As a result, moving one circle will cause the other to move with it. However, certain object snaps – such as Intersection or Extension – are not supported for automatic constraints.

To delete a constraint, hover the cursor over its icon and right-click to open the context menu (Fig. 2.107). Select Delete from the list.

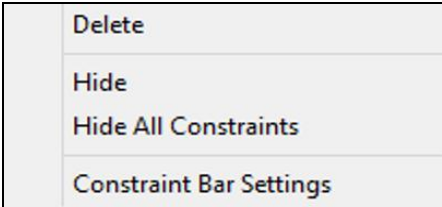


Fig. 2.107. Context menu of a constraint icon

This menu also provides additional management options for constraints, as shown in the figure.

2.3.14.3. Dimensional Constraints

Dimensional constraints define distances between objects (or their characteristic points), required angles, and the radii of arcs and circles. When these parameters are modified, all constraints applied to the geometry are recalculated automatically, and the associated objects update accordingly.

The types of dimensional constraints correspond to the icons on the Dimensional panel of the ribbon and are listed in Table 2.3.

Table 2.3

Dimensional Constraints		
Icon	Name	Function
	Aligned	Applies a constraint to the distance between two points
	Linear	Creates a horizontal or vertical distance constraint
	Horizontal	Constrains the X-distance between two points
	Vertical	Constrains the Y-distance between two points
	Diameter	Applies a constraint to the diameter of a circle
	Radius	Applies a constraint to the radius of an arc or circle
	Angular	Applies a constraint to an angle

Dimensional constraints may consist of numbers or expressions. The preferred display format can be selected from the Dimension name format drop-down menu in the Constraint Settings dialog box (Fig. 2.108). This dialog is opened by clicking the corresponding icon  on the Dimensional panel of the Parametric tab.

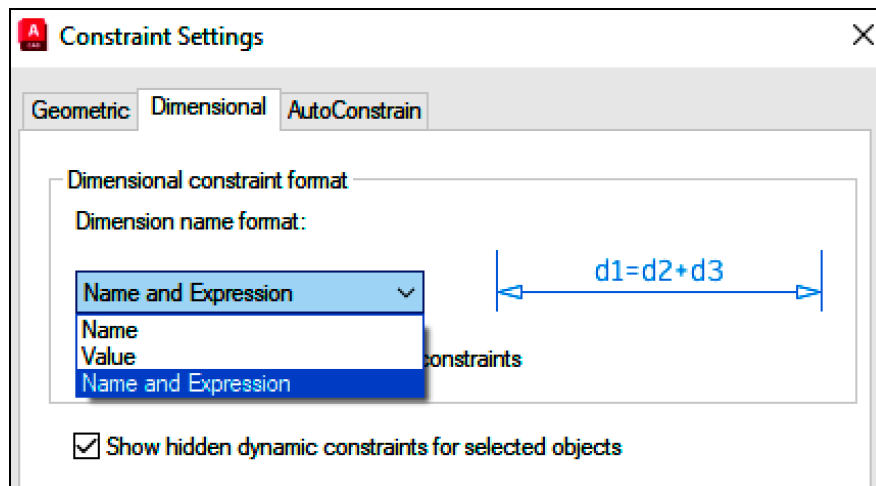


Fig. 2.108. Constraint Settings dialog box

Name – constraint name; Value – numerical value; Name and Expression – combined name and formula.

Dimensional constraints are displayed in the drawing using parameters, which visually represent the applied relationships (Fig. 2.109).

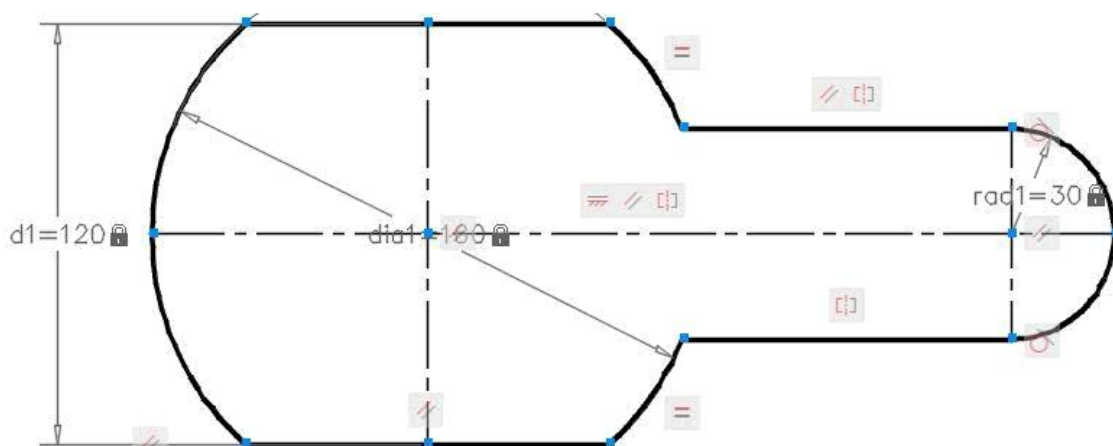


Fig. 2.109. Profile with applied dimensional parameters

Editing dimensional constraints is performed using the Parameters Manager  (Fig. 2.110), accessed via the icon located on the Manage panel of the Parametric tab. After double-clicking the name, expression, or value in the table, a new entry can be typed from the keyboard.

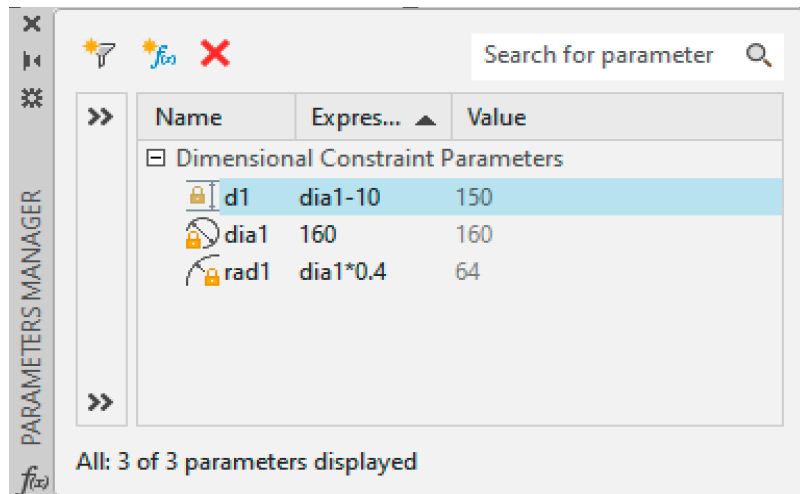


Fig. 2.110. Parameters Manager with defined constraints

Expressions may include the standard arithmetic operators—addition (+), subtraction (−), multiplication (*), division (/), exponentiation (^), and parentheses for defining the order of evaluation. AutoCAD also supports a range of mathematical functions within dimensional expressions, including the trigonometric functions cos, sin, and tan; their inverse counterparts acos, asin, and atan; the square-root function sqrt; and several numerical utilities such as round for rounding to the nearest integer, trunc for discarding the fractional part, floor and ceil for rounding down or up, and abs for returning the absolute value. When a parameter or its expression is modified in the Parameters Manager, the corresponding geometry automatically updates (Fig. 2.111).

Dimensional constraints may also be edited through the Properties panel (Fig. 2.112) for a selected parameter.

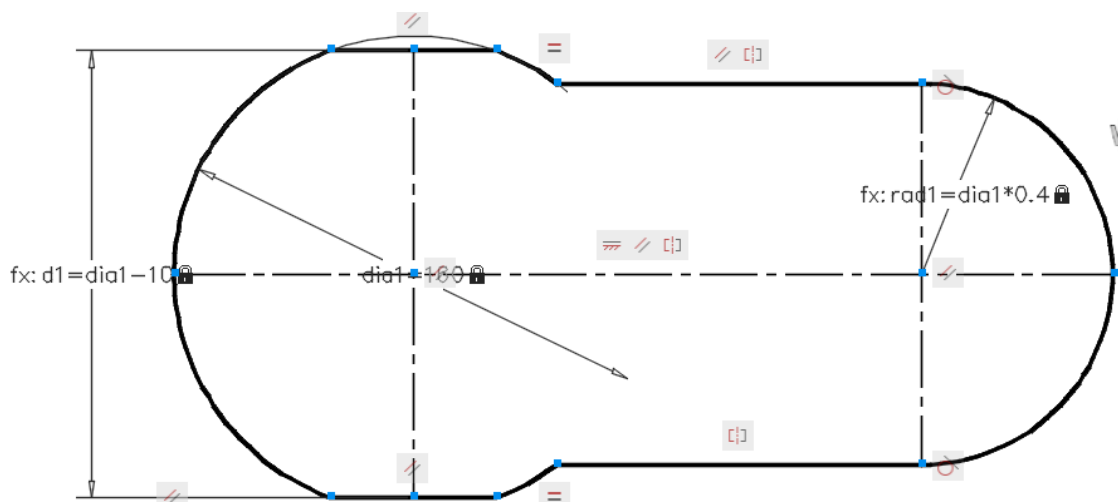


Fig. 2.111. Profile updated after modifying dimensional constraints

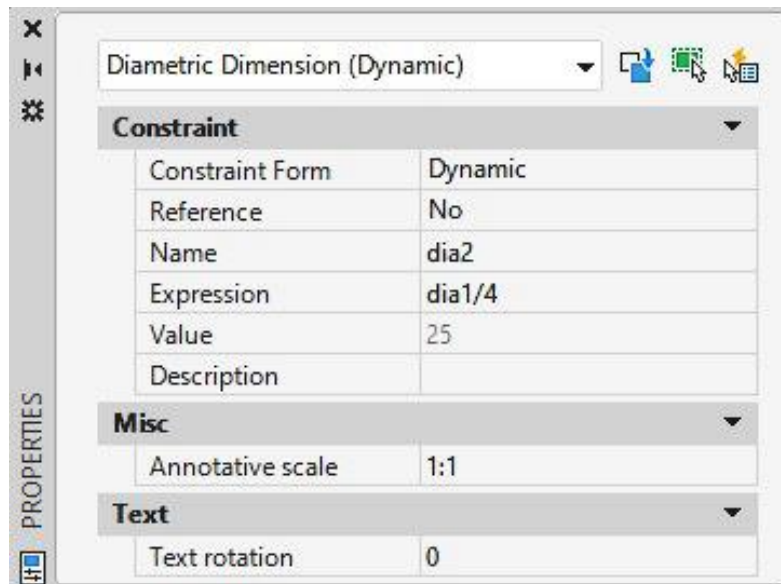


Fig. 2.112. Properties panel for the selected constraint

A constant or expression can additionally be edited by double-clicking the parameter directly in the drawing.

2.3.15. Projection Drawing

In technical drafting, the geometry of a three-dimensional object is represented through a system of orthographic projections – views, sections, and auxiliary cuts – that maintain precise projection correspondence (see Appendix A). AutoCAD provides a variety of tools for constructing such drawings directly in 2D, ensuring accurate alignment between related views.

This section introduces several practical methods for generating projectionally linked views within the AutoCAD environment.

2.3.15.1. Tracking Method

The Tracking method relies on a combination of auxiliary drafting modes designed to assist in determining point locations that must correspond across different views. These modes are actively used throughout the construction process and can be toggled directly from the Status Bar by clicking their respective icons:

 Object Snap (OSNAP) – for selecting exact geometric features (endpoints, midpoints, centers, etc.).

 Object Snap Tracking – for projecting temporary alignment paths from OSNAP points.

 Polar Tracking – for constraining cursor movement along preset angles.

Together, these tools allow the user to construct views that share common projection lines without manually drawing additional construction geometry.

It is advisable to keep only the necessary Object Snap modes enabled, as excessive snaps may interfere with precise point selection.

2.3.15.2. *Using the XLINE and RAY Commands*

The XLINE command generates construction lines of infinite length, while RAY creates half-infinite rays. These auxiliary lines should preferably be placed on a dedicated construction layer, which can later be turned off or deleted once the necessary projections have been completed. The main drawback of this method is the large number of lines that may accumulate when working with complex objects.

Consider the following example. Suppose it is required to construct the profile view of a rectangular prism based on its front and top views. On the construction layer, use the XLINE command to create horizontal and vertical projection lines.

Command: XLINE

Specify a point or [Hor/Ver/Ang/Bisect/Offset]: H – creates a horizontal line.

Command: XLINE

Specify a point or [Hor/Ver/Ang/Bisect/Offset]: V – creates a vertical line.

Next, draw the axes of symmetry: a horizontal axis on the top view and a vertical axis on the left-side view, extending both until they intersect. Through the obtained intersection point, construct the 45-degree miter line using XLINE on the construction layer (Fig. 2.113, a).

Command: XLINE

Specify a point or [Hor/Ver/Ang/Bisect/Offset]: A – selects the Angle option.

Enter angle of xline (0) or [Reference]: -45

From the points where the horizontal projection lines intersect the reference line, draw vertical construction lines. The distance between these lines corresponds to the width of the prism. Using Object Snap at the intersections of the projection lines, construct the left-side view. After turning off the construction layer, the three orthographic views of the prism are obtained (Fig. 2.113, b). The procedure is the same for the Ray command, which creates semi-infinite lines instead of full construction lines.

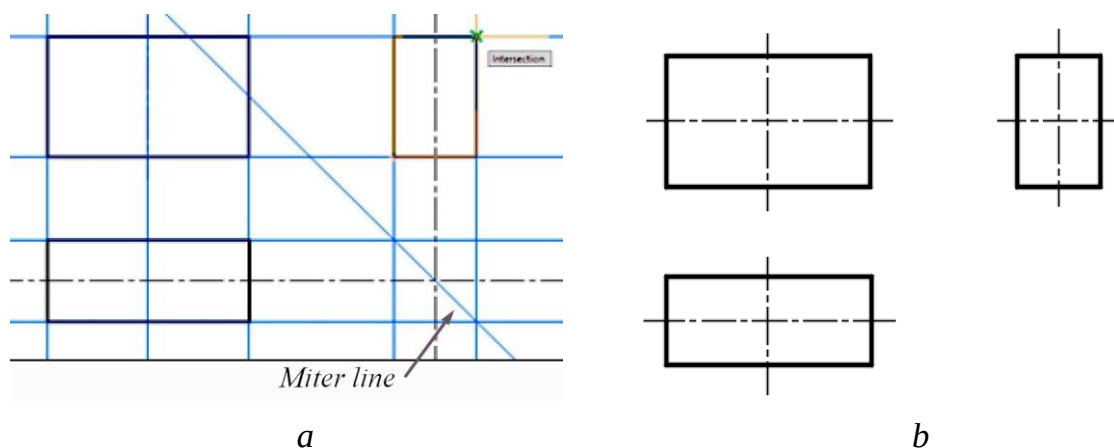


Fig. 2.113. Construction of the projection drawing, construction layer:
a – auxiliary layer on; *b* – auxiliary layer off

2.3.15.3. Hatching

The Hatch command in AutoCAD is intended for filling closed areas with hatch patterns, solid fills, or gradients. Access is provided as follows: Ribbon → HOME tab → DRAW panel →  HATCH command.

In the dropdown menu on the panel, you can select the command used to fill a boundary (Fig. 2.114).

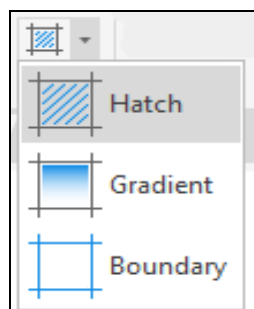


Fig. 2.114. Hatch and fill commands

The Hatch and Gradient commands begin by automatically analyzing the selected geometry to determine a valid closed boundary. Once the boundary is recognized, AutoCAD generates a preview and applies the chosen fill method. This preliminary boundary-detection step is essential because the quality and correctness of the hatch depend entirely on how accurately the enclosing region is defined.

The Hatch command then opens a contextual panel on the Ribbon, providing a full set of tools and options for controlling pattern type, scale, angle, origin, transparency, and boundary selection methods (Fig. 2.115). This panel appears only while the command is active, allowing the user to adjust all hatch parameters interactively.

The Gradient command opens a similar contextual panel (Fig. 2.116), but tailored specifically for gradient fills. It offers a selection of gradient styles, one-color or two-color transitions, and corresponding color settings. All other Ribbon tools remain unchanged while these panels are active, ensuring a focused workspace for defining the fill.

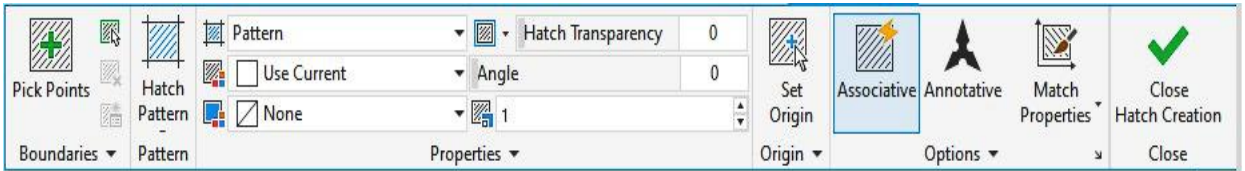


Fig. 2.115. Hatch creation tools

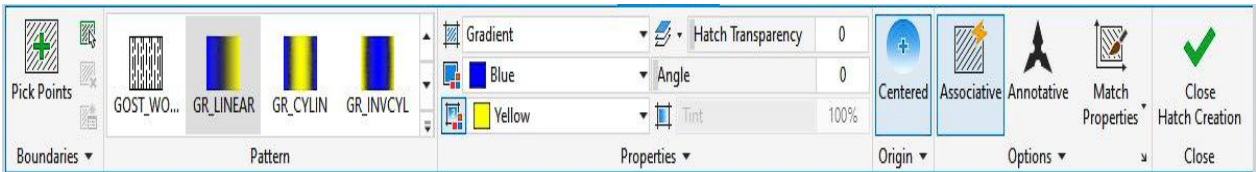


Fig. 2.116. Gradient and solid fill tools

The Boundary command is used to generate a single boundary from multiple elements. Its construction mode is defined in the Boundary Creation dialog box (Fig. 2.117).

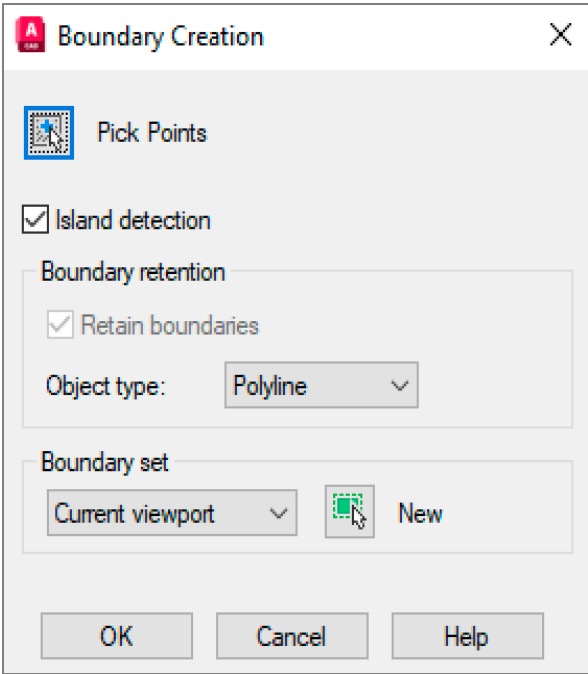


Fig. 2.117. Boundary Creation dialog box

The Boundaries panel is shown in Fig. 2.118.

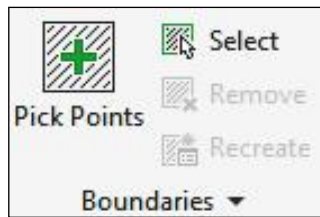


Fig. 2.118. Boundaries panel

 **Pick Points** – enables automatic boundary detection inside an enclosed region. To hatch the area, simply click inside it.

 **Select** – allows defining the boundary using explicitly selected objects. This method is suited for simple regions formed by one closed shape.

The Pattern panel contains a library of hatch patterns (Fig. 2.119), presented as preview slides for intuitive selection.

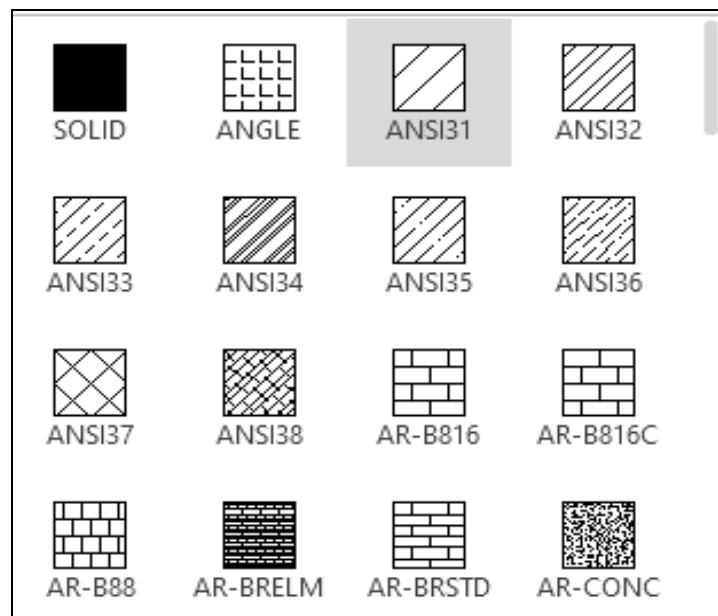


Fig. 2.119. Pattern panel contents

In the standard AutoCAD library, the most commonly used hatching patterns for mechanical engineering drawings are the diagonal ANSI-series patterns. The primary one is ANSI31, which consists of diagonal lines at a 45° angle with uniform spacing between them (hatch pitch) and is used for filling cross-sections of parts. In assembly drawings, adjacent components may be differentiated with ANSI32–ANSI38, which vary in line frequency and orientation.

The SOLID fill is used to create fully shaded areas or to highlight individual elements when necessary. Other patterns (such as AR-CONC, AR-B816) are intended mainly for architectural materials and are generally not used in mechanical engineering.

Selecting an appropriate hatching pattern ensures clear visual separation of contours and complies with the drafting standards for technical drawings.

On the Properties panel (Fig. 2.120), the Pattern drop-down menu allows choosing the fill type: a pattern hatch, a solid fill, a gradient, or a user-defined hatch.

These variants are shown in Fig. 2.121.

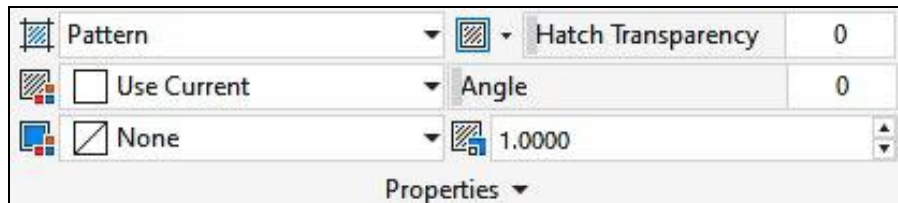


Fig. 2.120. Properties panel

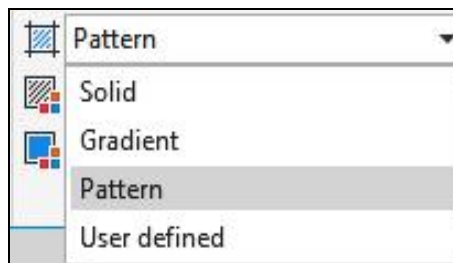


Fig. 2.121. Types of boundary fills

The User-defined hatch type is intended for custom settings based on parallel lines. Their spacing and angle can be adjusted as needed (Fig. 2.122).

The Hatch Transparency field controls transparency; Angle sets the hatch angle relative to the current X-axis; Scale defines the pattern scale (Fig. 2.123).

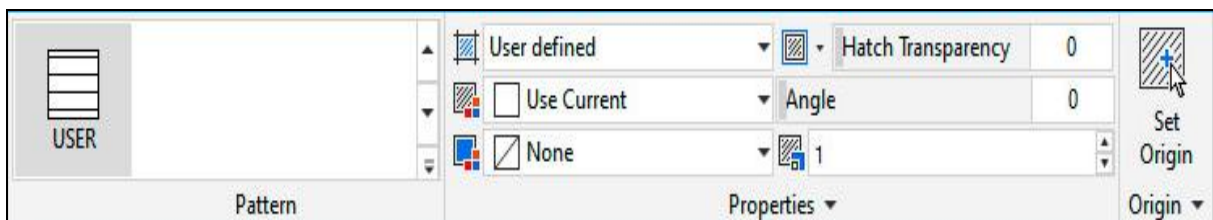


Fig. 2.122. User-defined hatch settings

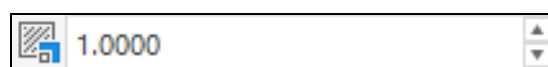


Fig. 2.123. Defining hatch scale

The Origin panel offers tools for specifying the hatch origin point (Fig. 2.124). This is particularly useful in assembly drawings where adjacent parts require offset hatch origins. By default, all origin options correspond to the current UCS origin.



Fig. 2.124. Hatch origin options

The Options panel is shown in Fig. 2.125.

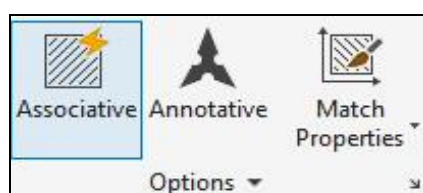


Fig. 2.125. Options panel

Associative – enables associativity: the hatch updates automatically when its boundary changes (Fig. 2.126, a). When disabled, the hatch remains unchanged (Fig. 2.126, b).

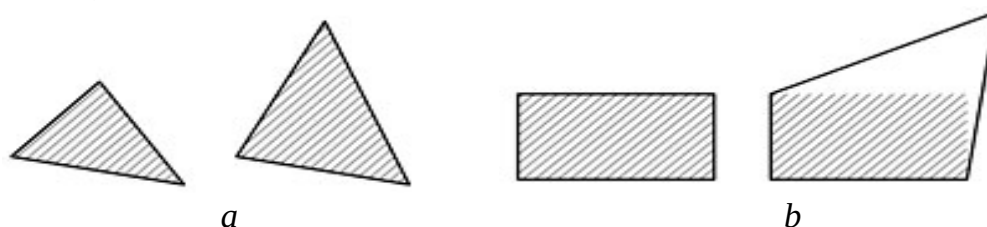


Рис.2.126. Hatch:
a – associative; b – non-associative

Annotative – makes the hatch scale automatically with the viewport annotation scale.

Match Properties – creates a hatch using the properties of an existing hatch, except for the origin point.

In the lower-right corner of the Options panel, a button opens the Hatch and Gradient dialog box (Fig. 2.127). This dialog box provides full access to the tools for defining and editing hatch patterns, gradient fills, and their parameters. Its structure consists of several functional sections:

Type and pattern. Allows the user to choose the hatch type (Predefined, User-defined, or Custom) and select a specific pattern from the Swatch preview. Color assignment is also performed here.

Angle and scale. Defines the orientation and scaling coefficient of the selected hatch pattern. The fields Angle and Scale provide numerical input for rotation and pattern size.

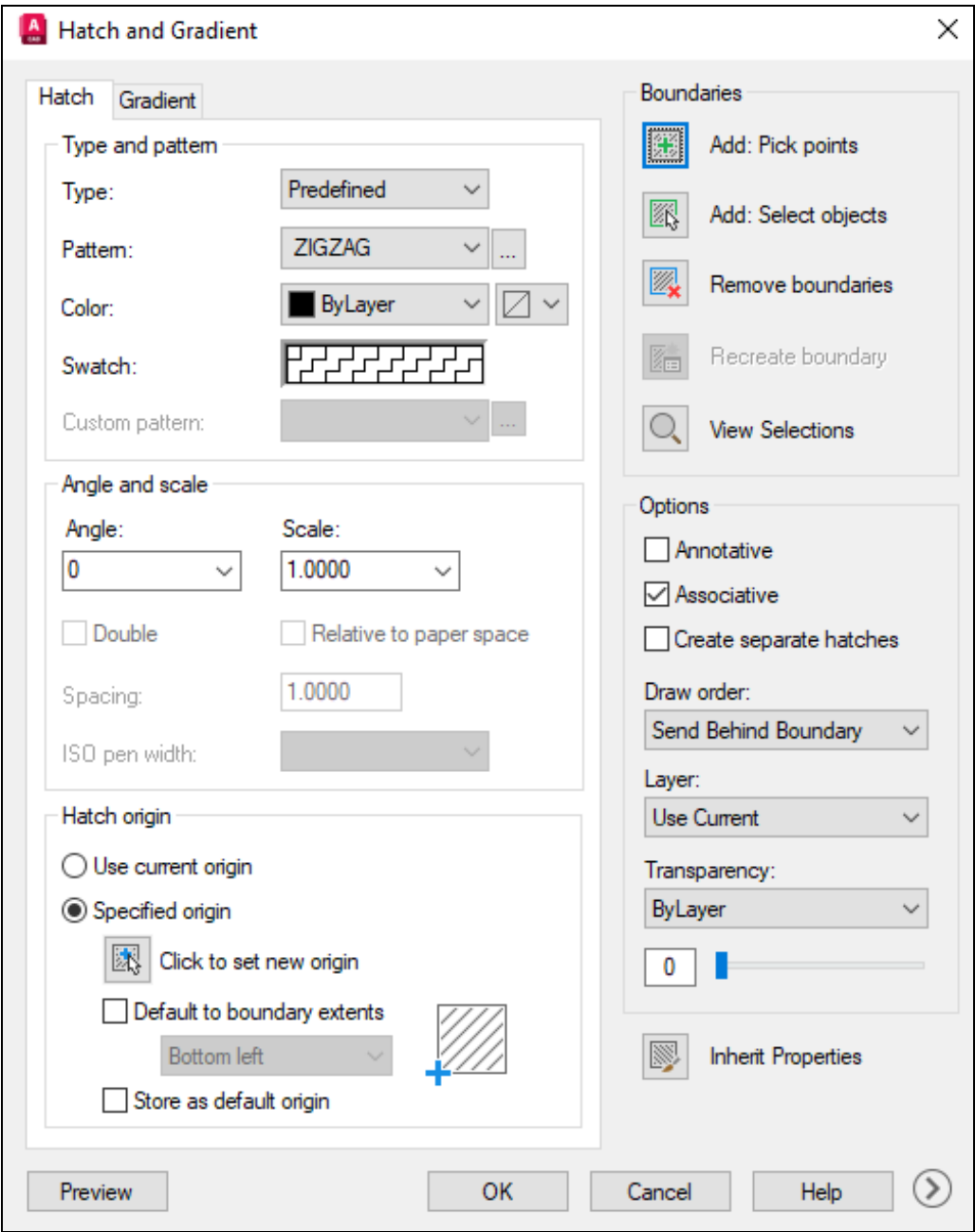


Fig. 2.127. Hatch and Gradient dialog box

Hatch origin. Controls the starting reference point for the hatch. The user may keep the current origin or specify a new one by picking a point on the drawing. This tool is especially useful when coordinating hatch alignment across adjacent regions in assembly drawings.

Boundaries panel. Contains commands for defining or modifying hatch boundaries: Pick Points, Select Objects, and Remove Boundaries. The View Selections option displays the boundary elements currently used.

Draw order. Defines the display priority of the hatch relative to other drawing elements (e.g., placing the hatch behind the boundary linework).

Layer and transparency. Allows the user to assign a specific layer for the hatch and set the transparency level. These properties help integrate the hatch into the structural hierarchy of the drawing.

Inherit Properties. Enables copying hatch properties from an existing hatched object for consistent formatting.

Clicking the small triangle next to the panel title opens an additional menu (Fig. 2.128). The Gap Tolerance field specifies the maximum allowed gap between boundary elements. The default tolerance is zero, requiring a perfectly closed boundary (Fig. 2.129).

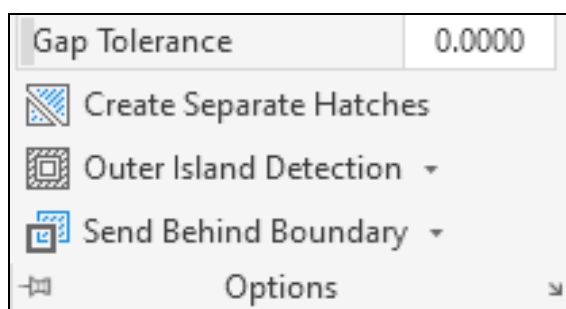


Fig. 2.128. Hatch boundary selection controls

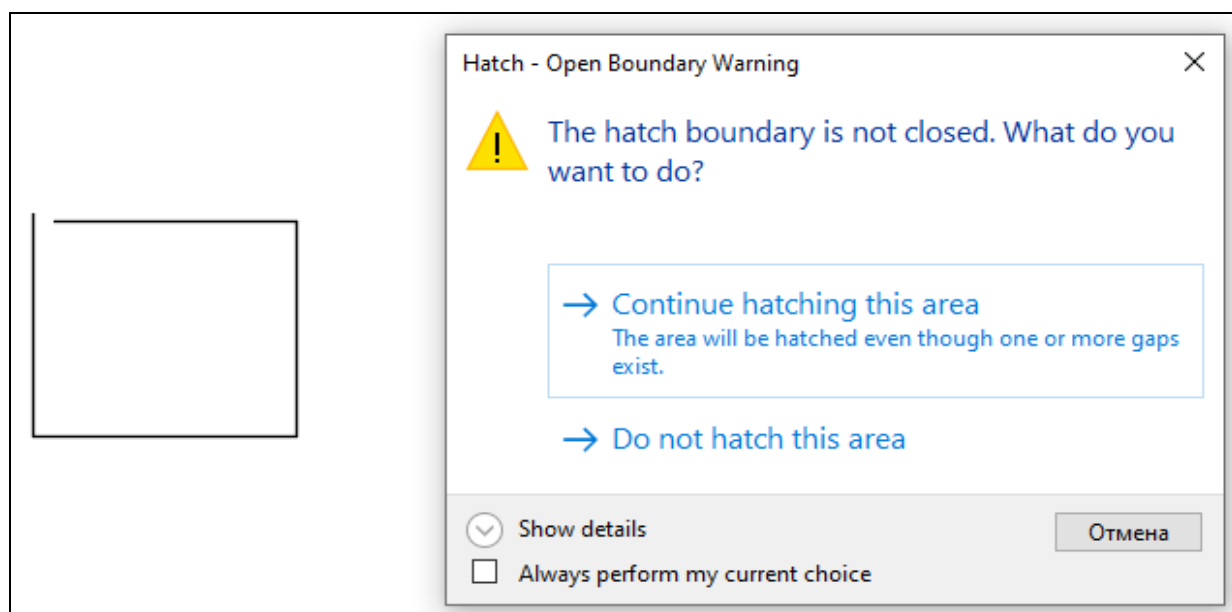


Fig. 2.129. Warning for open boundary gaps

If the tolerance is set to a value greater than zero, hatching is applied even to areas with gaps (Fig. 2.130).

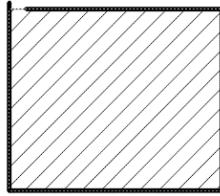


Fig. 2.130. Hatching with non-zero gap tolerance

Create Separate Hatches enables generating separate hatch objects that can be edited independently.

Outer Island Detection opens a menu for selecting how hatches behave when the boundary contains interior “islands” (Fig. 2.131).

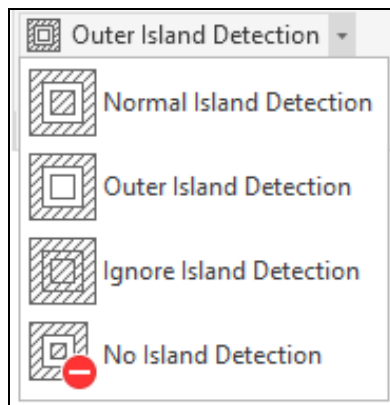


Fig. 2.131. Hatch options with internal “islands”

Send Behind Boundary defines the draw order of the hatch relative to other objects—behind or in front of the boundary and other drawing elements (Fig. 2.132).

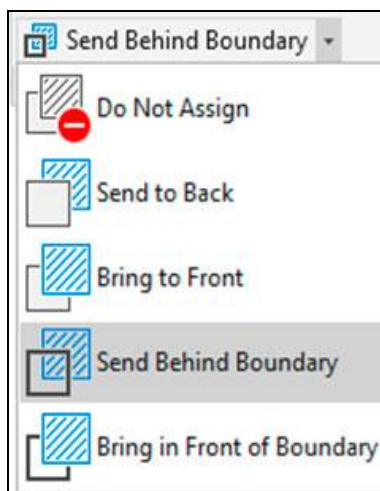


Fig. 2.132. Hatch placement variations

The Close Hatch Creation button collapses the hatch creation panel and finishes the command.

2.3.15.4. Example of Constructing an Orthographic Drawing

It is well known that orthographic projection is widely used in the preparation of technical documentation. The essential construction tools were outlined earlier in this section.

Before creating the drawing, it is necessary to:

- Analyze the geometric model and determine which basic forms the object consists of;
- Establish the principal view that provides the most complete representation of the part;
- Determine the minimum required number of views and sections.

To develop practical skills in constructing orthographic drawings, the following example demonstrates the creation of the three principal views with the corresponding sectional views. Variants of individual tasks are provided in Appendix B.2.

Let the object to be modeled have the appearance shown in Fig. 2.133. The task is to construct the required views and sections of the part. Use layers, polar and object tracking, and apply dimensions.

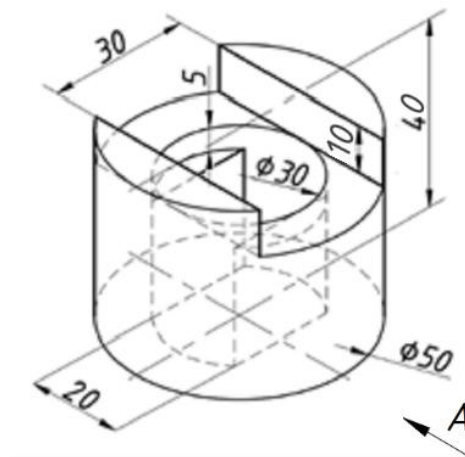


Fig. 2.133. Modeled part

Sequence of construction

1. Analyze the part, which contains cylindrical surfaces, a through slot, and a stepped vertical hole. The upper portion of the hole is circular and cylindrical, while the lower portion additionally includes flat vertical faces.

2. Open the template file, whose structure is assumed to be similar to those discussed in subsections 2.3.13.1 and 2.3.13.2. Add a layer named *hatch* for section *hatching*. Save the file under the name “Simple Cross-Sections” (“Rozrizy prosti”).

3. Create the drawing in model space at a scale of 1:1.

4. Set the current layer to *osi* and construct the axes using the XLINE command (Fig. 2.134).

5. Using the Circle command, draw the outer circle 1 (diameter 50 mm) and the inner circle 2 (diameter 30 mm) whose centers coincide at the intersection of the axes. With the Offset command, construct vertical lines (3) and horizontal lines (4) offset relative to the vertical and horizontal axes. Edit the resulting geometry using Trim.

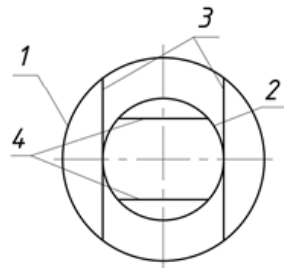


Fig. 2.134. Top view

6. Construct the front view in projection from the top view. Activate the auxiliary drafting modes  Polar Tracking,  Osnap, and  Object Snap Tracking via the Status Bar. Enable the essential object snaps: endpoint and intersection.

7. Using the Line command, draw the outline of the front view (Fig. 2.135) in projection from the top view.

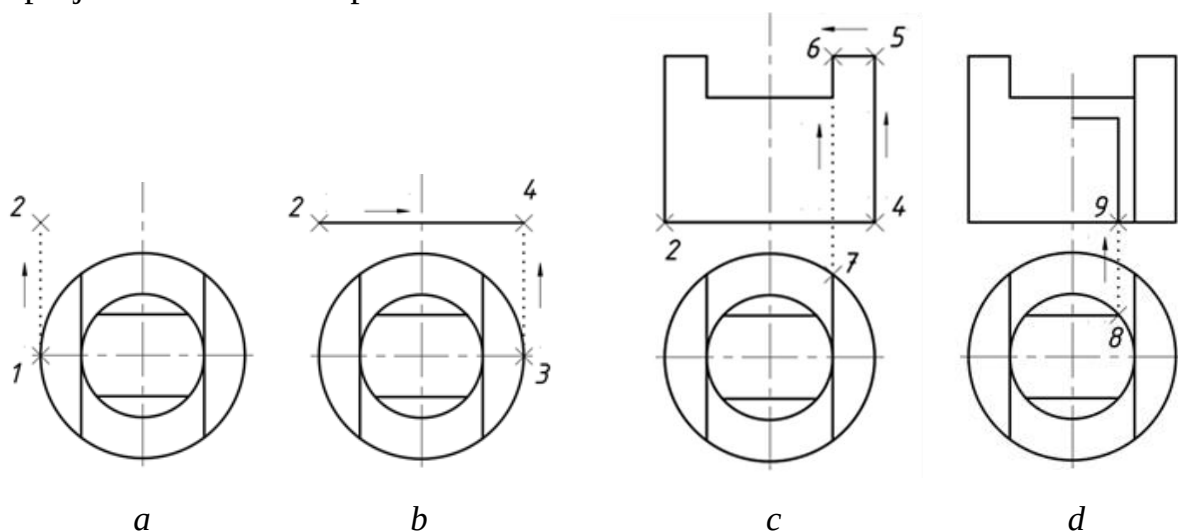


Fig. 2.135. Sequence of constructing the front view

To accomplish this, move the cursor to point 1 after starting the command and hold it there until the snap marker appears. Then move the cursor upward. A temporary construction tracking line will appear, projecting point 1 (Fig. 2.135, a). Click to create point 2 on this line. The views should be placed sufficiently close to one another to simplify tracking.

Move the cursor rightward from point 2, and when a tracking line appears, move it toward point 3 (Fig. 2.135, b). When the snap marker appears, move the cursor upward again. Continue until the tracking lines intersect at point 4, then click. Specify the vertical direction and draw the segment from point 4 to point 5 (Fig. 2.135, c). Then move the cursor leftward from point 5. After the tracking line appears, move it toward point 7. Once the snap marker is visible, move the cursor upward until the tracking lines from points 7 and 5 intersect at point 6. Click to confirm.

Construct the remaining segments in a similar manner. Construct the internal contour (Fig. 2.135, d) by tracking point 9 from point 8.

8. Construct the left-side view in projection from both the front and top views. To do this, create the 45-degree miter line (Fig. 2.136, a), which serves as a reference for transferring dimensions between the existing views.

It is recommended to place it on a construction layer, which can be turned off once the reference line is no longer needed. To create this line, use the Line command. Specify point 1 as close as possible to the existing views and, with polar angle tracking enabled, draw a line at an angle of -45° . Through the intersection of the horizontal axis of symmetry of the top view and the reference line, construct the vertical axis of symmetry for the left-side view, making *osi* the current layer.

Begin constructing the left-side view with the outer contour. Verify that the required object snaps are active. In this case, enable endpoint, intersection, and quadrant. After launching the Line command, track point 1 horizontally toward the reference line (Fig. 2.136, b). Type *ttp* (temporary track point) on the keyboard. When the cursor reaches the reference line at temporary point 2 (arrow 1) and the crosshair marker appears, click the left mouse button. Change the tracking direction to vertical (arrow 2). Move the cursor back to point 3 and begin shifting it to the right. A horizontal tracking line from point 3 will appear (arrow 2). At the intersection of this tracking line and the previously obtained vertical line from point 2, fix point 4. Then determine point 5 by tracking from point 6 and point 7 from point 8 (Fig. 2.136, c). Once half of the contour is completed, mirror it about the axis of symmetry using the Mirror command (Fig. 2.137).

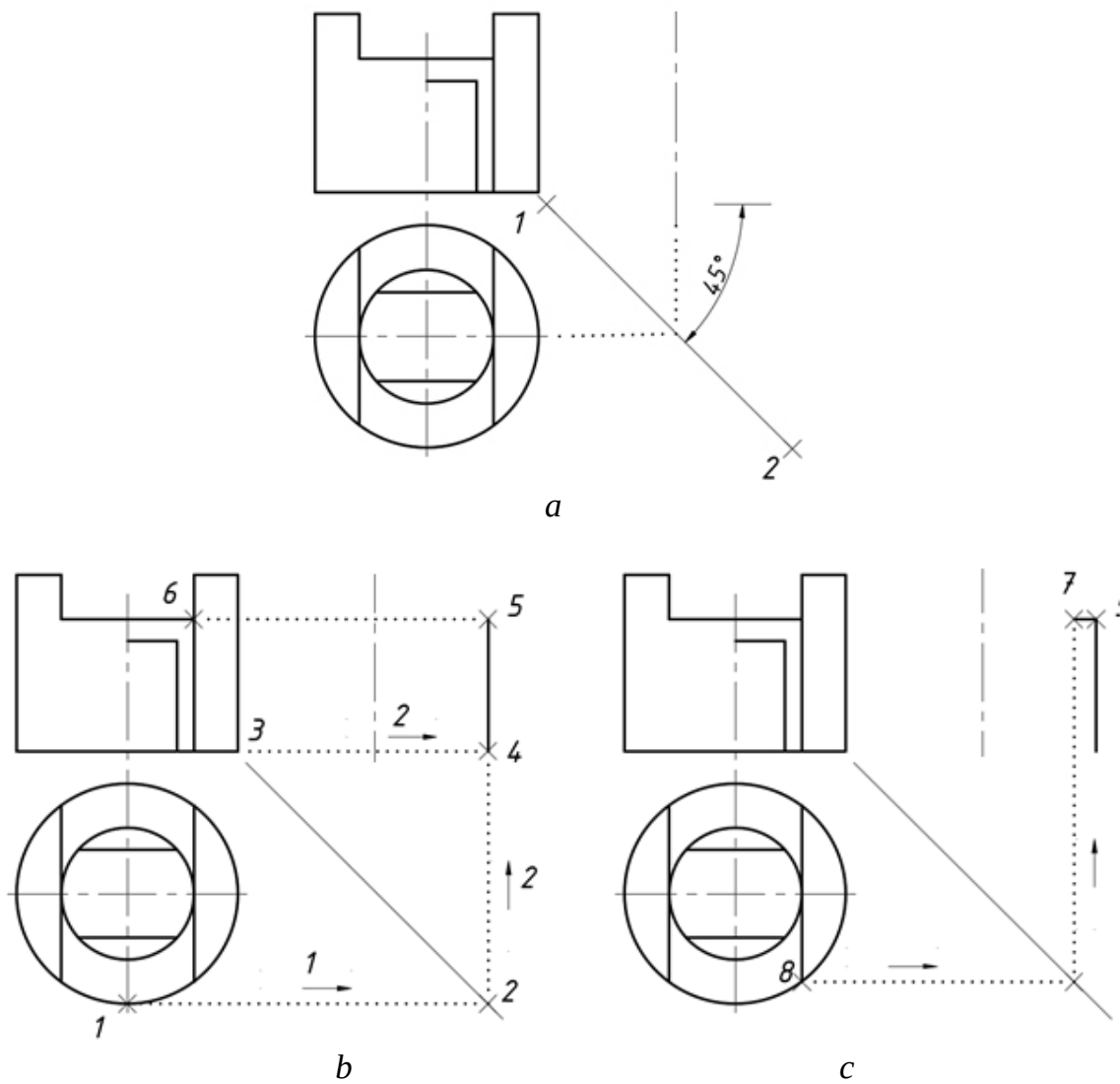


Fig. 2.136. Constructing the left-side view in projection from the other two views

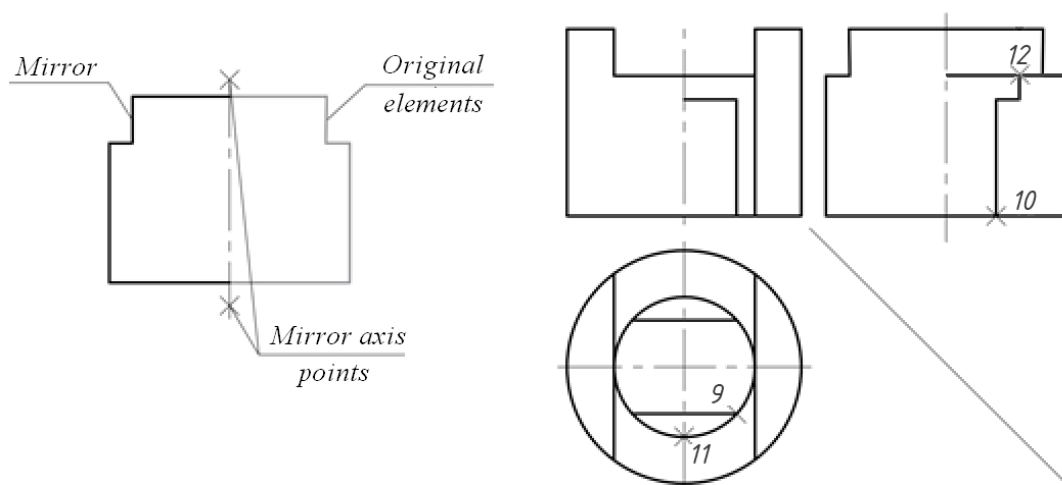


Fig. 2.137. Using the Mirror command to generate the symmetric half of the view

Construct the inner contour in the sectioned region using the same technique – by tracking point 10 from point 9, and point 12 from point 11.

9. Apply hatching to the sectional view. Set the *hatch* layer current. Select the required region in the front view (Fig. 2.138) and specify the hatch pattern (desired style, angle, and spacing). It is recommended to create the hatching for each view separately to simplify subsequent editing.

10. Add the dimensions.

11. Complete the drawing in paper space using a scale of 2:1 (Fig. 2.139).

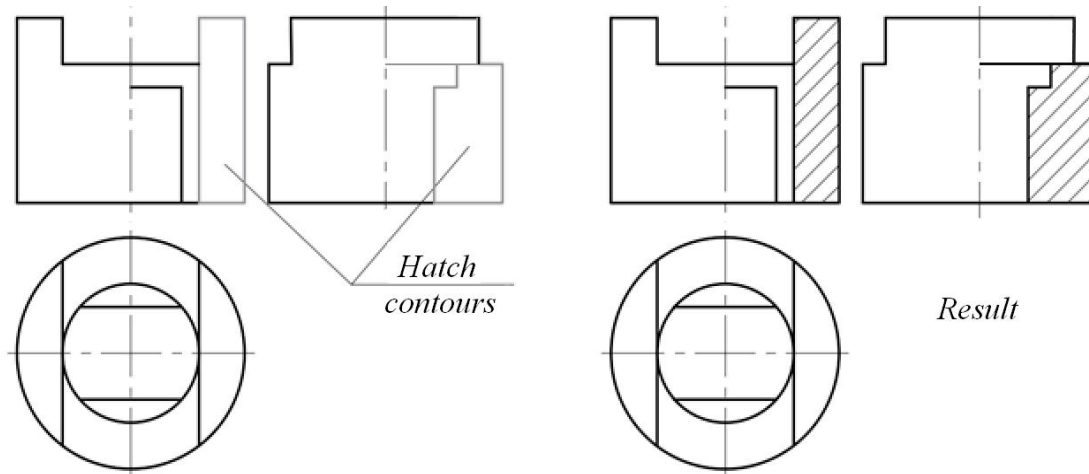


Fig. 2.138. Hatching in the sectional views

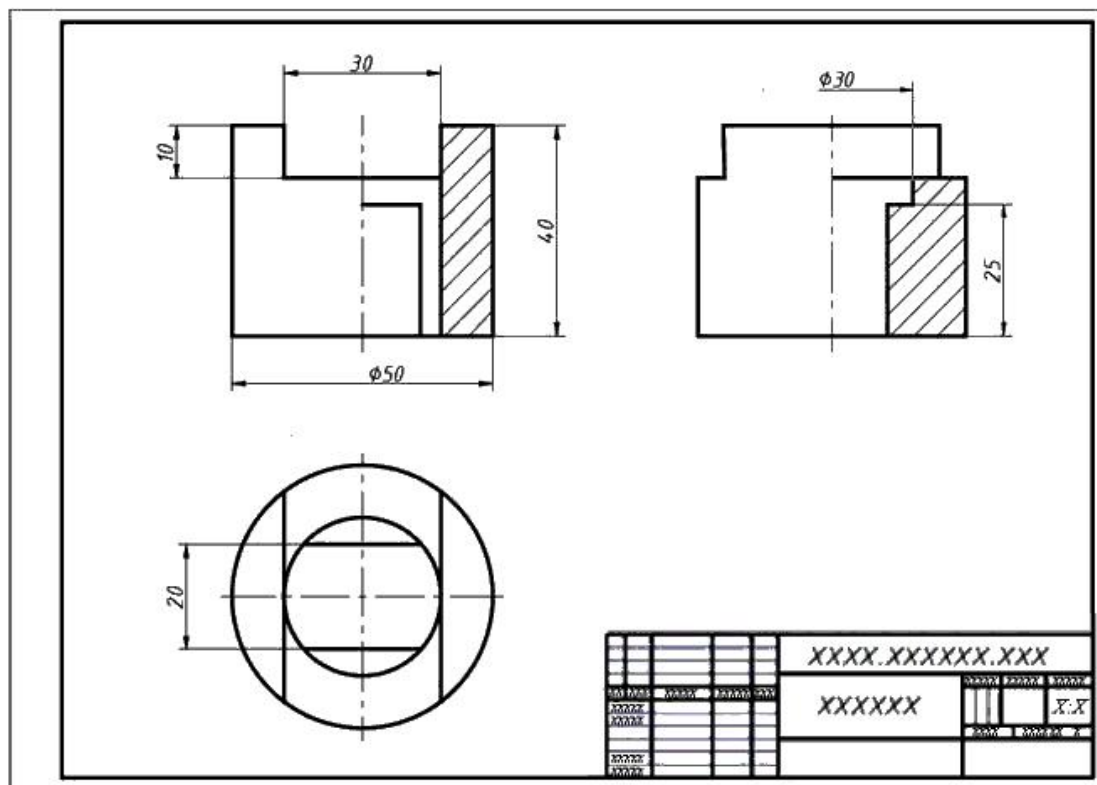


Fig. 2.139. Completed drawing

2.4. Solid Modeling

Solid modeling in computer-aided design is an advanced stage of two-dimensional automated geometry creation. The resulting 3D models offer high realism and are convenient for a wide range of scientific, engineering, and practical applications. All previously acquired knowledge and skills will be essential for understanding the material presented in this section.

2.4.1. 3D Modeling Workspace

AutoCAD includes a dedicated 3D Modeling workspace for creating and editing three-dimensional geometry. When you switch to this workspace using the status bar, the ribbon layout changes accordingly (see Fig. 2.140), and the available tool panels will be described later.

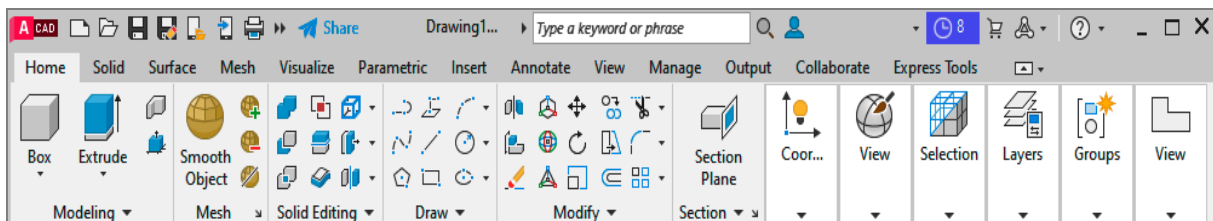


Fig. 2.140. Ribbon layout in the 3D Modeling workspace

Note that the graphical area used for 2D drafting is actually a special case of the 3D modeling environment, because in 2D mode the Z-axis is oriented perpendicular to the screen. When you rotate the view (Shift + mouse wheel), the Z-axis becomes visible, and all existing geometry lies in the XY plane.

2.4.2. Viewport Controls

The viewport controls appear in the upper-left corner of any active viewport (Fig. 2.141). They provide quick access to view settings, visual styles, and other display options.



Fig. 2.141. Viewport controls

Clicking the +/- control opens a menu where you can configure multiple independent viewports using the Viewport Configuration List (Fig. 2.142). When multiple viewports are active, each one includes its own control bar, letting you adjust each viewport separately.

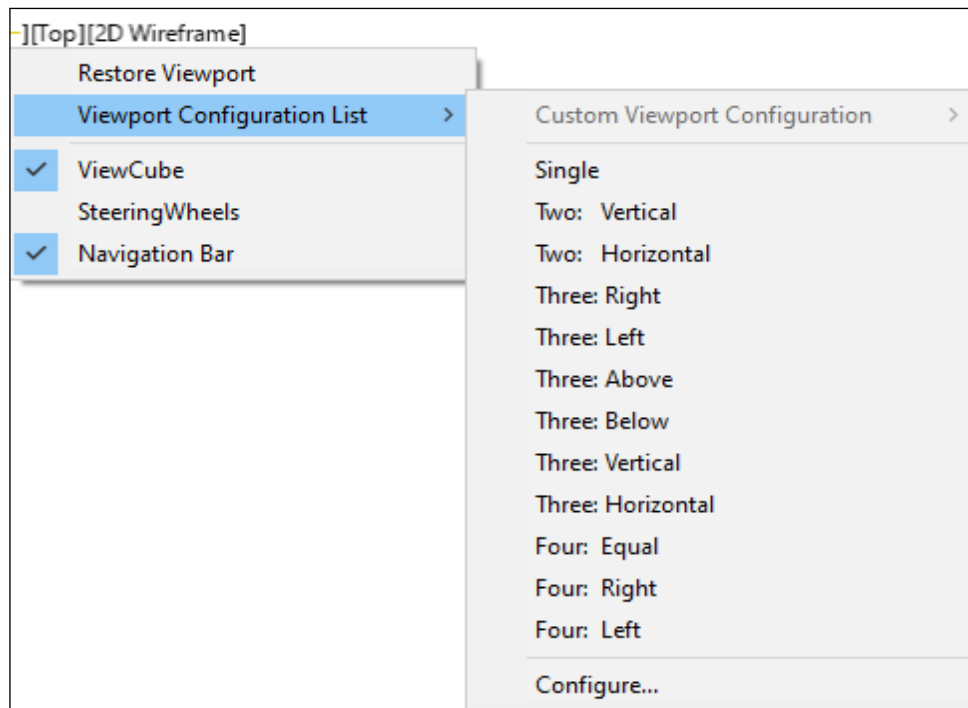


Fig. 2.142. Viewport Configuration List

The View control (default setting) displays the current viewing direction for the selected viewport. It expands to show a full list of preset views – top, front, isometric, etc. – as well as projection types such as parallel and perspective (Fig. 2.143).

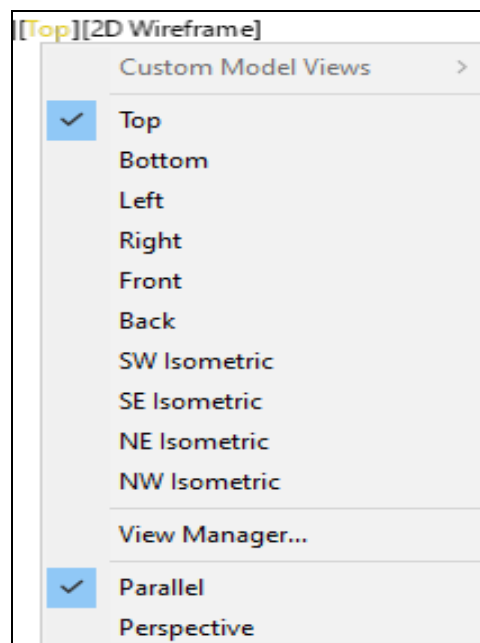


Fig. 2.143. List of views available for the active viewport

If up to four viewports are displayed simultaneously, each one can show a different view of the model. To activate a viewport, hover over it and click the left mouse button. The active viewport is highlighted in blue (Fig. 2.144).

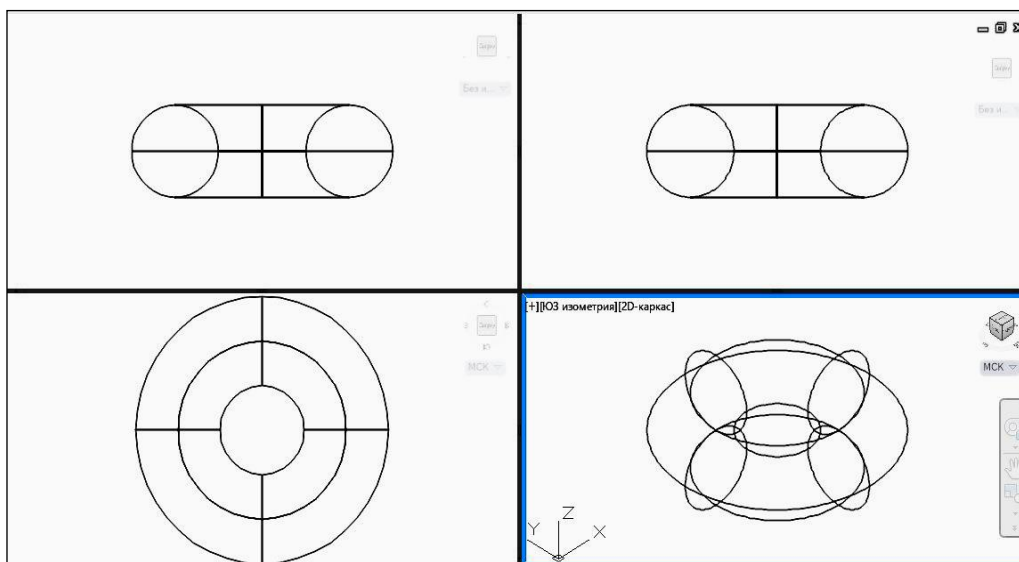


Fig. 2.144. Example of a four-viewport configuration with customized views

The 2D Wireframe button indicates the current visual style. Clicking it opens a selection of built-in visual styles (Fig. 2.145). From here, you can also access the Visual Styles Manager for advanced settings.

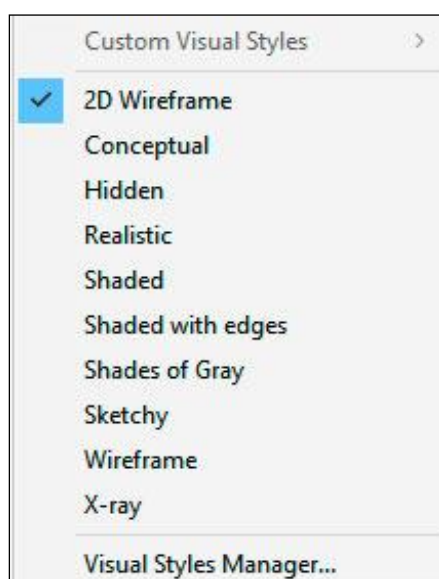


Fig. 2.145. List of visual styles

2.4.3. ViewCube

The ViewCube is a 3D navigation widget (Fig. 2.146) typically located in the upper-right corner of the drawing area. You can hide or show it using the ViewCube option in the navigation tools settings accessed through the [+/-] menu (Fig. 2.147).



Fig. 2.146. ViewCube

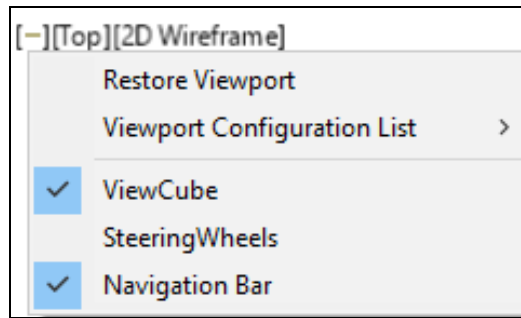


Fig. 2.147. Enabling or disabling the ViewCube

Each part of the ViewCube responds to selection:

- Clicking a *face* switches to a primary orthographic view (Top, Front, Left, etc.).
- Clicking an *edge* sets an intermediate view between major directions.
- Clicking a *corner* selects a standard isometric view.

2.4.4. Three-Dimensional Coordinates

3D coordinates, like 2D ones, can be defined in absolute values (measured from the origin) or relative values (measured from the last specified point). For objects in 3D space, Cartesian, cylindrical, and spherical coordinate systems are used.

Cartesian coordinates define a location using three coordinate directions X, Y, and Z, and are used similarly to their two-dimensional form: X, Y, Z.

Cylindrical coordinates correspond to 2D polar coordinates with the addition of the Z-coordinate.

Spherical coordinates are defined by the distance from the origin, the angle between the radius vector and the X-axis in the XY plane, and the angle of its inclination to that plane.

2.4.5. Coordinate System Management

The World Coordinate System (WCS) is the fixed, default Cartesian system used by AutoCAD. Any other coordinate system created by the user is a User Coordinate System (UCS). The active UCS defines the working XY plane for modeling and the orientation of the Z-axis.

To modify the current UCS or create a new one in the 3D Modeling workspace, use: Ribbon → HOME → COORDINATES panel (see Fig. 2.148).

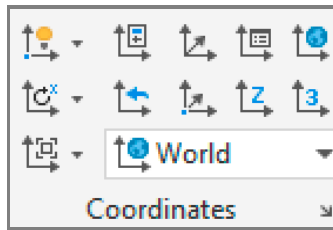


Fig. 2.148. Coordinates tool panel

AutoCAD provides the following UCS tools:

- Toggle the UCS icon and change its placement.
 - Set the UCS icon style.
 - Control and edit UCS settings.
 - Manage named UCS configuration.
 - Switch to the World Coordinate System.
 - Rotate the UCS around the X-axis (additional options for Y and Z are available)
 - Restore the previous UCS.
 - Define a new UCS by specifying a new origin.
 - Align the UCS with the positive Z direction.
 - Define a UCS by three points (origin, point on +X, point on +Y).
- Align the UCS XY plane to:
- the screen plane;
 - an object;
 - a solid face.

2.4.6. Visualization Control of Geometric Objects

AutoCAD provides a comprehensive system of visual styles, which determine how three-dimensional models are displayed on the screen. Visual styles define a combination of parameters such as shading, lighting, transparency, edge visibility, and the influence of materials. These settings allow users to choose the most appropriate representation of a model for drafting, analysis, or presentation.

By default, AutoCAD displays 3D geometry using a wireframe, which shows only the edges of objects. While this representation is lightweight and efficient for editing, it is not always visually clear—especially when dealing with complex or densely detailed models.

In the 3D Modeling workspace, the View panel of the Home ribbon tab contains a drop-down list of available visual styles (Fig. 2.149).

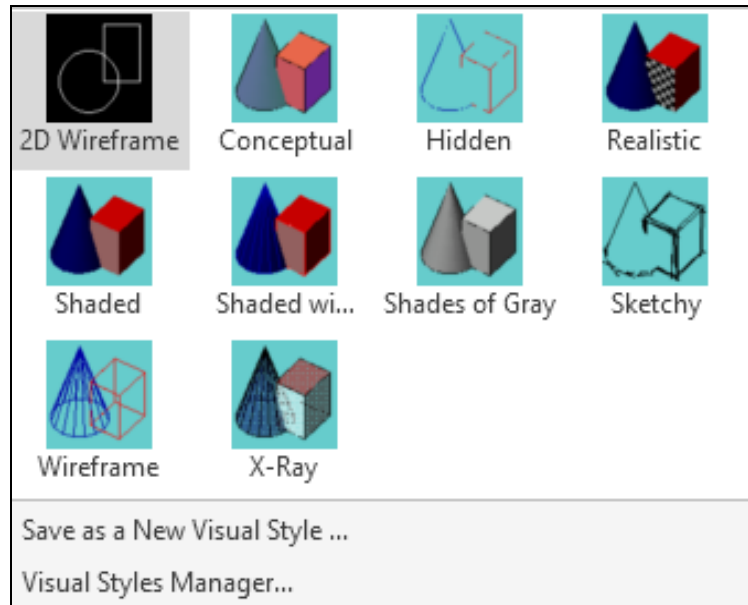


Fig. 2.149. Visual styles drop-down menu

Each style provides a different way of interpreting the geometry:

2D Wireframe – displays objects with clean, optimized linework suitable for precise 2D drafting; ideal when working primarily in plan or elevation.

Conceptual – applies stylized shading with smooth transitions between warm and cool tones, improving visual perception of form during the conceptual design stage.

Hidden – presents the model as a wireframe but automatically removes the lines associated with hidden edges, making the geometry easier to interpret.

Realistic, Shaded, Shaded with Edges, Shades of Gray – a group of shading-based modes:

- Realistic displays materials and textures, giving an almost rendered appearance.
- Shaded focuses on smooth color transitions without applying material properties.
- Shaded with Edges combines shading with clearly visible object edges.
- Shades of Gray provides a neutral grayscale visualization for better contrast and reduced distraction.

Sketchy – simulates a hand-drawn sketch, which may be useful for early design communication or stylistic presentations.

Wireframe – shows detailed 3D wireframe geometry, including all edges.

X-Ray – applies a semi-transparent effect, allowing the user to see internal elements or overlapping components without changing the model's structure.

The selected visual style affects only the active viewport, allowing different viewports to display the same model in different ways – for example, wireframe in one window and realistic shading in another.

At the end of the list, the Visual Styles Manager button (Fig. 2.150) provides access to advanced settings. Here, users can browse existing visual styles, review their predefined parameters (lighting models, edge settings, transparency, materials, etc.), and create customized styles tailored to specific project needs. This enables greater control over how the model is interpreted visually during various design stages.

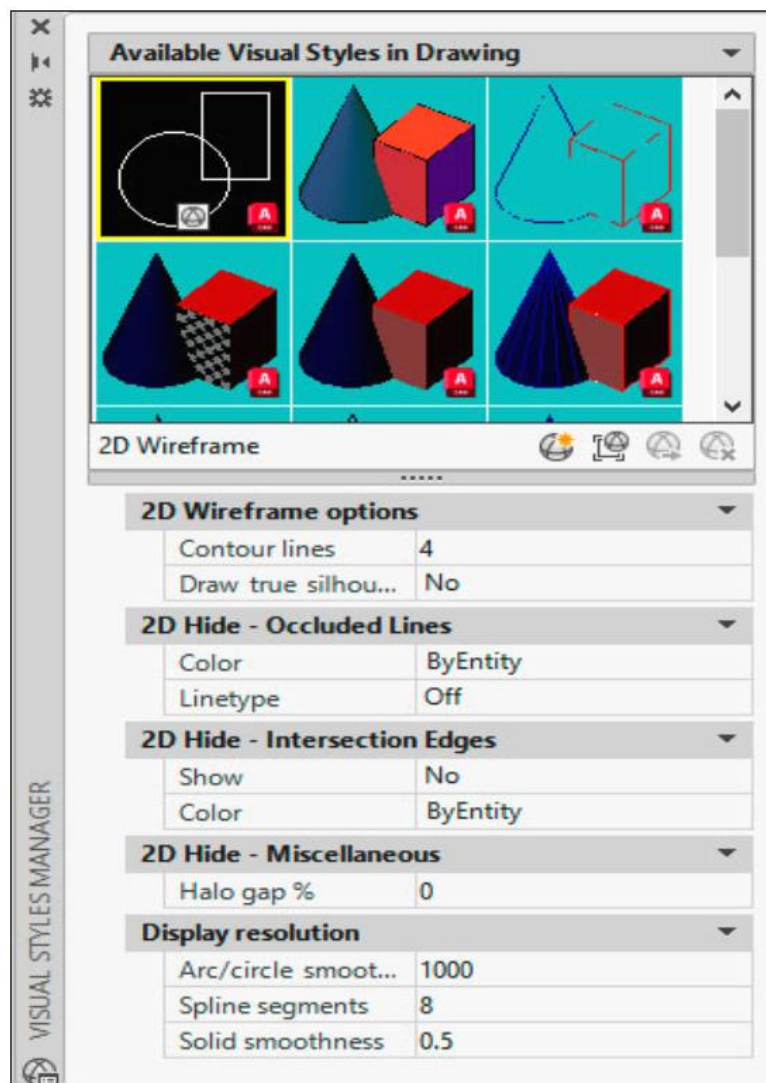


Fig. 2.150. Visual Styles Manager

2.4.7. Creating Solids (Primitives)

In the 3D Modeling workspace, AutoCAD provides a dedicated set of tools on the Solid tab for creating basic three-dimensional shapes known as solid primitives (Fig. 2.151). These primitives serve as the foundational elements for building more complex models and are often used as starting geometry for further editing, Boolean operations, or surface modifications.

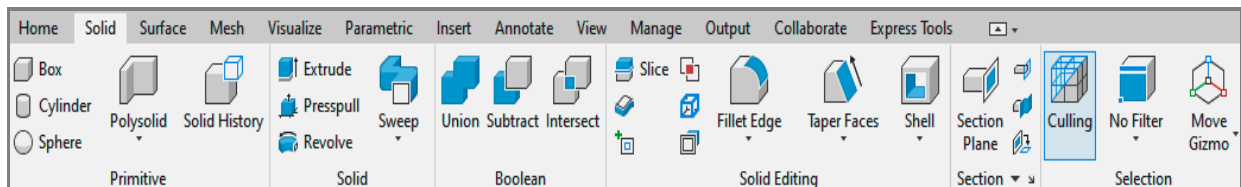


Fig. 2.151. Tool panels of the Solid tab

Let us now consider the tools for constructing several 3D primitives.

2.4.7.1. Box

The BOX command creates a rectangular solid whose base lies on the XY plane of the active UCS (Fig. 2.152). This is one of the most frequently used primitives, as many objects can be modeled efficiently by starting with a box and then modifying or combining it with other solids.

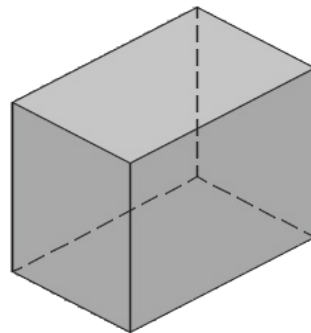


Fig. 2.152. Parallelepiped

To access the tool: Ribbon → SOLID → PRIMITIVE →  BOX. After activating the command, AutoCAD prompts the user through several input options:

Command: BOX

Specify first corner or [Center]: – specify the first corner of the base in the usual way.

Specify other corner or [Cube/Length]: – defines the opposite corner of the base rectangle.

The Center option allows creation of a box by specifying the center point of the base instead of a corner. This is useful for symmetrical modeling or when placing an object relative to a midpoint.

Specify center: – specify the base center;

Specify corner or [Cube/Length]: – specify a base corner or choose an option;

Specify height or [2Point] <>: – height can be either typed numerically or defined using two picked points.

Choosing Cube constructs a solid where all three dimensions are equal, forming a perfect cube based on the entered edge length.

The Length option enables modeling a parallelepiped by entering:

- Length – the dimension along the X-axis of the current UCS,
- Width – the dimension along the Y-axis,
- Height – which can be entered numerically or defined by two points.

This method gives precise control over orientation and proportions when the direction of the edges relative to the UCS matters.

2.4.7.2. Cylinder

The CYLINDER command creates a solid cylinder and supports both circular and elliptical bases. This makes it a versatile tool for modeling mechanical components, architectural elements, or any geometry based on rotational surfaces.

You can launch the command by navigating to: Ribbon → SOLID → PRIMITIVE →  CYLINDER.

When the command is activated, AutoCAD prompts the user to define the base:

Command: CYLINDER

Specify center point of base or [3P/2P/Ttr/Elliptical]: – specify the center point of the cylinder base.

Specify base radius or [Diameter] <>: – specify the radius or the diameter.

3P option – constructs the base of the cylinder by specifying three points that lie on the base circle.

2P option – defines the base using two opposite endpoints of the diameter.

Ttr option – constructs the base circle tangent to two selected objects in its plane and defined by a radius.

After defining the base using one of the above options, the program prompts for the height:

Specify height or [2Point/Axis endpoint] <>: – enter the height value or choose an option.

2Point option – define the height using two points.

Axis endpoint option – specify the second endpoint of the cylinder's axis. This option allows changing the inclination angle of the cylinder relative to the XY plane (Fig. 2.153).

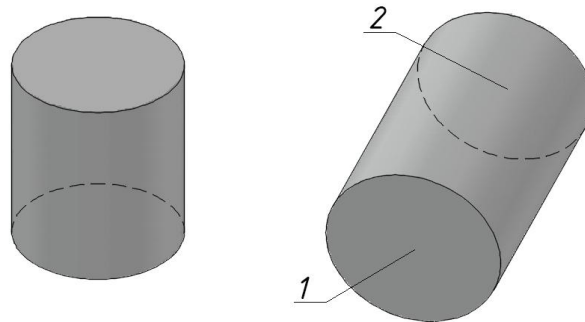


Fig. 2.153. Height and inclination defined by axis endpoints 1 and 2

Elliptical option – constructs an elliptical cylinder. When this option is selected, the program requests points for constructing the elliptical base and then the cylinder height.

Elliptical cylinder: choosing the Elliptical option switches the command to constructing an elliptical base:

- specify the major axis;
- specify the minor axis;
- define the height.

This enables modeling non-circular cylindrical forms, which are often required in architectural, industrial, or ergonomic design.

2.4.7.3. Cone

The CONE command allows the creation of several variations of conical solids, including:

- a full right cone,
- a truncated cone (frustum),
- an inclined cone,

with either a circular or elliptical base (Fig. 2.154). These forms are frequently used in mechanical design, architecture, and geometric modeling.

You can access the tool by navigating to: Ribbon → HOME → MODELING →  CONE.

When the command is activated, AutoCAD first prompts you to define the base of the cone:

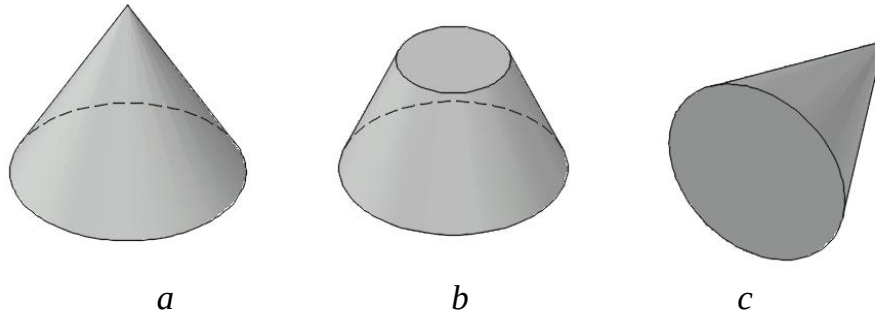


Fig. 2.154. Cones created with the Cone command:

a – full right cone; *b* – truncated right cone; *c* – full inclined cone

The available options are the same as those used for constructing a cylinder. Additionally, the radius of the top base of a truncated cone is specified.

Command: CONE

Specify center point of base or [3P/2P/Ttr/Elliptical]:

Specify base radius or [Diameter] <>:

Specify height or 2Point/Axis endpoint/Top radius] <>:T

Specify top radius <...>: – specify the radius of the top base of a truncated cone;

Specify height or [2Point/Axis endpoint] <>:

2.4.7.4. Sphere

The SPHERE command creates a spherical solid based on different input methods: by specifying the center and a radius or diameter, by three points on the surface, by two endpoints of the diameter, or by defining points of tangency to two 2D objects together with a radius (Fig. 2.155).

Command launch sequence: Ribbon → HOME tab → MODELING panel →  SPHERE command.

Command: SPHERE

Specify center point or [3P/2P/Ttr]: – specify the center point of the sphere, or choose one of the available construction options.

Specify radius or [Diameter] <>: – allows entering the radius, or selecting Diameter to define the sphere using its diameter value.

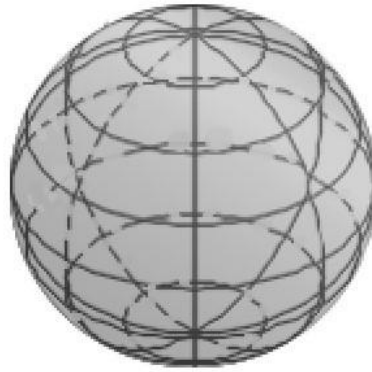


Fig. 2.155. Sphere model

3P option – sphere defined by three points. This method constructs a sphere that passes through any three points in 3D space. It is useful when the sphere must reference existing geometry.

Specify center point or [3P/2P/Ttr]: 3P

Specify first point:

Specify second point:

Specify third point:

2P option – sphere defined by diameter endpoints. This option constructs a sphere by specifying two opposite endpoints of its diameter.

Specify center point or [3P/2P/Ttr]: 2P

Specify first end point of diameter:

Specify second end point of diameter:

Ttr option – sphere tangent to two objects and defined by radius. This construction mode creates a sphere that is tangent to two selected 2D objects (lines, arcs, circles, etc.) and whose size is determined by a specified radius. It is especially effective when geometric constraints require an exact tangency relationship.

Specify center point or [3P/2P/Ttr]: T

Specify point on object for first tangent: – first object;

Specify point on object for second tangent: – second object;

Specify radius of circle <>: – specify the required radius.

Regardless of the chosen input method, the result is always a solid sphere, suitable for Boolean operations, sectioning, transformations, and further modeling workflows.

The 3P, 2P, and Ttr options provide advanced geometric control, allowing the sphere to be precisely integrated into complex models or aligned with existing elements.

2.4.7.5. Pyramid

The PYRAMID command allows constructing regular pyramids by specifying the center of the base, the radius (or diameter) of the circumscribed or inscribed circle of the base polygon, and the height. A truncated pyramid can also be created by using an additional option that requests the radius of the circle circumscribed around the top base (Fig. 2.156).

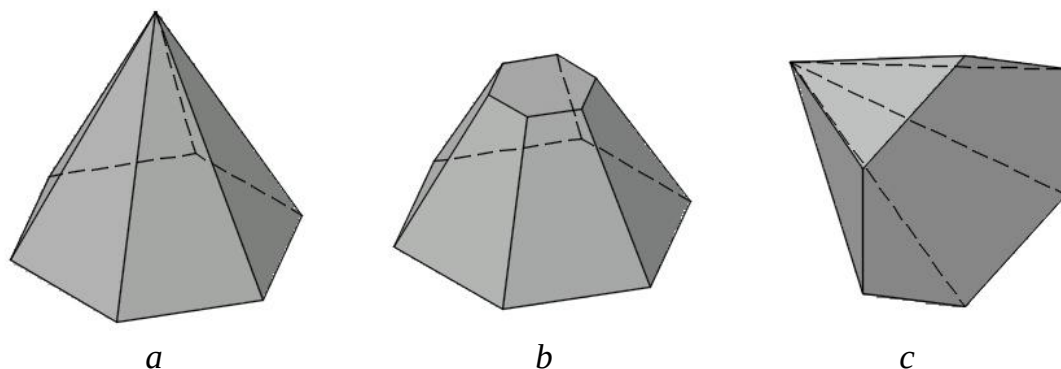


Fig. 2.156. Pyramids created with the Pyramid command:

a – full right pyramid; *b* – truncated right pyramid; *c* – full inclined pyramid

Accessing the command is performed through: Ribbon → HOME tab → MODELING panel →  PYRAMID command.

Command: PYRAMID

4 sides Circumscribed – this message indicates the current definition of the polygonal base – in this example, a four-sided base with a circumscribed circle.

Specify center point of base or [Edge/Sides]: – specify the center of the polygonal base, or choose one of the options.

Specify base radius or [Inscribed] <>: – enter the radius of the circumscribed circle, or select Inscribed to use the radius of the inscribed circle instead.

Edge option – constructs the polygonal base by specifying the length of a side, either numerically or by selecting two points.

Sides option – allows entering the number of sides of the polygonal base.

Specify height or [2Point/Axis endpoint/Top radius] <>: – the options are analogous to those described above for the *Cone* command.

2.4.7.6. Wedge

The WEDGE command creates a solid wedge whose edges are parallel to the X, Y, and Z axes of the current coordinate system. The available options correspond to the prompts used in the BOX command. When constructing a wedge, it is important to note that its base lies parallel to the XY plane, while the sloped face is positioned opposite the first base corner specified by the user. The height of the wedge is aligned with the Z-axis (see Fig. 2.157).

Access to the command is performed as follows: Ribbon → HOME tab → MODELING panel →  WEDGE command.

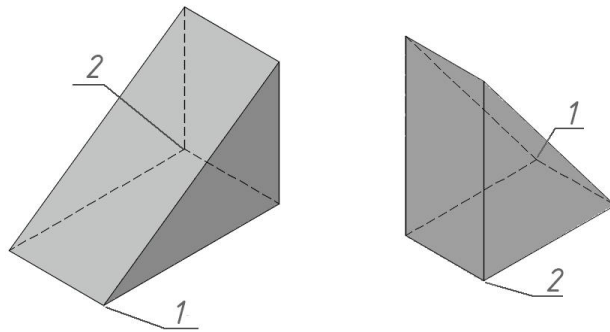


Fig. 2.157. Wedge shape depending on the order of specifying the base corners

Command: WEDGE

Specify first corner or [Center]: – specify the first corner of the rectangular base, or choose the Center option.

Center option – constructs the wedge by specifying the center of the rectangular base, followed by one of its corners.

Specify center: – define the center of the base;

Specify corner or [Cube/Length]: – specify a corner or choose an option;

Specify height or [2Point] <>: – enter the height or use 2Point to define height using two points in space.

Cube option – creates a wedge where all three primary dimensions (length, width, height) are equal.

Length option – Models a wedge by specifying the length, width, and height, similar to the BOX command. This option gives precise control over the rectangular base dimensions before the sloped face is generated.

When creating a wedge, the first specified base corner determines the position of the sloped face. To ensure correct orientation, select the corner along the intended slope direction or adjust the UCS beforehand.

2.4.7.7. Torus

Creation of a solid figure in the form of a torus (Fig. 2.158) is performed using the following command: Ribbon → HOME tab → MODELING panel →  TORUS command.

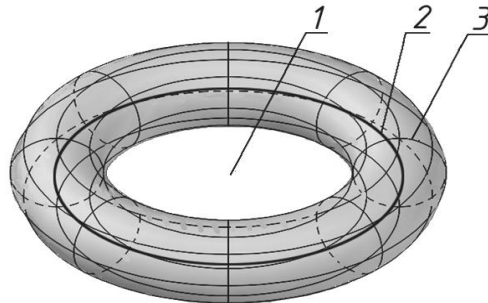


Fig. 2.158. Formation of torus elements:

1 – center; 2 – diameter of the path circle; 3 – generatrix

Command: TORUS

Specify center point or [3P/2P/Ttr]: – specify the center of the torus or choose one of the options (the options are described in subsection 2.4.7.2).

Specify radius or [Diameter] <>: – specify the radius or diameter of the path circle (the circle along which the center of the generatrix moves).

Specify tube radius or [2Point/Diameter]: – specify the radius of the generating circle (tube radius) or select an option to define it.

2.4.8. Projection Views of a 3D Object in the Layout Space

Generating projection views of 3D models in the layout space is performed using the VIEWBASE command. The command is launched as follows: Ribbon → HOME tab → right-side View panel → BASE menu →  FROM MODEL SPACE.

Command: VIEWBASE

Specify model source [Model space/File] <Model space>: – specify where the 3D object to be processed is located: in the model space of the current drawing or in an external file.

Select objects or [Entire model] <Entire model>: – select individual solids or all available geometry.

Enter new or existing layout name to make current or [?] <Layout1>: – enter the name of the layout or accept the default value shown in angle brackets by pressing Enter.

After this, AutoCAD switches to the specified layout space. The next prompts refer to view configuration:

Specify location of base view or [Type/Select/Orientation/Hidden lines/Scale/Visibility] <Type>: – specify the location of the main (base) view by clicking on the sheet, or choose one of the available options.

Type option – defines the number of views to place on the layout: only the base view, or the base view and all necessary projected views associated with it.

Enter a view creation option [Base only/base and Projected] <Base and Projected>: *B*

Select option – modifies the set of 3D objects to be represented on the layout.

Select objects to add or [Remove/Entire model/LAYout] <return to layout>: – definition of the objects in model space that should be added to the layout, or selection of the required option. *Remove* deletes the specified elements from the set displayed in the layout space; *Entire model* selects all existing 3D models; *Layout* returns to the layout space.

Orientation option – defines the projection direction.

Select orientation [Current/Top/Bottom/Left/Right/Front/Back/SWiso/SE iso/NE iso/NW iso] <Front>: – specifies the projection direction. The *Current* option uses the active view from model space, while *Top*, *Bottom*, *Left*, *Right*, *Front*, and *Back* generate standard orthographic projections from the corresponding directions. The options *SW iso*, *SE iso*, *NE iso*, and *NW iso* produce isometric views, where the projection directions are designated according to compass points (S – south, W – west, N – north, E – east).

Hidden lines option – controls the appearance of edges and shading.

Select style [Visible lines/Visible and hidden lines/Shaded with visible lines/sHaded with visible and hidden lines] <Visible and hidden lines>: – selects the display style of the generated view. The choices include visible lines only, visible and hidden lines, shaded representation with visible edges, and shaded representation with both visible and hidden edges.

Scale option – defines the drawing scale:

Enter scale <>: – enter the required scale value.

Visibility option – controls the visibility of intersection edges, tangent edges, and other display elements.

Editing the generated views becomes available after double-clicking on any view in the layout space. The corresponding modification tools appear on the ribbon (Fig. 2.159).

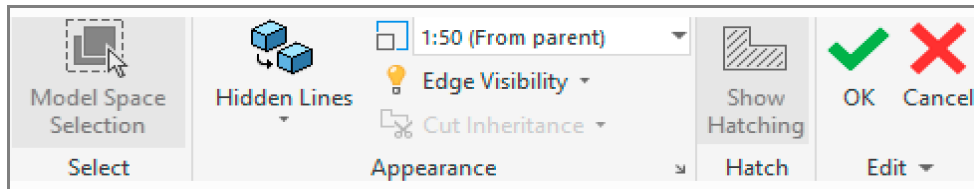


Fig. 2.159. View modification panel

As shown in the figure, the ribbon contains several options of the VIEWBASE command, including visualization settings and scale controls.

2.4.9. 3D Primitives and Development of Their Drawings

This example is intended to facilitate the acquisition of practical skills in constructing computer-based three-dimensional geometric models and preparing their corresponding engineering drawings. Variants of individual assignments are provided in Appendix B.3.

The purpose of the work is to generate solid models using primitive modeling commands based on the given projections (Fig. 2.160), and subsequently to develop the necessary drawing views – front, top, left, and axonometric – within the layout space of the system..

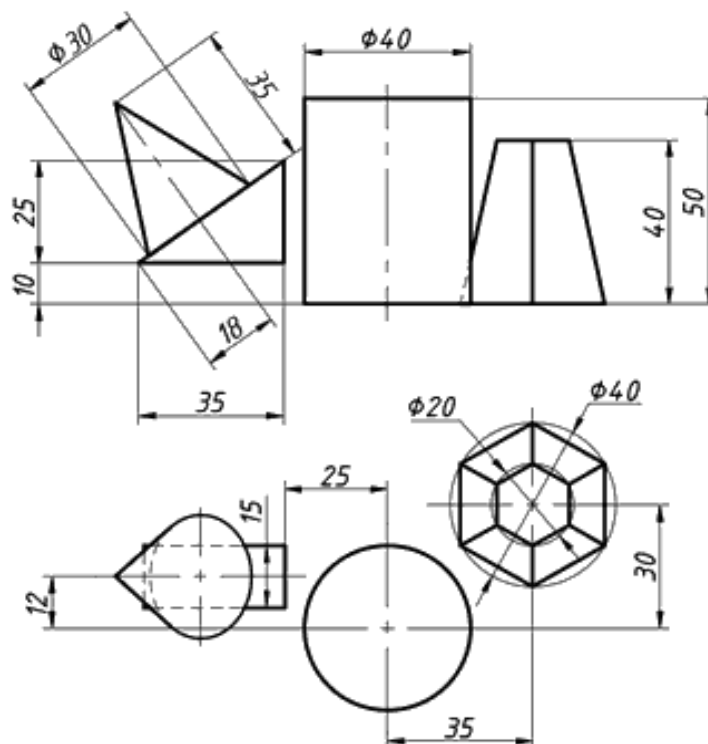


Fig. 2.160. Assignment “3D Primitives”

Sequence of construction

1. Study the assignment, identify the elementary geometric forms involved, and determine the required modeling operations.
2. Open the appropriate template file. Switch to the 3D Modeling workspace, which provides tools for both two-dimensional and three-dimensional constructions. Save the drawing under the name “3D primitives”.
3. Configure four viewports and assign to them the front, top, left, and axonometric views, respectively.
4. Select an initial primitive, for example, a cylinder. Construct it using the Cylinder command with a height of 50 mm and a base radius of 20 mm, placing the center at coordinates (0, 0, 0) (Fig. 2.161). All subsequent primitives should be positioned relative to this point.

For instance, construct a truncated pyramid using the Pyramid command, with the center of its lower base at (35, 30, 0). Specify 6 sides (option Sides), set the radius of the inscribed circle of the lower base to 20 mm (option Inscribed), define the upper base radius as 10 mm (option Top radius), and assign a height of 40 mm.

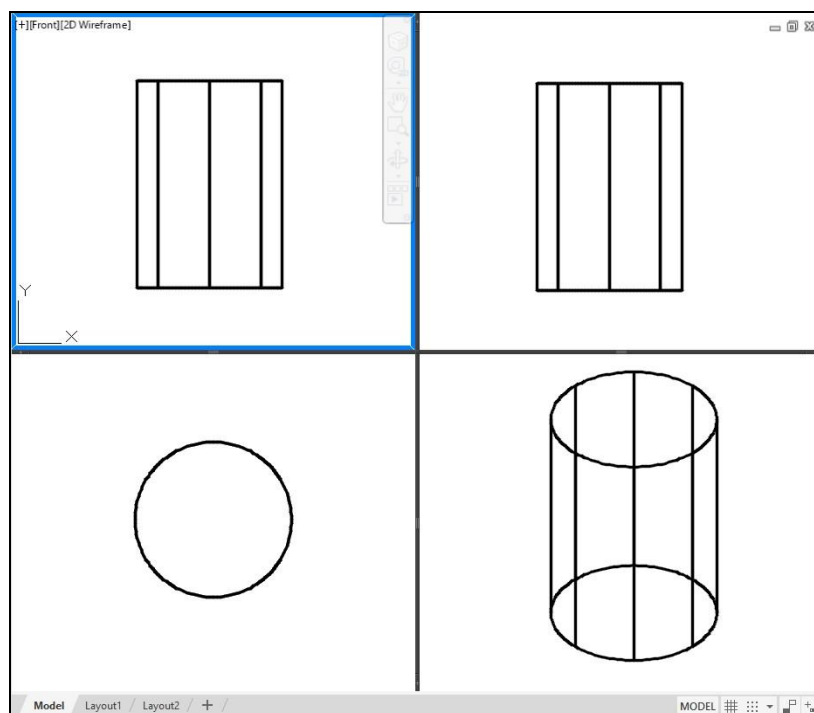


Fig. 2.161. Cylinder displayed in multiple viewports

5. Construct a wedge using the Wedge command (Fig. 2.162). Define the first and second base vertices at (0, 0, 0) and (–35, –15, 0). Assign a wedge height of 25 mm, then translate the solid by –25 mm along the X-axis (Fig. 2.163), by 12 mm along the Y-axis, and by 10 mm along the Z-axis.

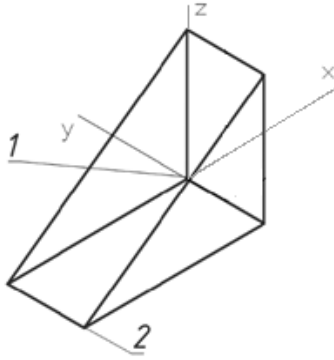


Fig. 2.162. Construction of the wedge

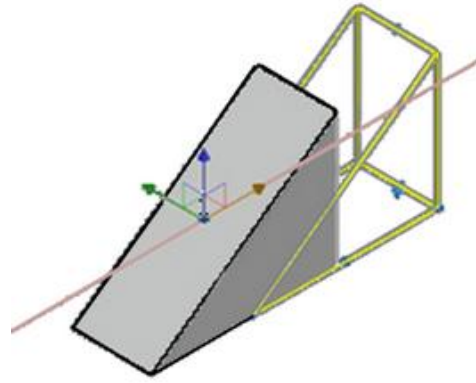


Fig. 2.163. Translation of the wedge

6. Create a cone using the Cone command. Its base must lie on the inclined face of the wedge; therefore, it is necessary to redefine the user coordinate system (UCS) as follows: Ribbon → HOME tab → COORDINATES panel →  Face command.

This aligns the UCS with the selected face of the 3D object (Fig. 2.164).

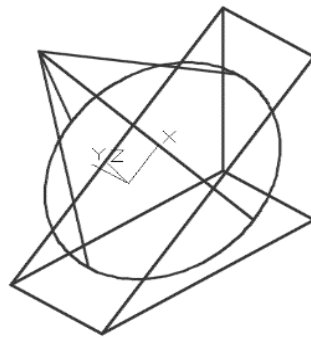


Fig. 2. 164. Cone positioned with its base on the wedge face

7. Construct the cone with a radius of 15 mm and a height of 35 mm, and then translate it into the required position in accordance with the dimension of 18 mm (see Fig. 2.160).

8. Generate the drawing views in the layout space using the following sequence: Ribbon → HOME tab → right View panel → BASE menu →  FROM MODEL SPACE.

Select the primitives in model space, specify the name of the layout on which the drawing will be prepared, and position the top, left, and axonometric views.

Note that upon transferring the model into the layout space, the system automatically creates the layers *MD_Hidden* (hidden lines) and *MD_Visible* (visible lines).

9. Add centerlines to the applied views and place dimensions on the *dim* layer.

10. Finalize the drawing (Fig. 2.165).

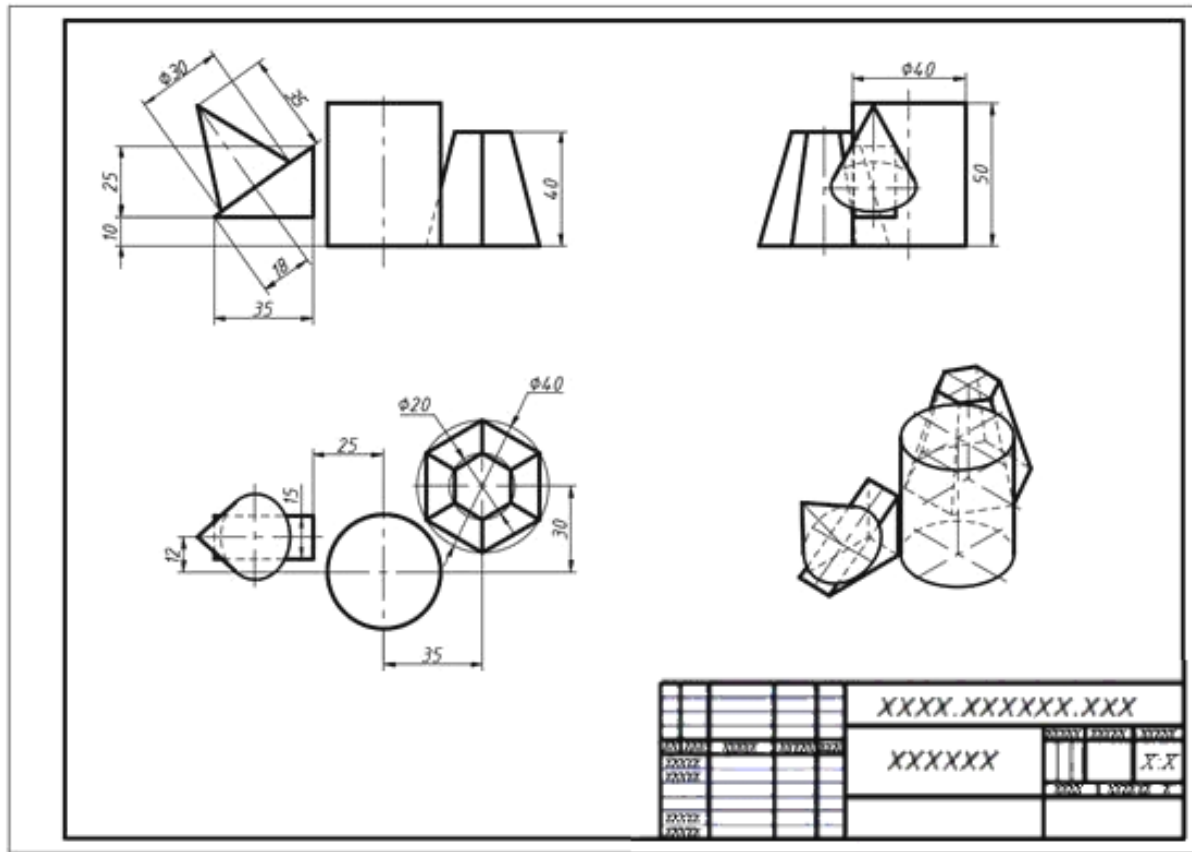


Fig. 2.165. Example of the “3D Primitives” assignment

2.4.10. Commands for Constructing Complex Objects

Solids may be constructed not only by means of the solid primitives described above but also through kinematic methods of geometric generation – such as extruding a closed profile along a specified direction or revolving it about a defined axis. When the profile is an open contour, these operations result in the creation of surfaces rather than solid bodies.

More complex geometries may additionally be obtained through the application of Boolean operations, including union, subtraction, and intersection, which allow combining or modifying solids to form intricate composite objects.

2.4.10.1. Extrude

The EXTRUDE command generates solid bodies by extruding closed profiles along a specified direction or path.

Access to the command is provided as follows: Ribbon → HOME tab → MODELING panel →  EXTRUDE command.

Command: *EXTRUDE*

Current wire frame density: ISOLINES=8, Closed profiles creation mode = Solid – a message displaying the current settings of the command.

Select objects to extrude or [MOde]: MO Closed profiles creation mode [SOlid/SUrface] <Solid>: _SO – defines the type of model to be created: a solid or a surface.

Select objects to extrude or [MOde]: – select the objects to be extruded.

Specify height of extrusion or [Direction/Path/Taper angle/Expression] <>: – specify the extrusion height or choose one of the available options.

As a result, the command constructs the required 3D figure (Fig. 2.166).

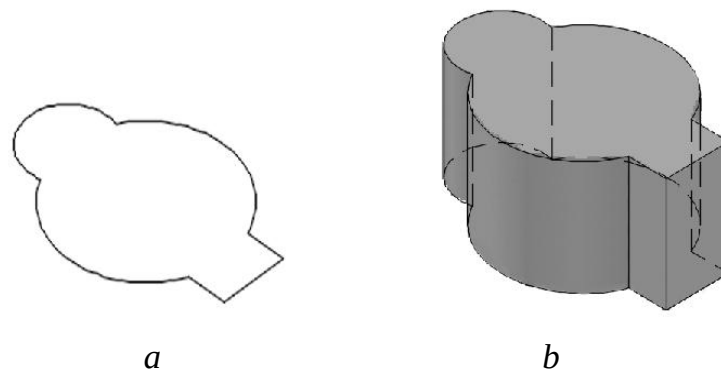


Fig. 2.166. Creating a model of the required height:
a – original profile; *b* – resulting solid

The Direction option generates the object by translating the profile along a direction and distance defined by two points (Fig. 2.167).

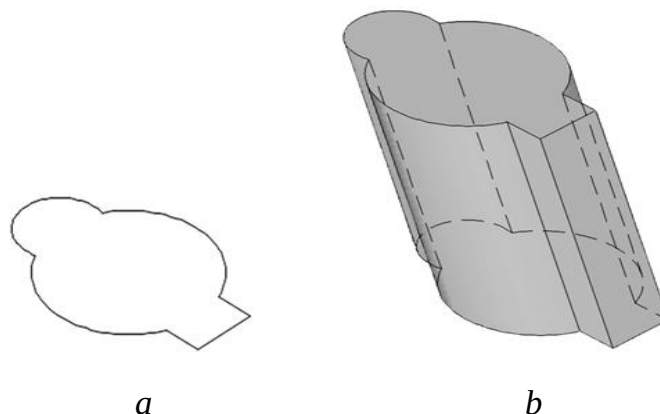


Fig. 2.167. Solid created using the Direction option:
a – original profile; *b* – resulting model

The Path option allows the profile (generatrix) to be extruded along a specified path, that is, a guiding curve (Fig. 2.168).

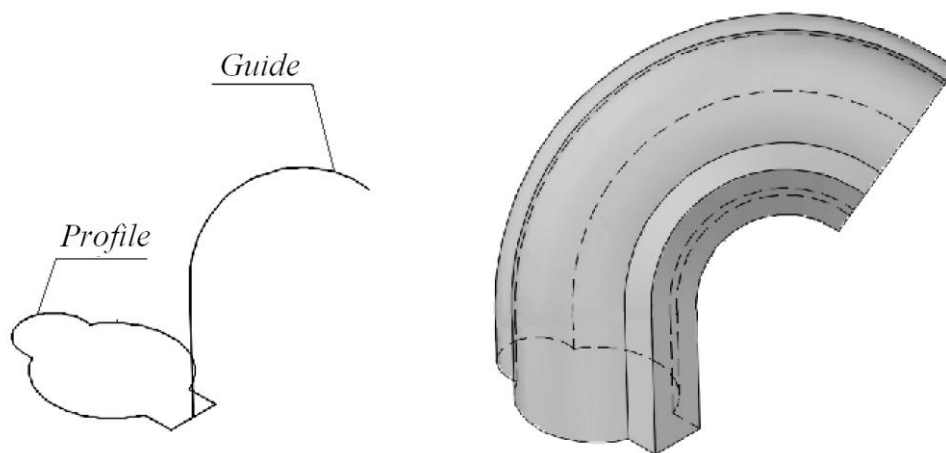


Fig. 2.168. Use of the Path option:
a – generatrix and guiding path; *b* – resulting solid

Line segments, circles, arcs, ellipses, polylines, splines, and similar entities may be used as extrusion paths.

The Taper angle option provides the construction of a solid that expands or tapers relative to the original profile (Fig. 2.169).

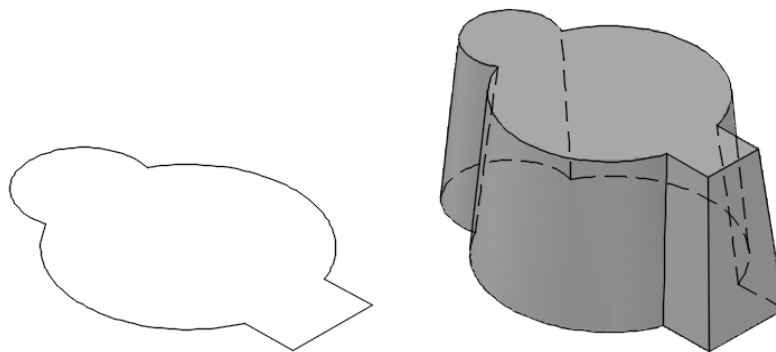


Fig. 2.169. Use of the Taper angle option

In response to the system prompts, the user must specify the taper angle and the height of the modeled figure. Permitted angle values range from -90° to $+90^\circ$. Positive values produce a tapered (narrowing) solid, whereas negative values result in an expanding one.

2.4.10.2. Revolve

The REVOLVE command enables the creation of solid objects by revolving a profile (closed or open contour) around a specified axis (Fig. 2.170).

Access to this command is provided as follows: Ribbon → HOME tab → MODELING panel →  REVOLVE command.

Command: *REVOLVE*

Current wire frame density: ISOLINES=8, Closed profiles creation mode = Solid – a message indicating the current settings of the command.

Select objects to revolve or [MDe]: _MO Closed profiles creation mode [SOlid/SUrface] <Solid>: _SO – specifies the type of model to be created: solid or surface.

Select objects to revolve or [MDe]: – select the objects to be revolved.

Specify axis start point or define axis by [Object/X/Y/Z] <Object>: – specify the starting point of the axis of revolution or choose one of the available options.

Specify axis endpoint: – specify the endpoint of the rotation axis.

Specify angle of revolution or [STart angle/Reverse/EXpression] <>: – enter the required angle of revolution.

Figures 2.170 and 2.171 illustrate solids created by revolving the original profile: Fig. 2.170 shows the result of a full 360° rotation, while Fig. 2.171 demonstrates the use of a different axis and a 270° rotation angle.

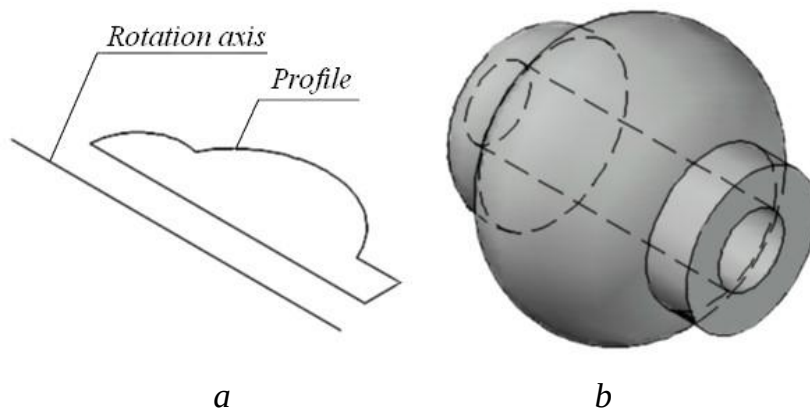


Fig. 2.170. Application of the Revolve command:
a – profile, axis, 360° rotation; *b* – resulting solid

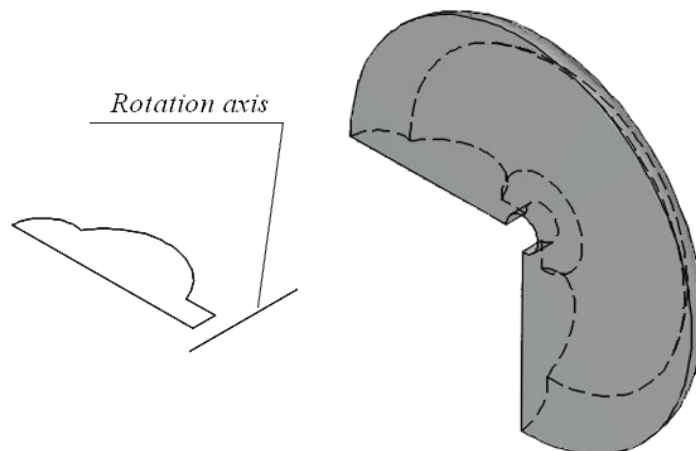


Fig. 2.171. Other rotation parameters

The Object option defines the axis of revolution using an existing line segment or a polyline segment.

The X, Y, and Z options perform the revolution around the corresponding X, Y, or Z axes of the current coordinate system.

2.4.10.3. Presspull

The PRESSPULL command creates solid models by pulling the boundaries of enclosed regions perpendicular to their defining plane (an alternative to the Extrude command). The profile may consist of separate objects that, when overlapping, form closed regions suitable for extrusion (Fig. 2.172).

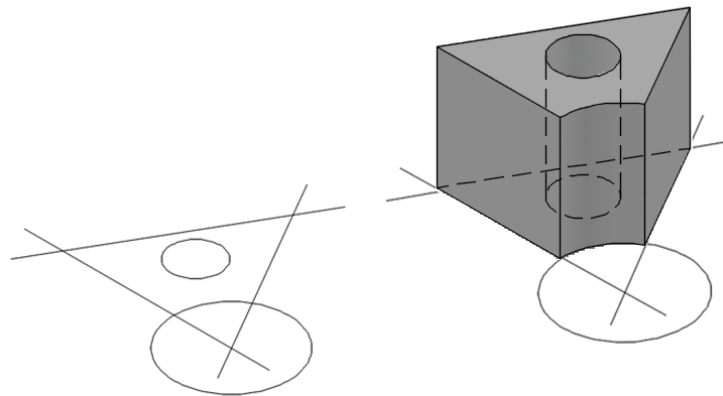


Fig. 2.172. Creating a solid using the Presspull command

These tools not only allow the creation of solids but also support their modification (Fig. 2.173).

Access to the command is provided as follows: Ribbon → HOME tab → MODELING panel →  PRESSPULL command.

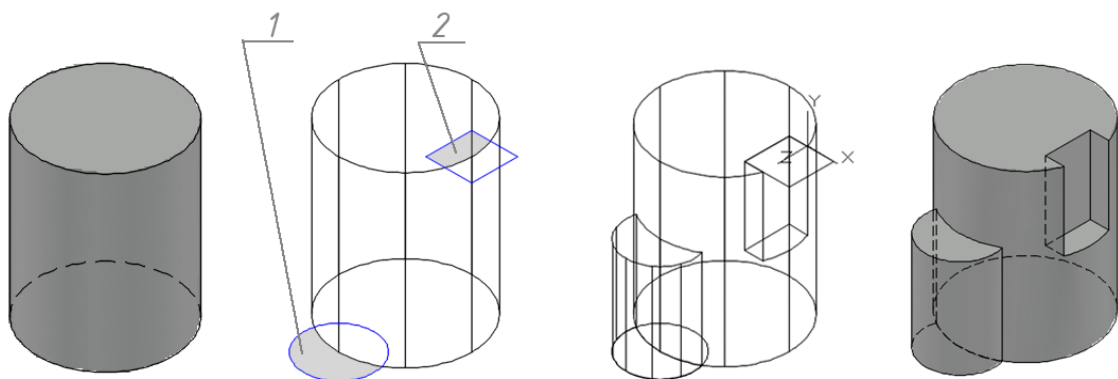


Fig. 2.173. Modifying a cylinder (1 and 2 – pull contours)

Command: *PRESSPULL*

Select object or bounded area: – specify a bounded region or a face.

Specify extrusion height or [Multiple]: – enter the extrusion height.

The command automatically repeats after each operation. To terminate the command, press Esc, Enter, or Spacebar.

2.4.11. Boolean Operations

Complex solids can be constructed more effectively by first analyzing the object and identifying the simple geometric forms from which it can be composed. This makes it possible to determine how the existing elements should be combined in order to obtain the required model.

Three Boolean operations are available for such constructions: union, subtraction, and intersection.

The Union command combines two or more solids into a single composite body.

The Subtract command removes from the selected solid the volumes occupied by other specified solids. This operation is commonly used to create openings by subtracting appropriately positioned cylindrical or prismatic shapes.

The Intersect command produces a solid that represents only the common volume shared by two or more solids.

The tools for Boolean operations are available on the Solid tab of the ribbon, within the Boolean panel.

2.4.11.1. Union

The UNION command creates a single solid model from several existing solids by merging them into one object (Fig. 2.174). It is not possible to unite a region with a solid body.

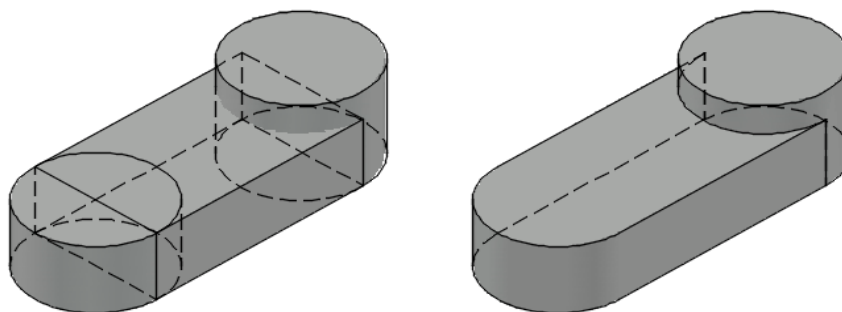


Fig. 2.174. Union of three solids using the Union command

Access to this tool is provided as follows:

Ribbon → SOLID tab → BOOLEAN panel →  UNION command.

Command: UNION

Select objects – select the solids that are to be unified.

2.4.11.2. Subtract

The SUBTRACT command forms a solid by subtracting the volumes of other models from the first selected one. The command is launched as follows:

Ribbon → SOLID tab → BOOLEAN panel →  SUBTRACT command.

Command: SUBTRACT

Select objects: – select object 1, from which the subtraction will be performed (Fig. 2.175).

Select solids, surfaces, and regions to subtract .. – a reminder of the allowable object types.

Select objects: – select object 2, whose volume is to be removed.

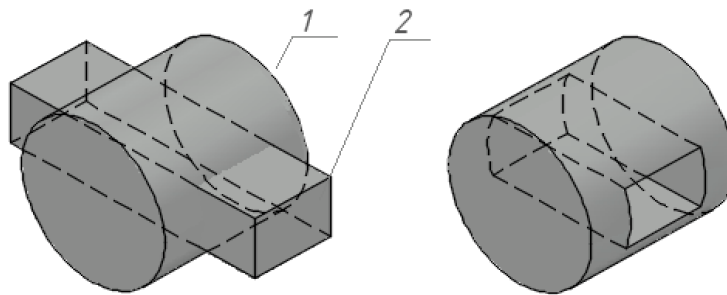


Fig. 2.175. Subtraction of solids

A particular feature of the command is that the first selected object is the one from which all subsequent selected objects will be subtracted. To proceed to selecting subtraction objects, press Enter.

2.4.11.3. Intersect

The INTERSECT command constructs a solid by generating the volume common to the selected objects (Fig. 2.176). The original solids are not preserved after the operation.

Access to the tool is provided as follows:

Ribbon → SOLID tab → BOOLEAN panel →  INTERSECT command.

Command: INTERSECT

Select objects: – select the objects and press Enter.

The Intersect operation is particularly useful for deriving precise geometric relationships between overlapping solids, as it retains only the exact common

volume shared by them. This allows the creation of complex shapes whose definition cannot be achieved by primitive modeling alone.

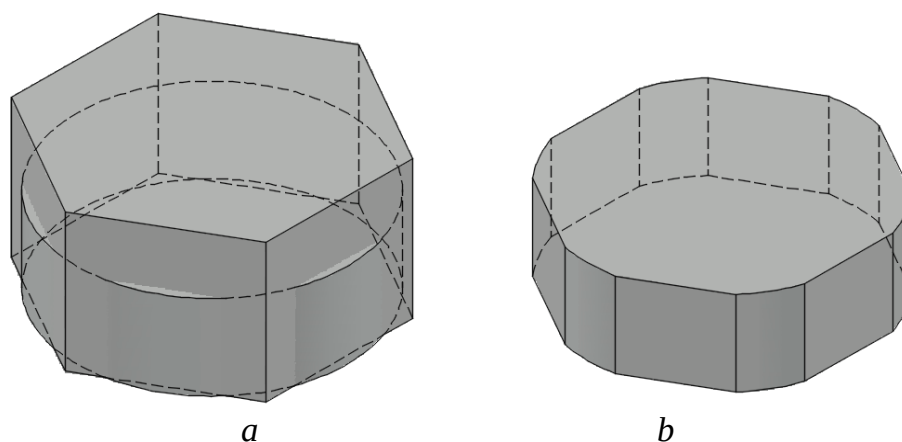


Fig. 2.176. 3D model obtained as the common volume of a circular cylinder and a regular prism with a hexagonal lateral surface:
a – original objects; *b* – result of the operation

2.4.12. Editing 3D Objects

This section of the textbook examines several methods for modifying three-dimensional geometric models. In particular, it concerns tools such as the Gizmo, which provides functions for moving, rotating, and scaling objects, as well as the Mirror, Chamfer, and Fillet commands.

2.4.12.1. Using the Gizmo

The Gizmo – a set of graphical, dynamic manipulator guides – serves as an interactive control mechanism for moving, rotating, and scaling three-dimensional objects with respect to the principal coordinate axes and planes. It provides a visual and intuitive interface for spatial transformations, enabling the user to specify the direction and magnitude of modifications directly within the drawing area.

By default, the Gizmo is displayed automatically when a 3D object is selected, provided that the object is shown using a visual style other than 2D Wireframe. In such cases, the Gizmo becomes the primary tool for precise manipulation of solids, offering axis-specific handles and planar control regions that highlight and extend as the cursor hovers over them, thereby indicating the active transformation direction.

The icons for  3D Move,  3D Rotate, and  3D Scale are available on the ribbon under the Home and Solid tabs: Ribbon → HOME (or SOLID) tab → MODIFY (or SELECTION) panel. The corresponding menu is shown in Fig. 2.177.

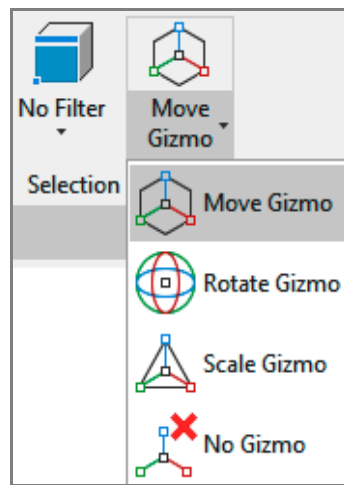


Fig. 2.177. Gizmo tools

Let us consider the  3D Move command.

Select objects: – select the objects to be moved; press Enter to finish the selection. The Move Gizmo appears (Fig. 2.178), allowing the direction of motion to be specified easily along coordinate axes or within a coordinate plane. When the cursor is hovered over an axis, the axis is visually extended, indicating the direction of displacement.

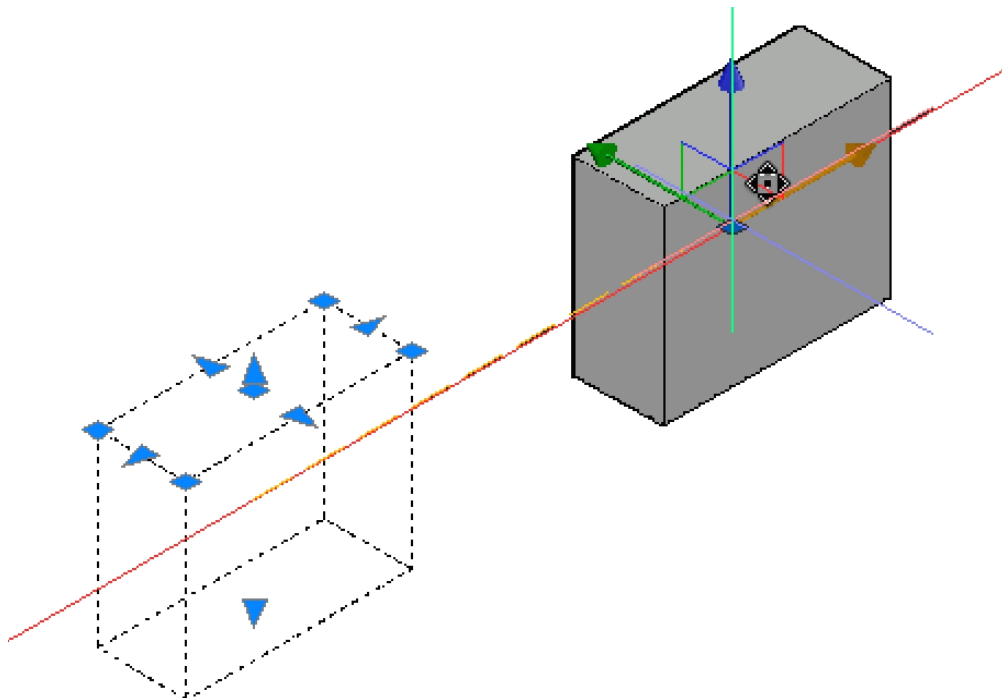


Fig. 2.178. Example of modifying the position of a prism

Specify base point or [Displacement] <Displacement>: – specify the base point of the move.

Specify move point or [Base point/Copy/Undo/eXit]: – define the displacement distance along the chosen axis or specify the new location directly in the drawing area by clicking with the left mouse button. The object is then moved in the selected direction by the indicated amount.

When executing the  3D Rotate command, the Gizmo is used to define the axis of rotation. The direction of this axis is established by selecting one of the coordinate planes, which is displayed as a circular projection; the rotation is then performed around the axis perpendicular to the chosen plane (Fig. 2.179).

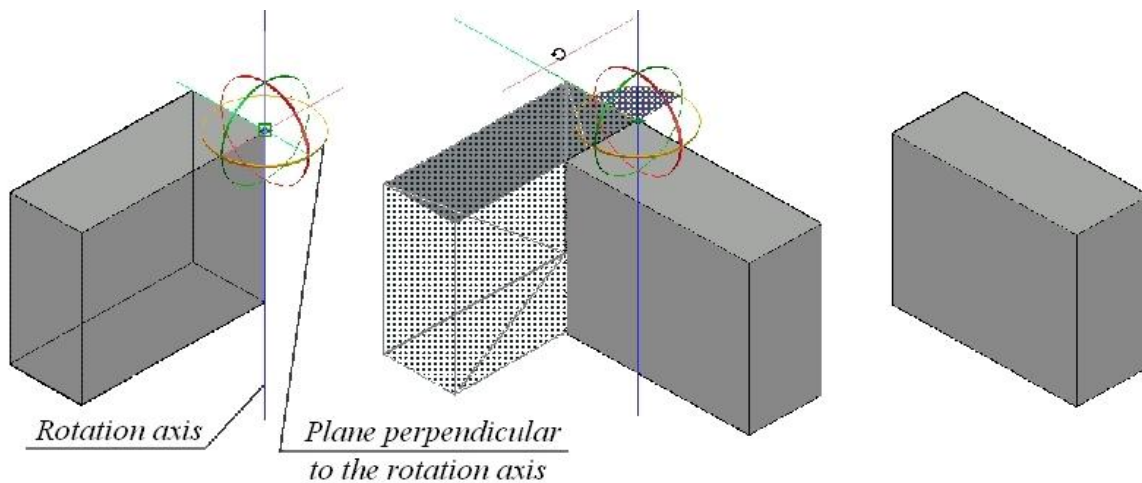


Fig. 2.179. Rotating a prism using the Gizmo

The  3D Scale command applies uniform scale factors to all dimensions of a solid model. These factors may be specified numerically or as a ratio of two selected lengths.

When an object is selected in a 3D visual style, clicking the Gizmo marker with the left mouse button opens a context menu where the required command may also be activated (Fig. 2.180).

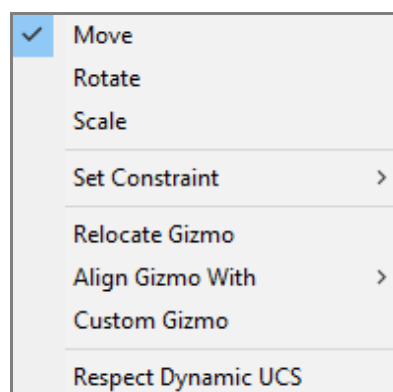


Fig. 2.180. Gizmo context menu

The context menu item *Align Gizmo With* contains a drop-down list of alignment options: with the World Coordinate System (WCS), with the current User Coordinate System (UCS), or with a selected object.

2.4.12.2. 3D Mirror

This tool operates in a manner analogous to the Mirror command used in 2D space; however, in three-dimensional modeling the reflection is carried out with respect to a plane rather than a line. The user defines this mirror plane explicitly, which makes it possible to produce accurate mirrored copies of complex 3D geometries in any spatial orientation.

Figure 2.181 demonstrates the outcome of the operation performed with the 3 points option, where the plane of symmetry is established by three user-specified points, and the original object is preserved while its mirrored counterpart is generated.

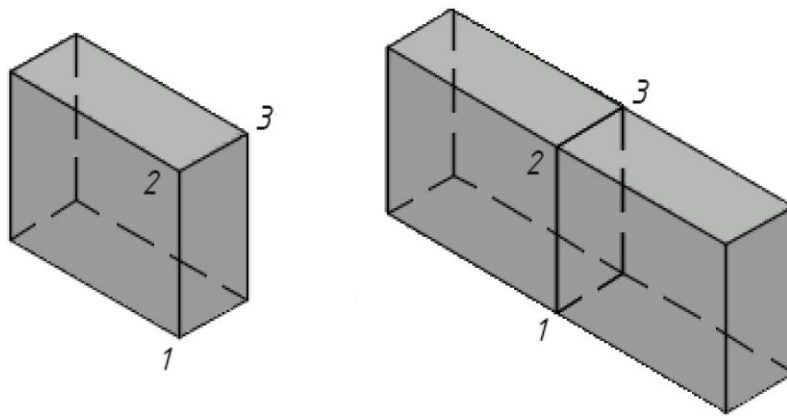


Fig. 2.181. Mirroring with the original object preserved

Access to the command is provided as follows: Ribbon → HOME tab → MODIFY panel →  3D MIRROR command.

Command: MIRROR3D

Select objects: – select the objects to be mirrored.

Specify first point of mirror plane (3 points) or [Object/Last/Zaxis/View/XY/YZ/ZX/3points] <3points>: – specify the first of three points that define the mirror plane, or choose an alternative method for defining this plane:

- Object uses a 2D object as the reference.
- Last applies the plane used in the previous invocation of the command.
- Zaxis defines the plane via a normal vector.
- View uses the current view direction.

– XY, YZ, ZX correspond to planes parallel to the respective planes of the current UCS.

Delete source objects? [Yes/No] <N>: – specifies whether the original object should be deleted.

2.4.12.3. Chamfer Edge

The Chamfer Edge command is widely used in the modeling of engineering components and mechanical parts. It creates chamfer elements along the edges formed by the intersection of adjacent faces of a solid. Chamfers are frequently applied to remove sharp edges, improve part manufacturability, enhance structural strength, or comply with standard design requirements (Fig. 2.182).

In 3D modeling, chamfers are constructed directly on the solid's edges, and the resulting geometry updates automatically to maintain consistency with the surrounding faces.

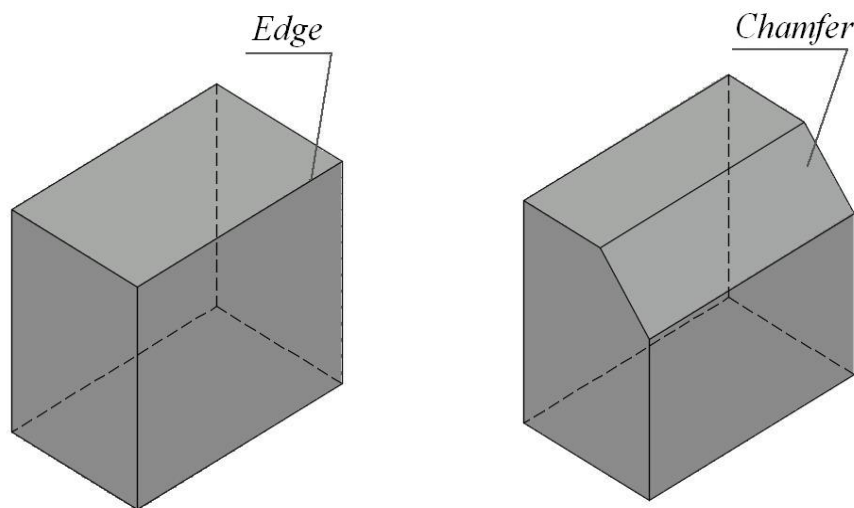


Fig. 2.182. Example of chamfer creation

Access to the command is provided through:

Ribbon → SOLID tab → SOLID EDITING panel →  CHAMFER EDGE command.

Command: CHAMFEREDGE Distance1 = 1.0000, Distance2 = 1.0000 – This message reports the current values of the two chamfer distances, which represent the lengths of the two legs of the chamfer measured along the adjacent faces.

Select an edge or [Loop/Distance]: – Here the user selects the edge to be chamfered or chooses an option.

The *Loop* option applies the chamfer to all edges bounding a selected face, which is convenient when a consistent chamfer is required around the entire perimeter of a feature.

The *Distance* option allows redefining the chamfer parameters by specifying the lengths of the two legs.

Specify Distance1 or [Expression] <..>: – enter the first distance value (leg 1).

Specify Distance2 or [Expression] <..>: – enter the second distance value (leg 2).

Select another edge on the same face or [Loop/Distance]: – continue applying chamfers to additional edges as needed.

In AutoCAD, chamfer creation is associative: the chamfered edges automatically adjust to the connected geometry, maintaining clean intersections between modified faces. This makes the tool highly effective when refining 3D models prior to fabrication or detailed documentation.

2.4.12.4. Fillet Edge

The Fillet Edge command is likewise used in the modeling of various components and generates design elements known as edge fillets. A fillet is a rounded transition between adjacent faces of a solid body, applied to eliminate sharp edges, improve part strength, increase safety, enhance manufacturability, or meet design standards (Fig. 2.183).

In contrast to 2D fillets, which modify line segments or curves, the 3D Fillet Edge tool applies a smooth, curved surface along the edge of a solid, producing a physically accurate blending of adjacent faces.

Access to the command is provided through:

Ribbon → SOLID tab → SOLID EDITING panel →  FILLET EDGE command.

Command: FILLETEDGE

Select an edge or [Chain/Loop/Radius]: – specify the edge to be filleted or choose one of the available options.

Press Enter to accept the fillet or [Radius]: – press Enter to accept the default radius or select the Radius option.

Enter fillet radius or [Expression] <>: – enter the required fillet radius.

The Chain option selects a series of connected edges, allowing fillets to be applied along an continuous path – useful for pipes, frames, or contour-following transitions.

The diagram consists of two 3D perspective drawings of rectangular blocks. The left block has sharp corners, and a line points to one of them with the label "Edge". The right block has rounded corners, and a line points to one of them with the label "Fillet".

2.4.13. Example of Creating a Solid Model from Its Projections

151

The task is to construct a 3D model based on the available front and top views, along with the sections that reveal the internal geometry of the part.

Sequence of construction

1. Examine the drawing to determine which basic geometric forms constitute the 3D model and identify their dimensions. Analyze the commands required for generating individual elements and establish the relationships among them.

2. Open the appropriate template file. Switch to the 3D Modeling workspace. Save the model under the name “Housing” (“Korpus”).

3. Construct the right section of the part using the Cylinder and Box commands to create a cylinder and a rectangular parallelepiped, respectively. Combine these primitives using the Boolean Union operation (Figs. 2.185, a and 2.185, b). Generate the inner cavity (wall thickness 10 mm) using the Shell command (Fig. 2.185, c).

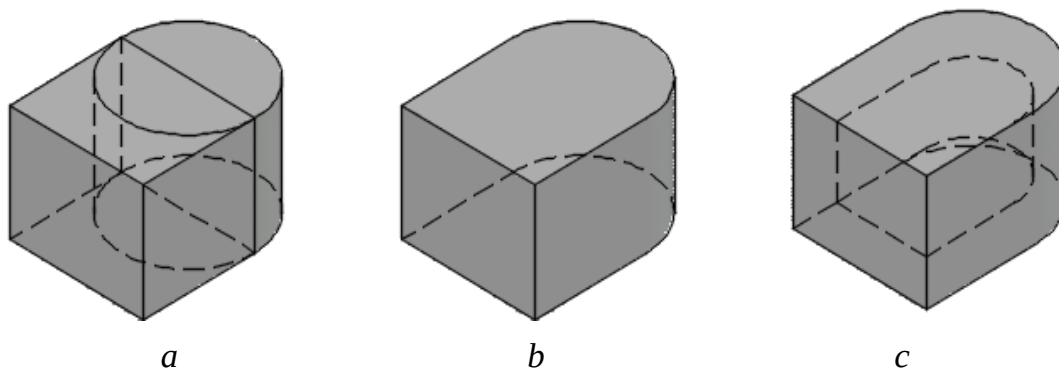


Fig. 2.185. Construction of the right section of the housing:
a – cylinder and parallelepiped; b – union of primitives;
c – interior cavity created using Shell

4. In the central region of the housing, construct a horizontal cylinder with a base diameter of 56 mm and a length of 60 mm. Unite it with the right section. Create a through hole with a diameter of 40 mm using the Presspull command (Fig. 2.186).

5. Remove the internal wall formed between the hole and the cavity using Solidedit (option Delete), as shown in Fig. 2.187.

6. Construct a slot using the Presspull command (Fig. 2.188).

7. Form the left section of the housing by creating a horizontal cylinder with a diameter of 32 mm and a length of 50 mm, located 65 mm from the previously created cylinder (Fig. 2.186).

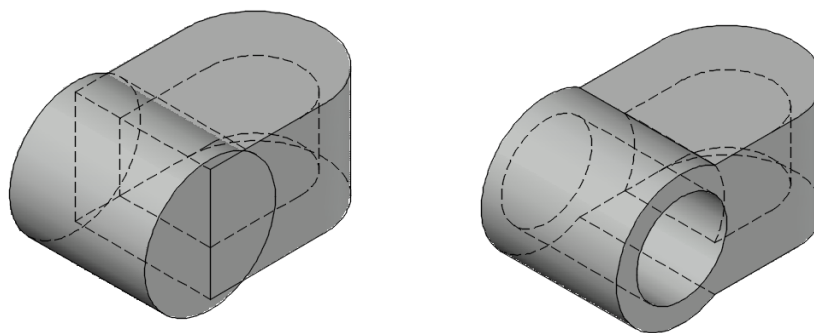


Fig. 2.186. Formation of the central section of the housing

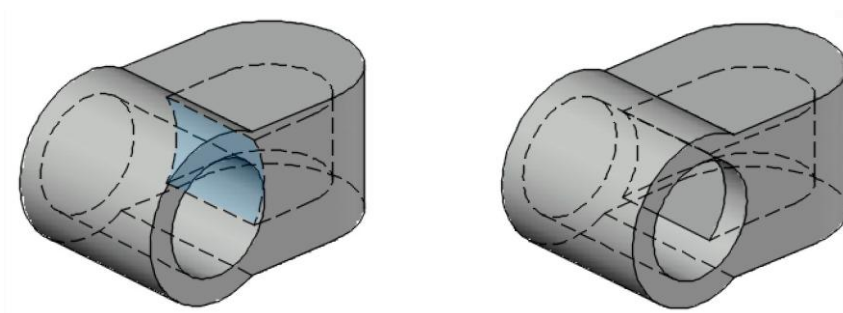


Fig. 2.187. Removal of the internal wall

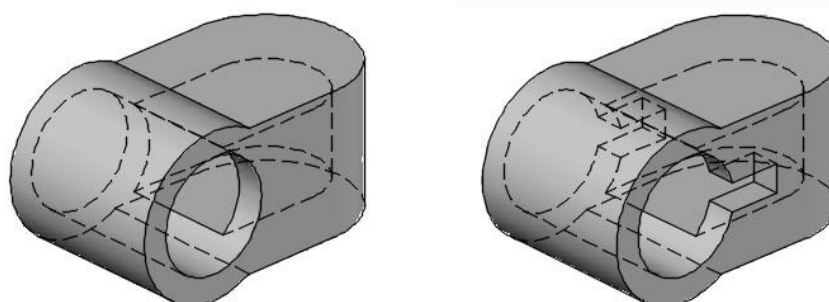


Fig. 2.188. Creation of the slot

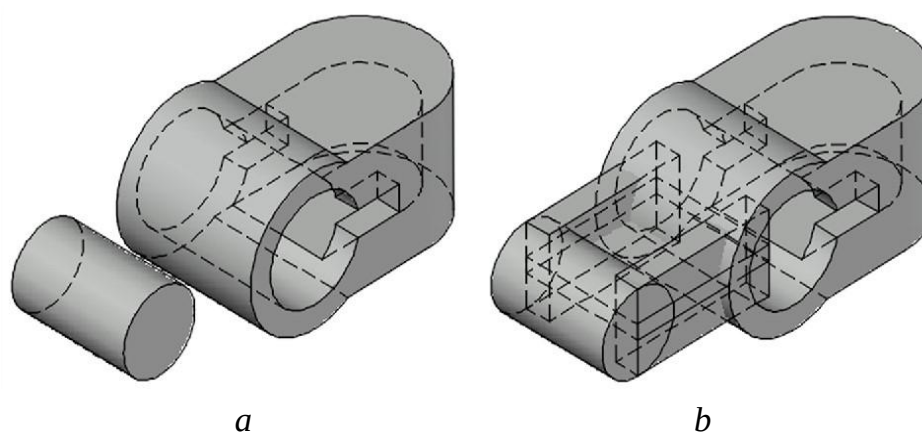


Fig. 2.189. Formation of the left section of the housing:
a – cylinder positioned 65 mm from the central one;
b – reinforcing ribs

8. Construct the vertical and horizontal reinforcing ribs (Fig. 2.189, b).
9. Combine all components of the housing using the Union command.
10. Create the required cylindrical features (a boss on the right side, a horizontal hole on the left side, and vertical holes), as well as the conical chamfers shown in Fig. 2.190. Remove unnecessary segments of the reinforcing ribs.

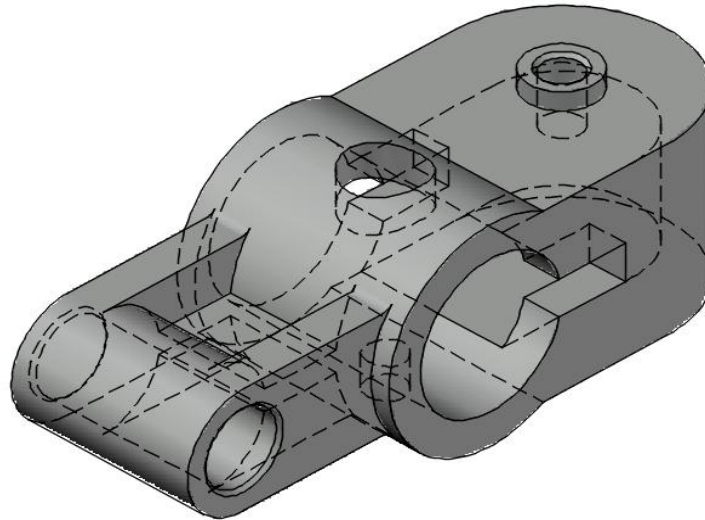


Fig. 2.190. Solid model of the housing

The illustration above demonstrates the completed task of developing a solid model of the part based on its projection views.

2.5. Conclusions

This chapter has presented the foundations of computer-aided geometric modeling in AutoCAD. The material covered essential aspects of working with the software, including the user interface, coordinate systems, units of measurement, layer management, and viewport control. Considerable attention was given to both 2D and 3D modeling techniques and to the properties and operational principles of the corresponding geometric entities.

The chapter provided detailed descriptions of the tools used to construct fundamental geometric primitives—points, line segments, circles, ellipses, polygons, splines, boxes, cylinders, cones, spheres, pyramids, and others—as well as methods for modifying these objects, including the application of parametric dependencies.

In addition, key tools for preparing engineering drawings were considered, such as linetypes and lineweights, text and dimension styles, and hatching

methods. Methods for creating and editing solid models, including chamfers, fillets, Boolean operations, and kinematic constructions, were also discussed.

Practical examples illustrated how these tools and techniques are applied when developing solid models from their projection views.

Mastering the material of this chapter enables students to design components, assemblies, and technical devices of high complexity and quality during the execution of course and diploma projects. The stated objectives are achieved through the effective use of modern computer-aided design technologies, particularly the AutoCAD system.

Control Questions

1. Characterize the primary purpose of the AutoCAD system.
2. Describe the main interface elements of the software presented in this chapter of the textbook.
3. Name and explain the methods of entering coordinates.
4. Explain the techniques for selecting objects and provide examples of their use.
5. List the main commands for creating two-dimensional primitives and describe their practical application.
6. Provide a concise overview of the tools for editing 2D objects.
7. Analyze the available workspaces in the AutoCAD system.
8. Justify the necessity of creating template files that include specialized layers, linetypes and lineweights, text and dimension styles, and other settings.
9. Highlight the advantages of parametric geometric models.
10. Describe the procedure for constructing projection views and the tools used for this purpose.
11. List the commands for creating 3D primitives and explain their practical use.
12. Explain the main stages of generating projection views of an object in the layout space.
13. Discuss the commands for kinematic construction of complex solids.
14. Provide an overview of the purpose of Boolean operations.
15. Analyze the tools for editing three-dimensional figures in AutoCAD.
16. Describe the procedure for creating a solid model based on its projections.

3. AUTOMATION OF CONSTRUCTIONS BY MEANS OF VISUAL BASIC FOR APPLICATIONS

As already noted, the main goal of this textbook is to ensure the efficient mastering by students of geometric modeling in modern computer-aided design systems at a more professional level than that of an ordinary user. This is achieved not only by presenting, in the previous chapter, the functional capabilities of a particular system, but also by covering general information on computer-aided engineering graphics (Chapter 1) and the relevant software tools for automated constructions in this part of the textbook, which includes the description of the necessary mathematical foundations of shape generation.

In addition, the textbook emphasizes the importance of understanding both theoretical principles and practical implementation tools, enabling students to build a coherent system of knowledge. Such an integrated approach trains future specialists not only to perform constructions mechanically but also to analyze and optimise them. It also develops algorithmic thinking, which is essential for working with modern engineering environments and for automating repetitive tasks in CAD systems.

Such an approach provides solid knowledge that can be productively applied in practice during geometric modeling of technical objects, for example, in the course and diploma design process. The provided information on computer-aided shape generation contributes to successfully mastering other, more complex CAD systems than AutoCAD, such as SolidWorks, CATIA, etc. The relevance of these tasks is driven by the high requirements for modern technical products and for the qualification level of specialists who develop them.

3.1. Visual Basic for Applications in AutoCAD

The Visual Basic environment (Fig. 3.1) is launched from the AutoCAD drop-down menu using the sequence *Tools* → *Macro* → *Visual Basic Editor*. Detailed help on the capabilities of Visual Basic can be accessed through the *Help* item or by pressing F1 (Fig. 3.2). Information may be searched either through its hierarchical structure or by using relevant keywords. Additional reference materials can be found online.

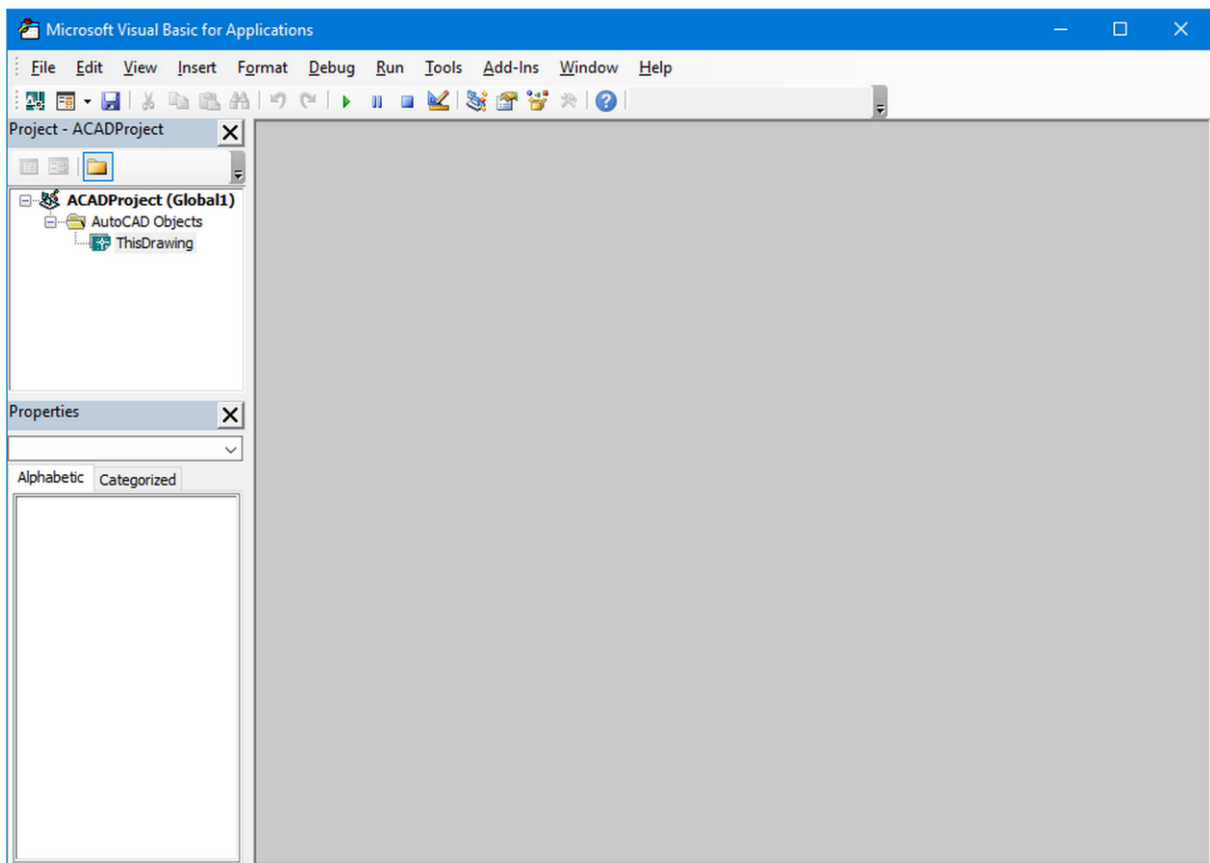


Fig. 3.1. Visual Basic Environment

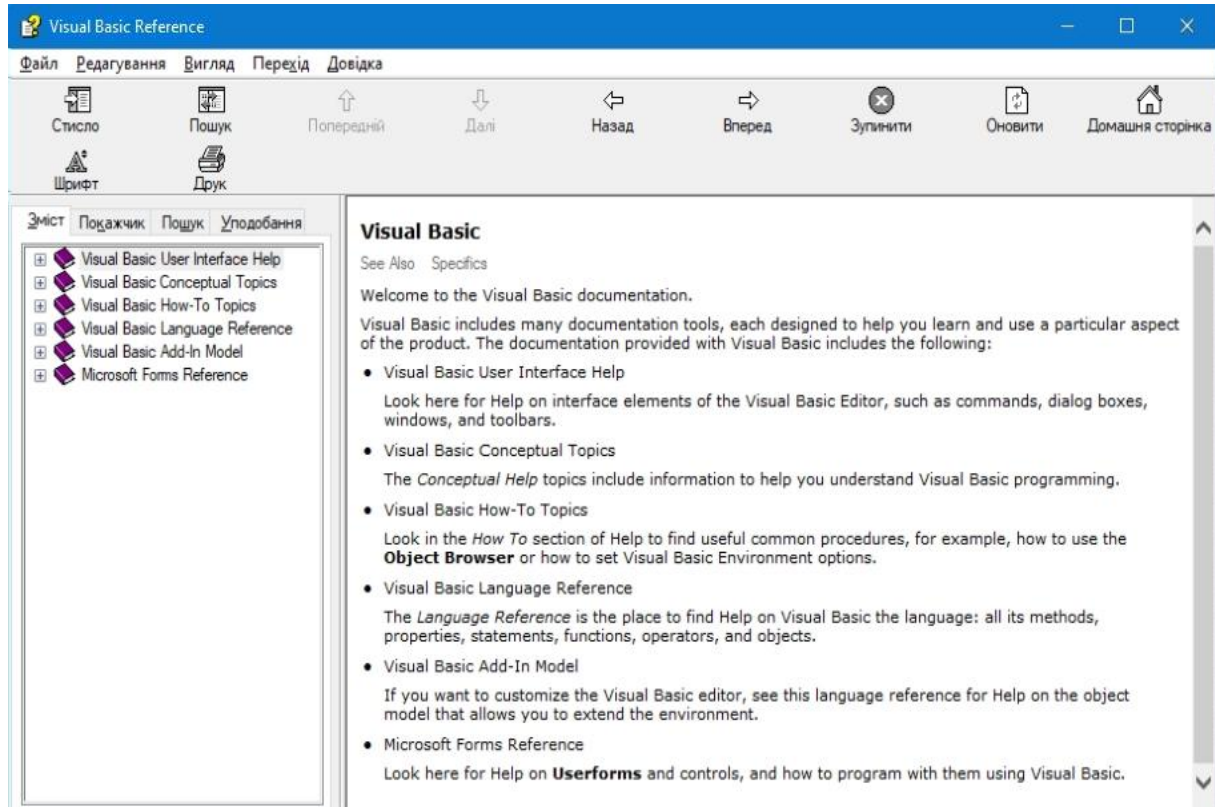


Fig. 3.2. Visual Basic Help

Access to macro management in AutoCAD (Fig. 3.3) is provided through *Tools* → *Macro* → *Macros...* or by pressing Alt+F8.

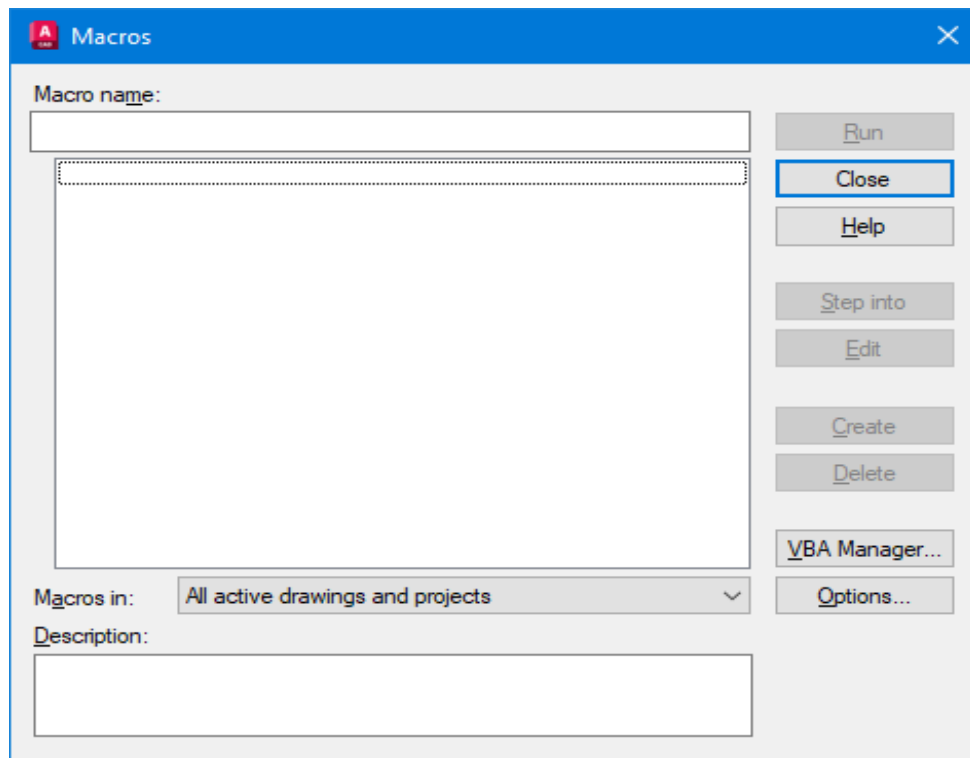


Fig. 3.3. Macro Management Menu

When creating a new macro, its name is specified (Fig. 3.4), the Create button is pressed, and the desired location is selected.

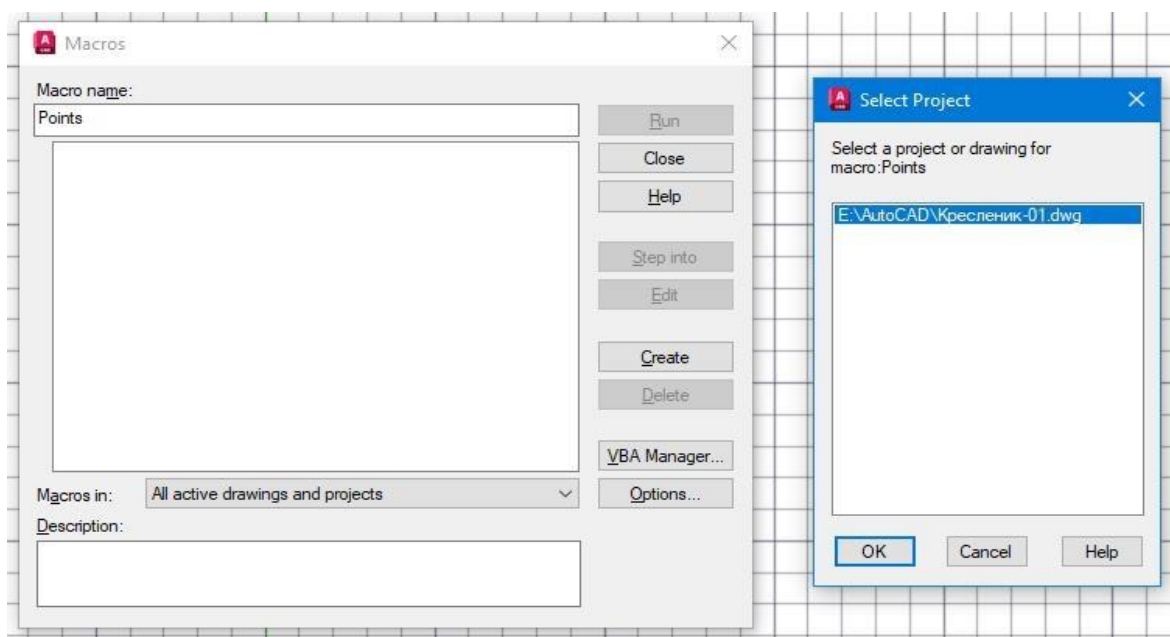


Fig. 3.4. Creating a New Macro

After that, the Visual Basic editor automatically opens with a new empty macro (Fig. 3.5).

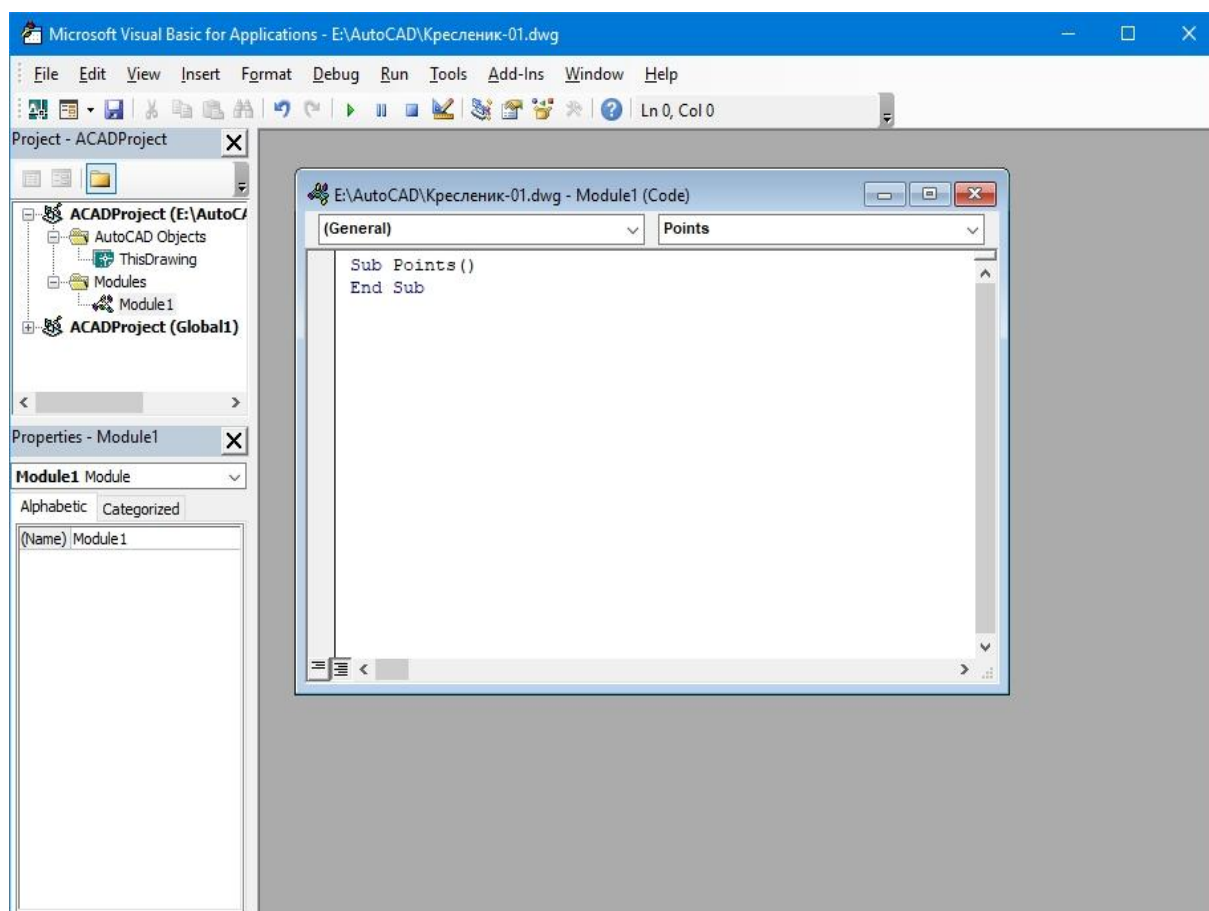


Fig. 3.5. Empty Macro

Next, the required program text is entered – an example is shown in Fig. 3.6. The macro is executed from within the Visual Basic environment using the Run menu or by pressing F5. This opens the window displayed in Fig. 3.7.

A similar result (Fig. 3.8) can be achieved directly in AutoCAD by selecting *Tools* → *Macro* → *Macros...* in the drop-down menu.

In both cases, the macro is executed by clicking the Run button.

The result of the program (Fig. 3.9) is the creation of three points with the required coordinates and display parameters.

A detailed explanation of the code is provided below.

Here we focus only on the interface and general workflow for automated geometric constructions using Visual Basic for Applications in AutoCAD system.

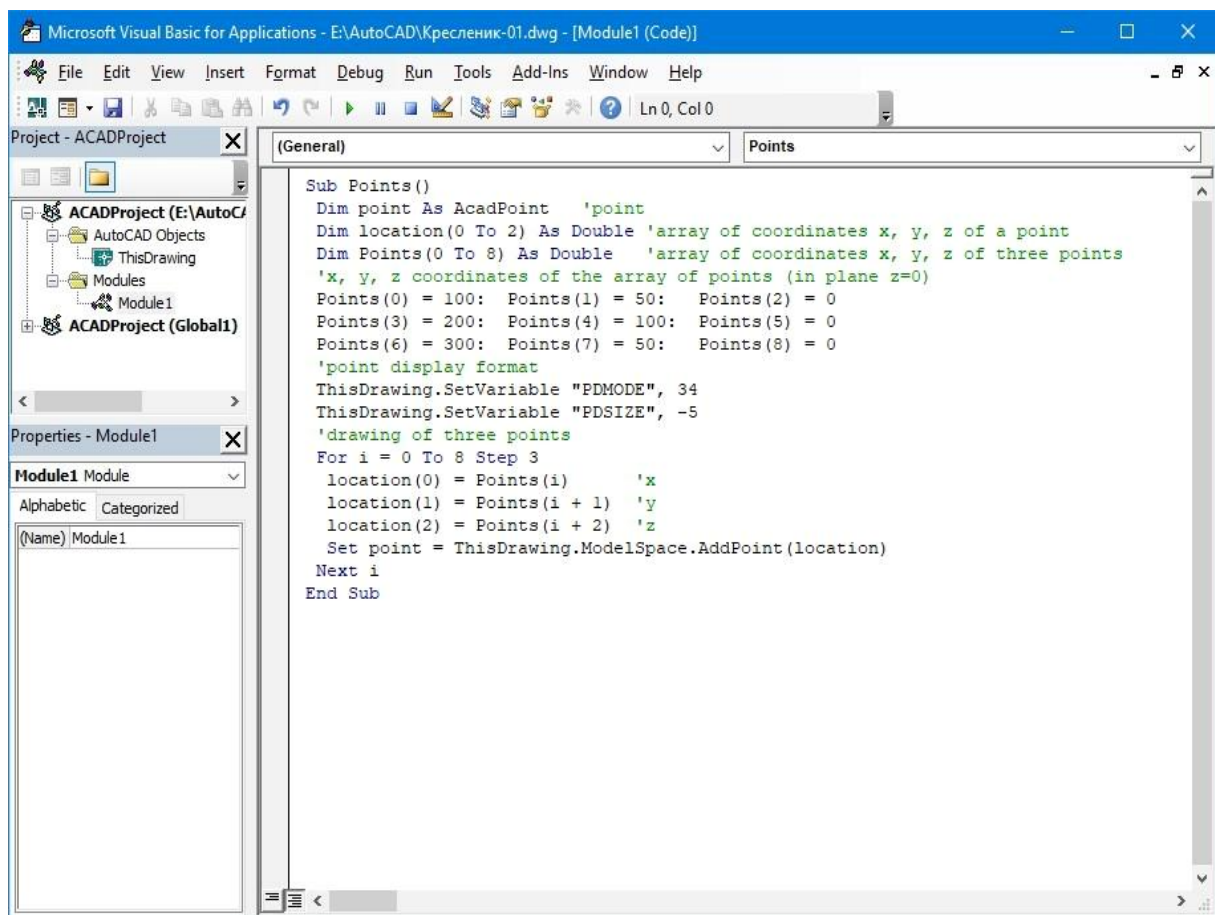


Рис.3.6. Example of a Macro

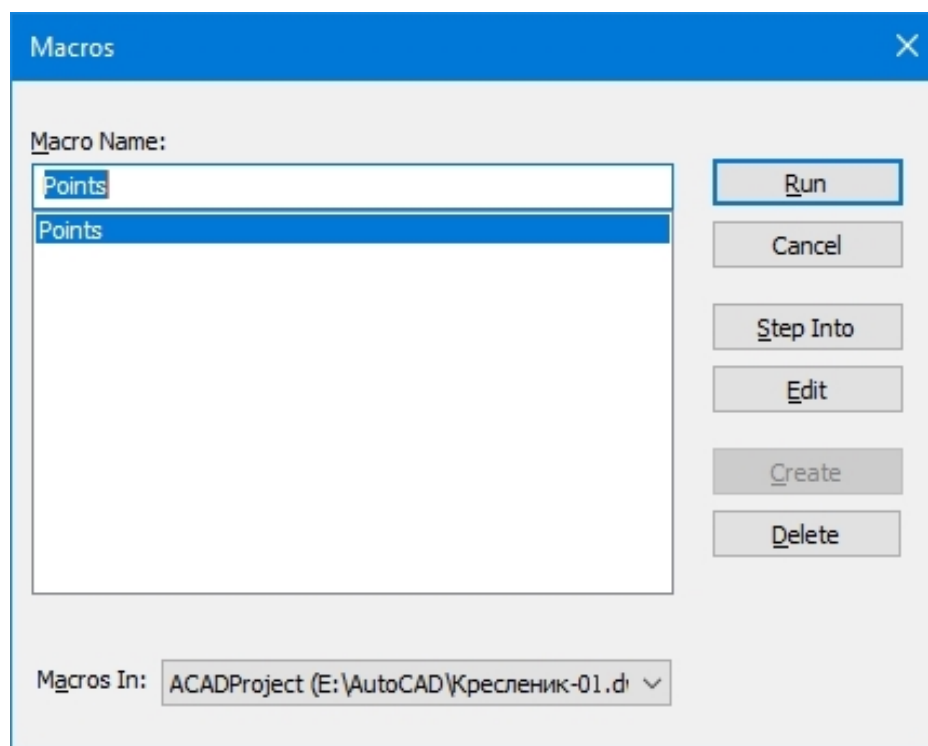


Fig. 3.7. Running a Macro in the Visual Basic Environment

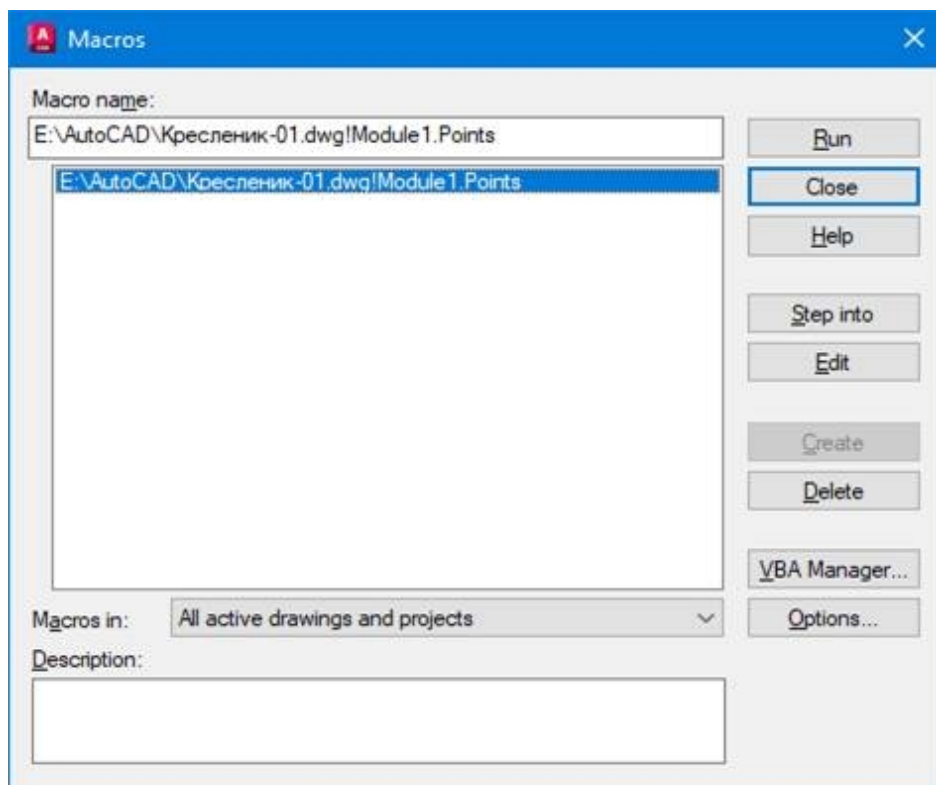


Fig. 3.8. Running a Macro from AutoCAD

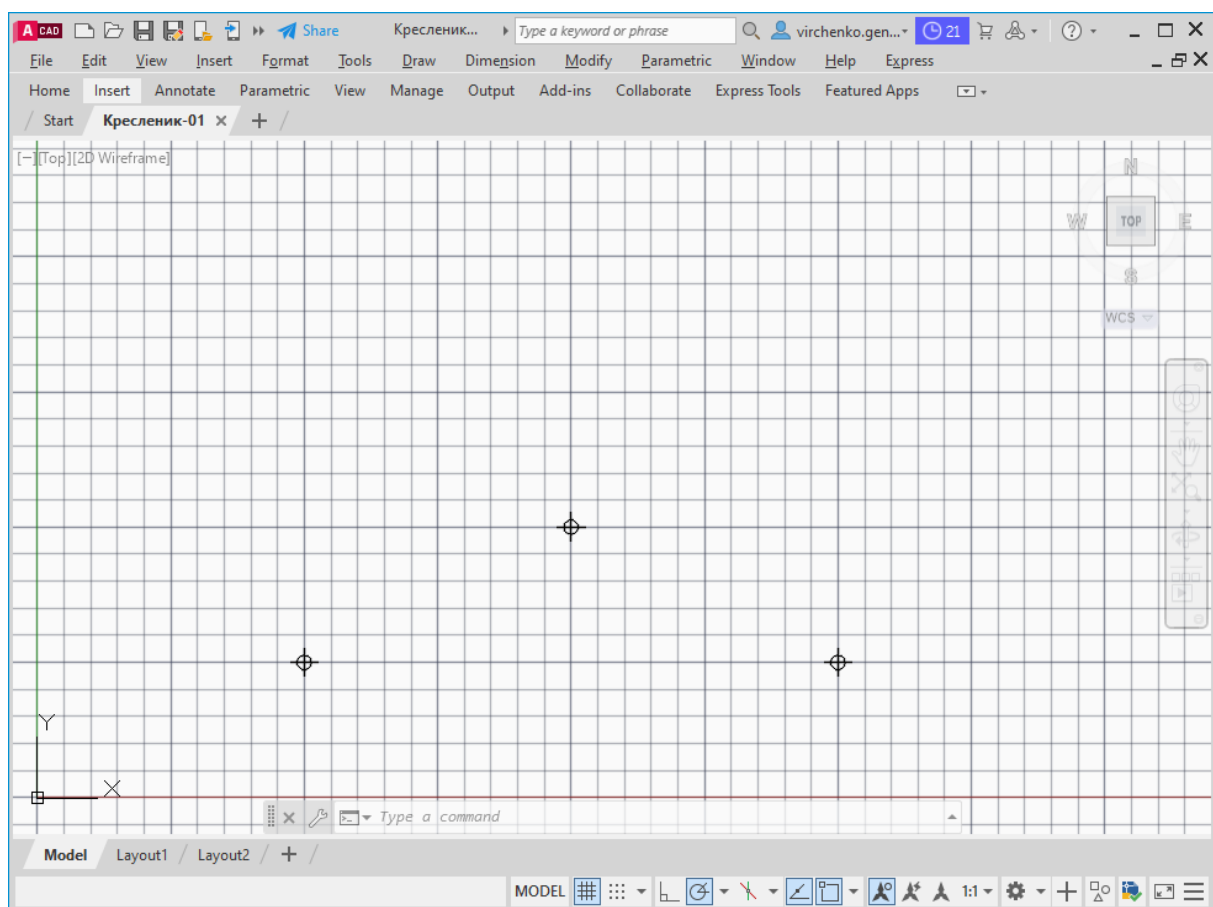


Fig. 3.9. Result of Macro Execution

In the Visual Basic environment, the macro is saved by clicking the *Save* button on the Standard toolbar.

The following sections present automated constructions of specific geometric figures. Each example includes a brief discussion of the underlying mathematical principles and demonstrates how the figures can be created using both AutoCAD tools and Visual Basic for Applications.

3.2. Construction of Basic Figures

Experience shows that the rapid and effective adoption of new computer-aided design systems is greatly facilitated by a solid theoretical foundation. Such preparation forms the user's understanding of how graphical software packages operate at the fundamental level. Part of this groundwork was established in Chapter 1, where modern computer-aided geometric modeling was examined, including the development and application of tools used to represent various objects, processes, and phenomena. The chapter also described common types of geometric models and their structure.

The purpose of two-dimensional geometric modeling, typically performed in a plane, is to create graphical constructions that rely heavily on points and different types of lines. Points are the simplest geometric entities, defined solely by their spatial coordinates. In CAD systems, a wide range of symbols and colors can be used to visualise them. Lines, viewed as the paths traced by moving points, possess length as a key property. Depending on whether a line lies in a plane, it is considered planar or spatial; in two-dimensional modeling we work only with the former.

Points are usually defined by their coordinates, while lines are represented by mathematical equations. Since parametric curves expressed in the rectangular coordinate system Oxy are the most common in computer-aided shape generation, they will be the focus of our further consideration. Parametric definitions are especially convenient in automated modeling, as they allow figures to be generated algorithmically and easily adapted through changes in parameter values.

In this subsection, we examine the automated construction of points, line segments, rectangles, polygons, circles, ellipses, their arcs, and splines. These figures constitute the foundational elements of two-dimensional modeling and serve as building blocks for the creation of more complex technical drawings requiring precision and formal accuracy.

3.2.1. Points

A large number of computer graphics algorithms rely on operations with points, which are defined solely by their position in space. Their importance follows from a fundamental systems principle: any geometric figure can be viewed as a set of points. In the rectangular Cartesian coordinate system Oxy , a point $\mathbf{P}$ is defined by the radius vector:

$$\mathbf{P}=(x, y). \quad (3.1)$$

In polar coordinates, the point can be represented as $\mathbf{P}=(\rho, \varphi)$, where ρ is the radius (distance from the origin) and φ is the angle. In AutoCAD, creating a point simply involves specifying its coordinates. The visual appearance of points on the screen is controlled through the Point Style dialog, which is accessed using *Format* $\rightarrow$ *Point Style...* .

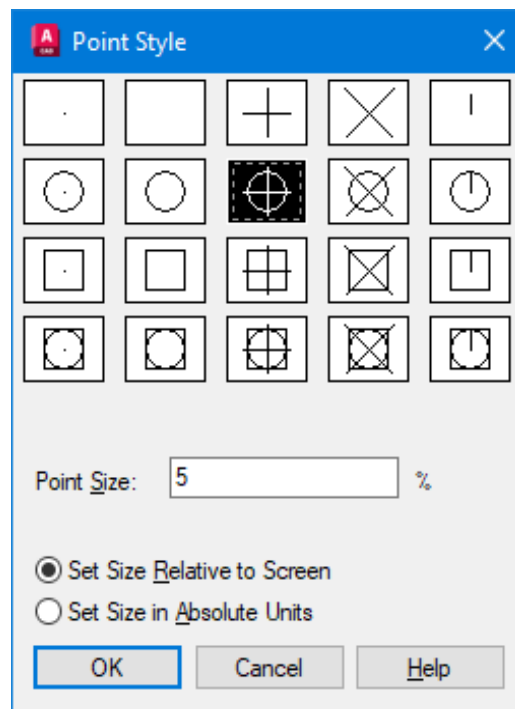


Fig. 3.10. Menu for setting point display parameters

Figure 3.11 shows an example of constructing points directly in AutoCAD using the sequence *Draw* $\rightarrow$ *Point* $\rightarrow$ *Multiple Point* from the traditional menu. The points were created using the vectors $\mathbf{P}_0=(100, 50)$, $\mathbf{P}_1=(200, 100)$, $\mathbf{P}_2=(300, 50)$ and the display settings shown in Fig. 3.10.

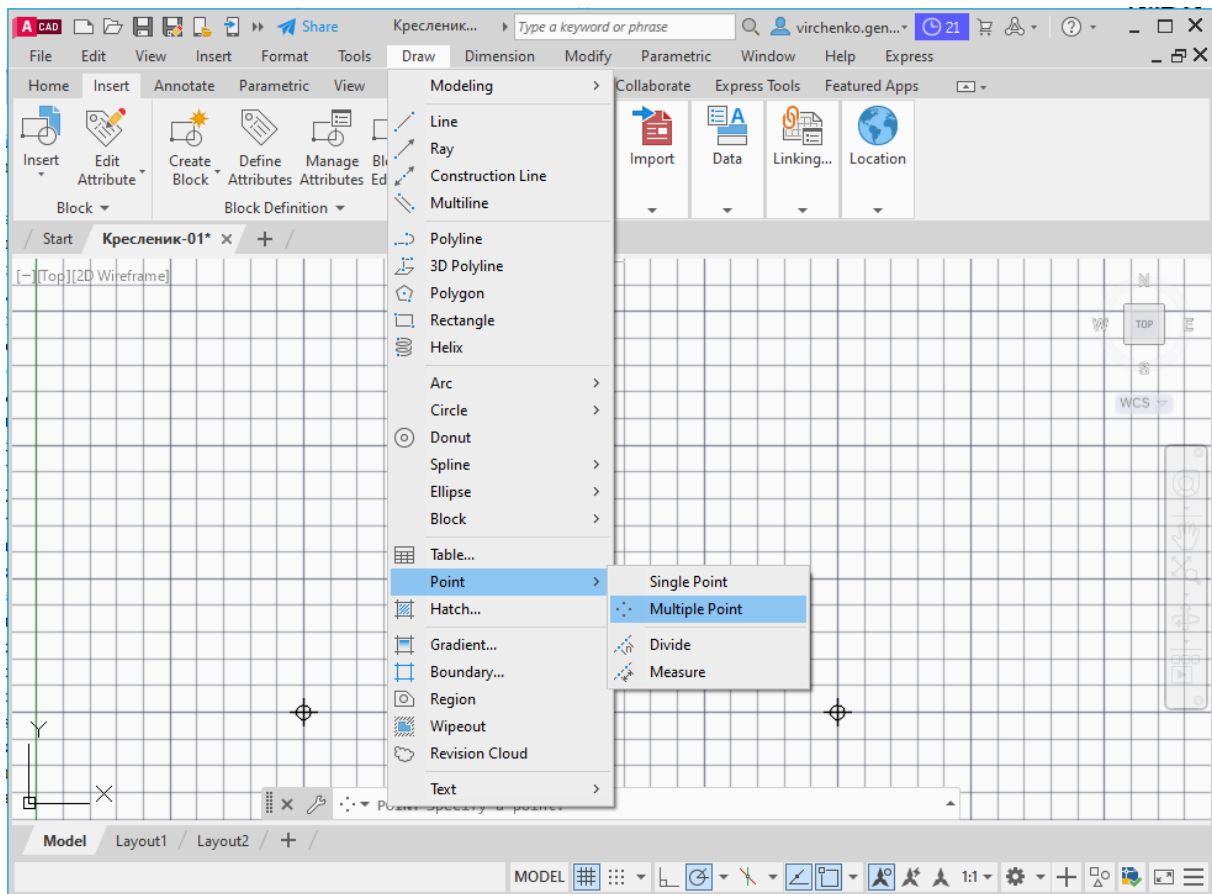


Fig. 3.11. Creating points directly in AutoCAD

The same task (see Fig. 3.9) can be completed using the *Points* macro shown in Fig. 3.6, the full code of which is provided below.

```
Sub Points()
  Dim point As AcadPoint 'point
  Dim location(0 To 2) As Double 'array of coordinates x, y, z of a point
  Dim Points(0 To 8) As Double 'array of coordinates x, y, z of three points
  'x, y, z coordinates of the array of points (in plane z=0)
  Points(0) = 100: Points(1) = 50: Points(2) = 0
  Points(3) = 200: Points(4) = 100: Points(5) = 0
  Points(6) = 300: Points(7) = 50: Points(8) = 0
  'point display format
  ThisDrawing.SetVariable "PDMODE", 34
  ThisDrawing.SetVariable "PDSIZE", -5
  'drawing of three points
  For i = 0 To 8 Step 3
    location(0) = Points(i) 'x
    location(1) = Points(i + 1) 'y
    location(2) = Points(i + 2) 'z
    Set point = ThisDrawing.ModelSpace.AddPoint(location)
  Next i
End Sub
```

Let us clarify how this program works. In the first block (the three lines immediately following the header), the variables used in the macro are declared. In the second block (after the fourth comment), the coordinates of the points are defined. The next block specifies the settings for how these points will be displayed. Finally, inside the loop, the program generates the corresponding point objects.

The point styles shown in Fig. 3.10 are controlled by the AutoCAD system variable PDMODE. The first row of symbols corresponds to the values 0, 1, 2, 3, and 4. Adding 32, 64, or 96 to any of these values selects the second, third, or fourth row of point styles, respectively.

The PDSIZE variable controls the size of the displayed point markers (except when PDMODE = 0 or PDMODE = 1). A positive value specifies the absolute size of the marker, while a negative value specifies a percentage of the height of the AutoCAD drawing area (PDSIZE = 0 corresponds to 5%).

Point markers are updated after the geometric model is regenerated.

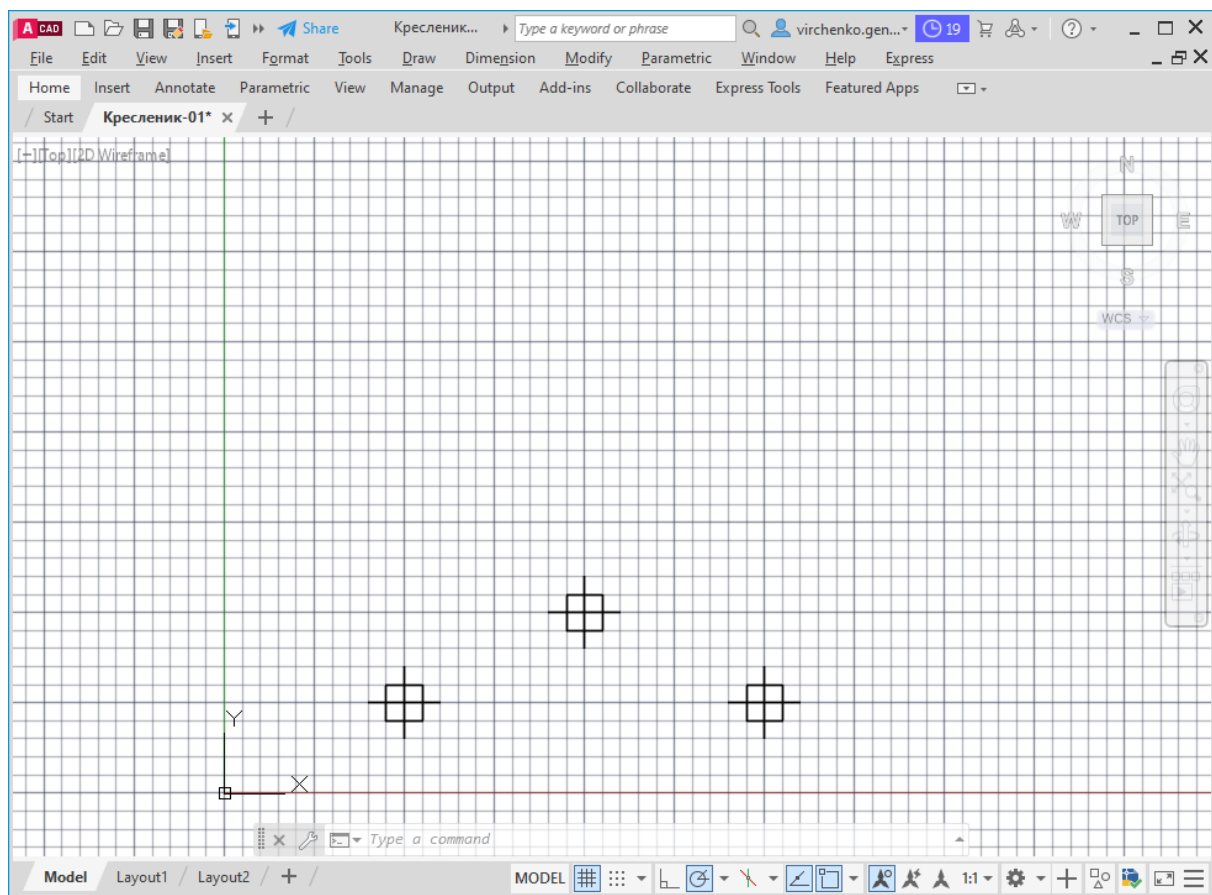


Fig. 3.12. Example of modifying point display parameters

The figure corresponds to the following fragment of the macro:

```

'point display format
ThisDrawing.SetVariable "PDMODE", 66
ThisDrawing.SetVariable "PDSIZE", 20

```

An analysis of Fig. 3.12 shows that a drawing grid with a spacing of 10 mm was used, as confirmed by the coordinates and sizes of the constructed points.

3.2.2. Line Segments

The analytical definition of a line segment with endpoints $\mathbf{P}_0$ and $\mathbf{P}_1$ can be conveniently represented using the following parametric equation:

$$\mathbf{P}(u) = (1-u)\mathbf{P}_0 + u\mathbf{P}_1, \quad (3.2)$$

where $u \in [0, 1]$ is a parameter.

Any dependence $\mathbf{P}(t)$, where $t \in [t_{\min}, t_{\max}]$, can similarly be rewritten using $\mathbf{P}(t(u))$, where $u \in [0, 1]$, by applying the expression $t(u) = t_{\min}(1-u) + ut_{\max}$.

Although the points $\mathbf{P}_0$ and $\mathbf{P}_1$ may be represented by their two-dimensional position vectors (3.1), we will continue to express them in the more general form of three-dimensional coordinates, as in the previous programming example. This allows greater flexibility in defining arbitrary modeling planes within 3D space, or even performing two-dimensional modeling on other surfaces. These considerations apply to the remaining material presented in this chapter.

In AutoCAD, straight line segments are created using the sequence *Draw* → *Line* (see the second item from the top in the menu shown in Fig. 3.11). The user simply enters the coordinates of the segment's endpoints.

The corresponding macro is shown below.

Using this Lines macro, a triangle is constructed (Fig. 3.13) whose vertex coordinates match the points examined earlier.

```

Sub Lines()
  Dim line As AcadLine          'line segment
  'arrays of coordinates x, y, z of the endpoints of the line segment
  Dim startPoint(0 To 2) As Double, endPoint(0 To 2) As Double
  Dim Points(0 To 8) As Double  'array of coordinates x, y, z of triangle
  'vertices
  'x, y, z coordinates of triangle vertices (in plane z=0 by default)
  Points(0) = 100: Points(1) = 50:
  Points(3) = 200: Points(4) = 100:
  Points(6) = 300: Points(7) = 50:
  'triangle construction

```

```

For i = 0 To 8 Step 3
  j = IIf(i < 6, 0, -9)           'shift in the Points array
  startPoint(0) = Points(i)
  startPoint(1) = Points(i + 1)
  startPoint(2) = Points(i + 2)
  endPoint(0) = Points(i + 3 + j)
  endPoint(1) = Points(i + 4 + j)
  endPoint(2) = Points(i + 5 + j)
  'drawing the line segment
  Set line = ThisDrawing.ModelSpace.AddLine(startPoint, endPoint)
Next i
End Sub

```

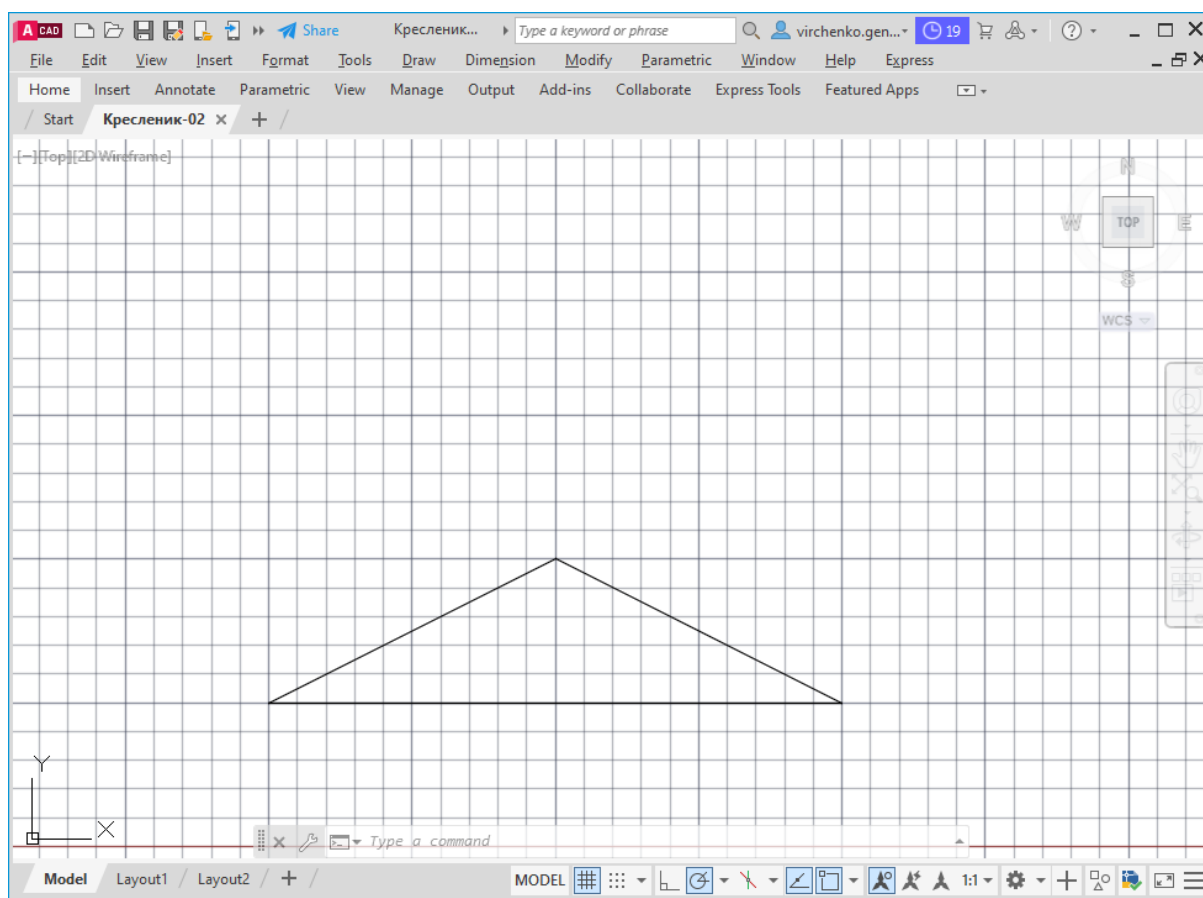


Fig. 3.13. Result of executing the Lines macro

The comments embedded in the program provide a sufficiently clear explanation of how it operates.

Here we highlight only the role of the variable j , which becomes non-zero when constructing the closing side of the triangle. In that step, the coordinates of the first vertex are reused to complete the final segment.

The adjustments to the z-coordinates in the block that defines the Points array demonstrate how two-dimensional geometric modeling can be performed in a plane other than $z=0$.

```
'x, y, z coordinates of triangle vertices
Points(0) = 100: Points(1) = 50: Points(2) = 0
Points(3) = 200: Points(4) = 100: Points(5) = 200
Points(6) = 300: Points(7) = 50: Points(8) = -200
```

The resulting construction is shown in Fig. 3.14, produced using the sequence *View* → *Orbit* → *Free Orbit* from the traditional AutoCAD menu. As seen in the figure, the triangles lie in different planes. This illustrates how two-dimensional objects may be modeled in three-dimensional space.

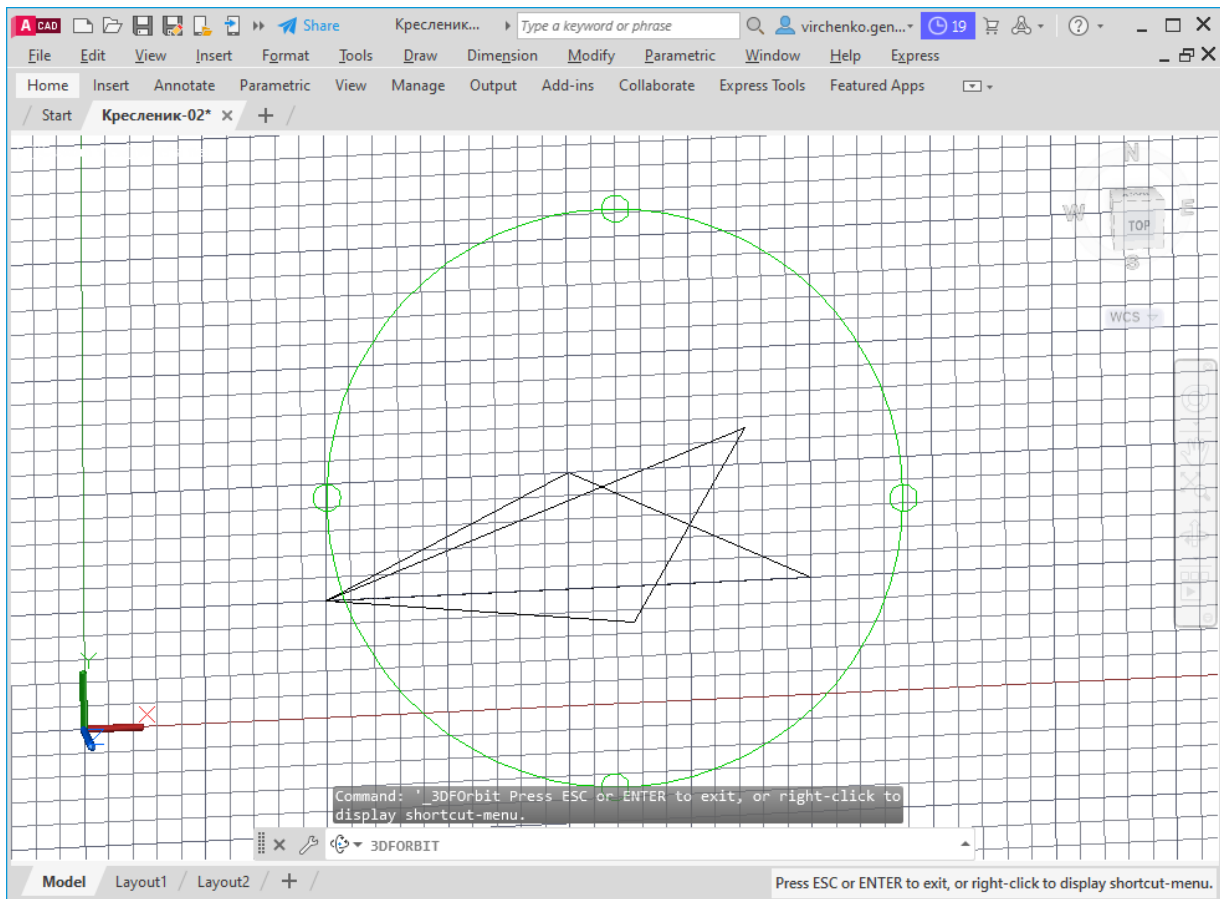


Fig. 3.14. Construction of a triangle in an arbitrary plane using a macro

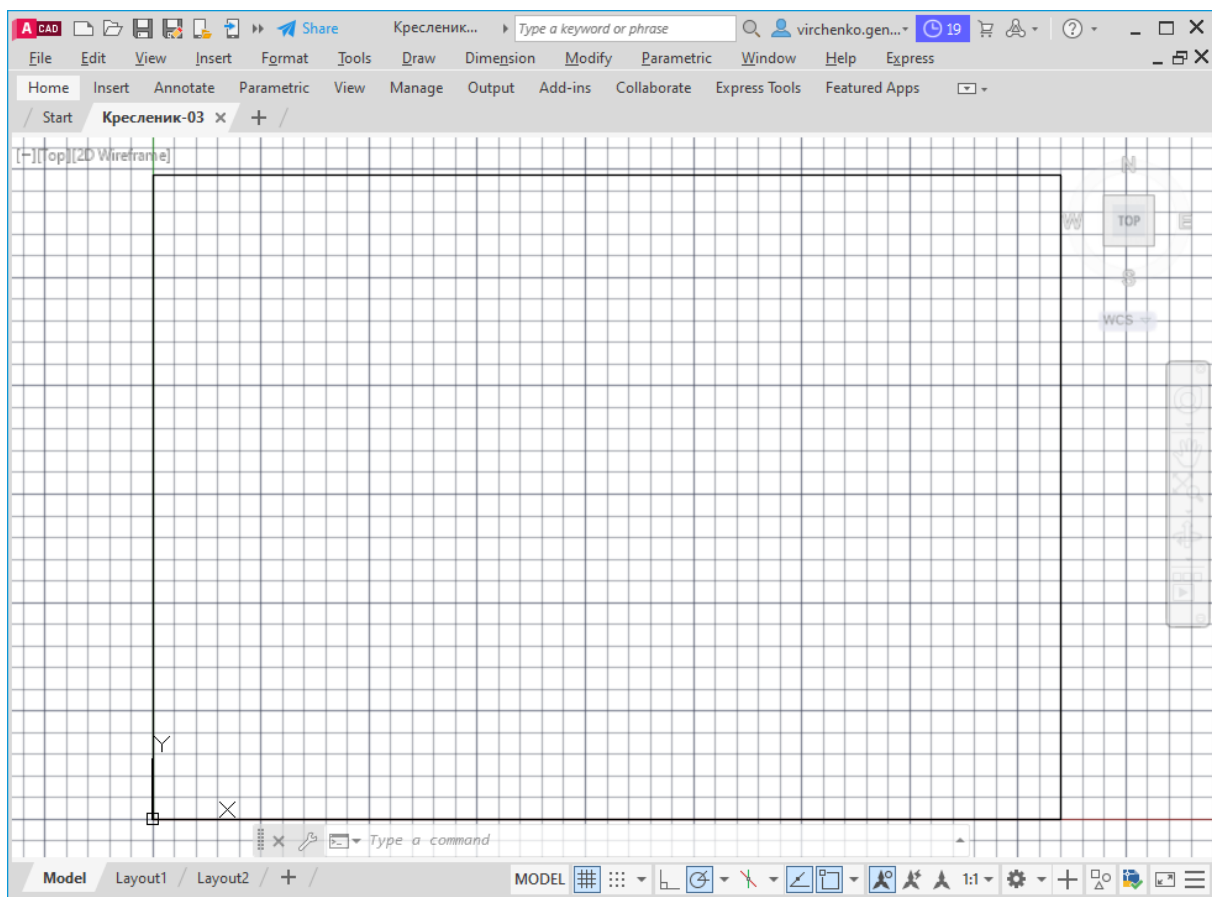
Thus, at this point we have covered – in both theory and practice – the automated generation of points and straight line segments in AutoCAD using Visual Basic for Applications. These elements form the foundation for constructing more advanced geometric figures.

To build upon this knowledge, we now turn to the automated creation of rectangles and polygons. These figures extend the basic principles already introduced and demonstrate how simple geometric entities can be combined into more complex planar constructions.

3.2.3. Rectangles

From a systems perspective, a rectangle can be viewed as a composite figure made up of points (3.1) and line segments (3.2). Since we have already examined the automated generation of these basic elements, we can now focus on aspects that have not yet been discussed.

Although a rectangle in AutoCAD can be created manually using four individual line segments, a more efficient approach is to use the command sequence *Draw* → *Rectangle* (see the ninth item from the top in the drop-down menu shown in Fig. 3.11). An example of defining this figure with the coordinates (x, y) of the first corner at $(0, 0)$ and the second corner at $(420, 297)$, dimensions corresponding to an A3 sheet – is shown in Fig. 3.15.



1. Fig. 3.15. Creating a rectangle using AutoCAD tools

When the rectangle is selected, the *Properties* context menu (opened with the right mouse button) can be used to adjust attributes such as color, layer, linetype, linetype scale, linewidth, and the coordinates of its vertices, if needed (Fig. 3.16).

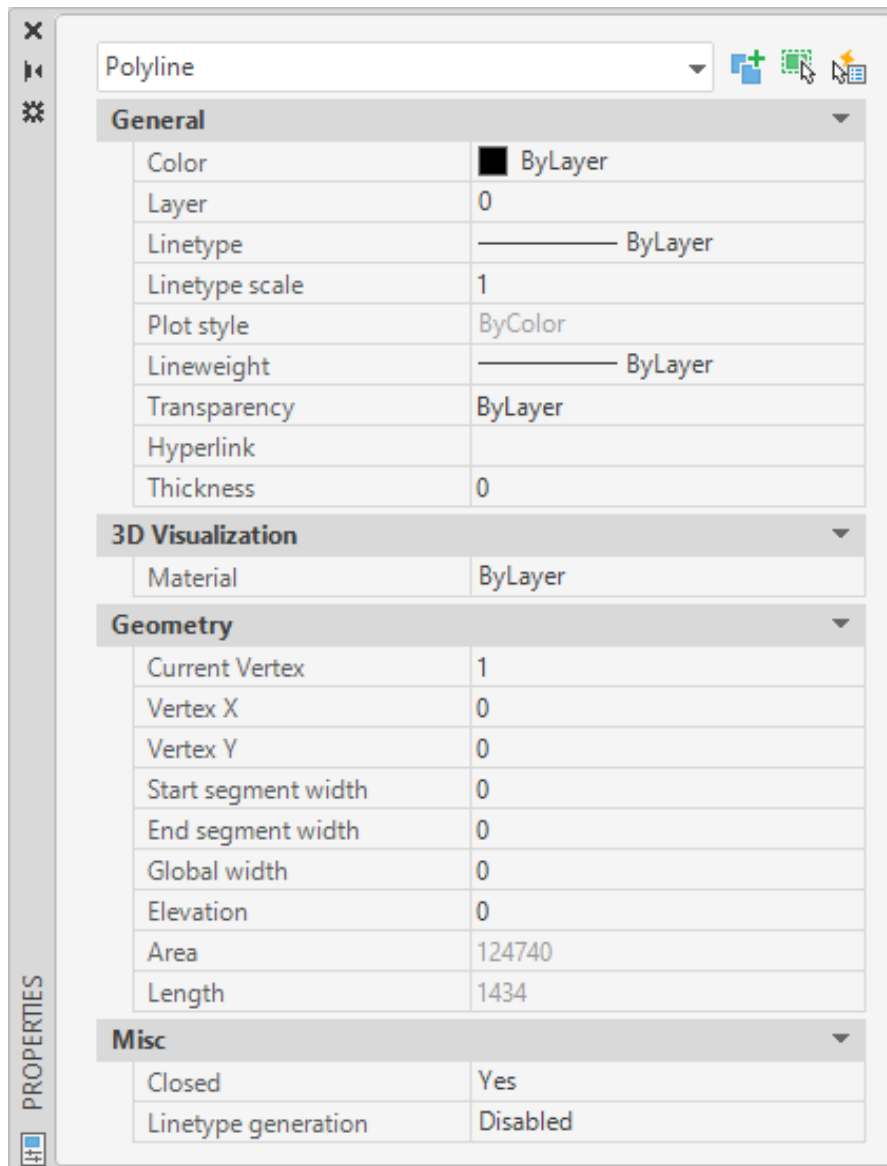


Fig. 3.16. Using the Properties menu for a rectangle

We can also construct the rectangle using a Visual Basic for Applications macro. In this case, we introduce a new automation technique used in modern CAD environments: sending a command directly from the macro together with the required argument values. These arguments replicate the information a user enters during an interactive command session in AutoCAD.

These arguments replicate the inputs a user would normally provide when running the command interactively in AutoCAD. The *Rectangle* macro is shown below.

```
Sub Rectangle()  
    'sending a command to AutoCAD to create a rectangle  
    ThisDrawing.SendCommand "rectang 0,0 420,297 "  
End Sub
```

This macro uses the `SendCommand` method, which simulates issuing the AutoCAD command `RECTANG` along with the required parameters. In the string placed in quotation marks, the command and its arguments are specified exactly as AutoCAD expects them. Note that empty spaces within the string represent the user pressing the Enter key.

The `SendCommand` method can be used not only for rectangles but also for many other AutoCAD commands. By integrating such macros into the design workflow, even simple geometric constructions can be generated more consistently and with significantly less effort.

3.2.4. Polygons

Rectangles represent a special case of polygons, so extending automated modeling techniques to more general polygonal shapes is a next step. Unlike rectangles, which have fixed orthogonal structure, polygons may vary in the number of sides, orientation, and geometric properties – making their automated construction particularly useful in engineering practice.

Polygons can be viewed as a further generalization of points, line segments, triangles, and rectangles. A polygon is defined as a closed polyline consisting of vertices (points) connected by straight line segments. In this section, we consider only simple polygons – those without self-intersections. Depending on the number of sides n , where $n \geq 3$ and $n \in \mathbb{N}$, polygon is referred to as an n -gon, (triangle, quadrilateral, pentagon, etc.).

A planar polygon is the finite region of the plane enclosed by the polygon. A polygon is convex if it lies entirely within one half-plane determined by any of its sides.

A convex polygon is called regular if all of its sides and interior angles are equal (e.g., an equilateral triangle, square, pentagon, etc.). A polygon is inscribed in a circle if all vertices lie on the circumference, and circumscribed about a circle if all sides are tangent to it.

In AutoCAD a *polyline* is constructed using the sequence *Draw* → *Polyline* (see the sixth item from the top in the drop-down menu shown in Fig. 3.11).

A corresponding example is shown in Fig. 3.17.

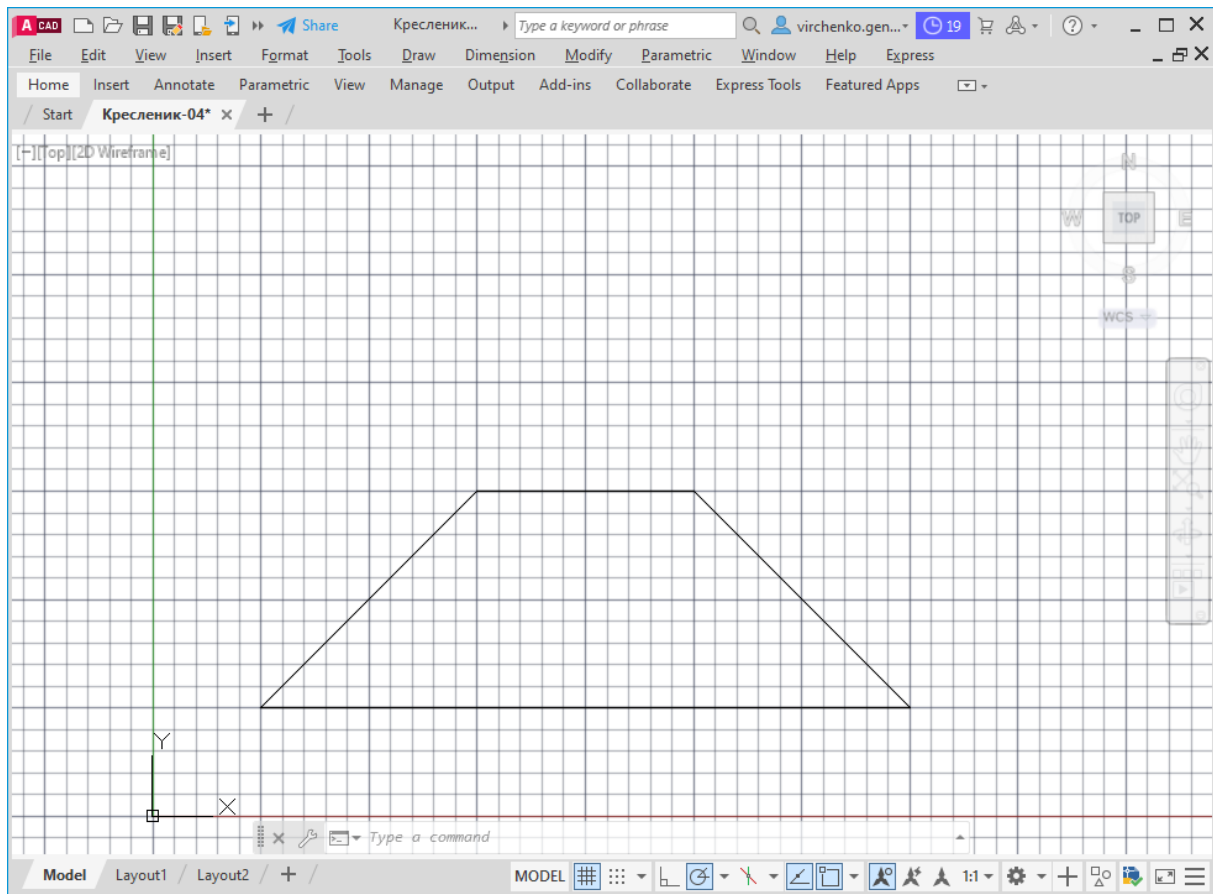


Fig. 3.17. A closed polyline constructed using AutoCAD tools

The polyline was created with the following vertex coordinates (x, y): $P_0=(50, 50)$, $P_1=(150, 150)$, $P_2=(250, 150)$, $P_3=(350, 50)$, $P_0=(50, 50)$.

The same construction can be reproduced using the *Polylines* macro in Visual Basic for Applications, as demonstrated below.

```
Sub PolyLines()
    Dim polyline As AcadPolyline    'polyline
    Dim Points(0 To 14) As Double    'array of coordinates x, y, z of polyline
    'vertices
    'x, y, z coordinates of polyline vertices in plane z=0
    Points(0) = 50: Points(1) = 50: Points(2) = 0
    Points(3) = 150: Points(4) = 150: Points(5) = 0
    Points(6) = 250: Points(7) = 150: Points(8) = 0
    Points(9) = 350: Points(10) = 50: Points(11) = 0
    Points(12) = 50: Points(13) = 50: Points(14) = 0
    'polyline construction
    Set polyline = ThisDrawing.ModelSpace.AddPolyline(Points)
End Sub
```

Let us now consider the geometric modeling of regular polygons (Fig. 3.18). In AutoCAD, this is performed using the sequence *Draw* → *Polygon* (see the eighth item from the top from the drop-down menu in Fig. 3.11).

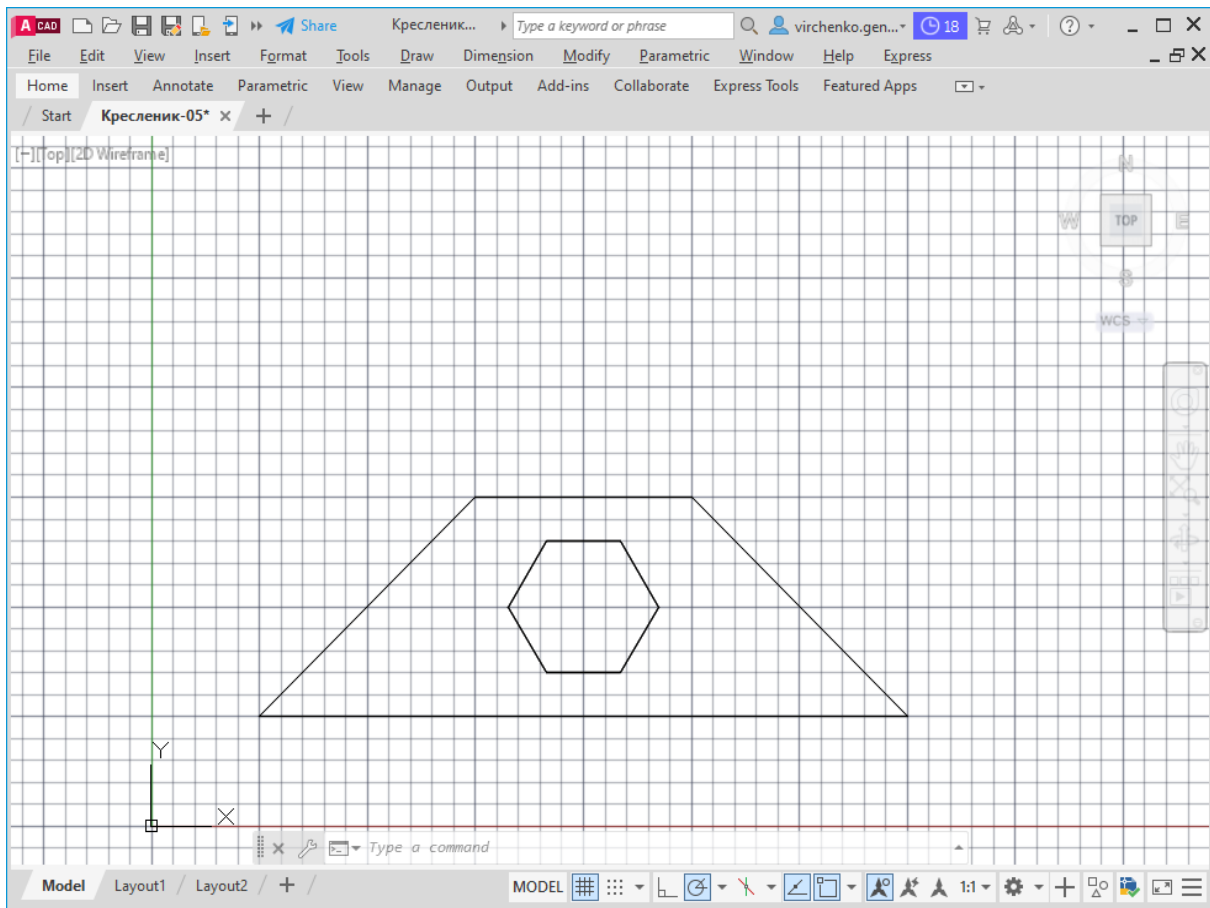


Fig. 3.18. Adding a regular hexagon

To construct a regular polygon, the user in AutoCAD specifies: the required number n of sides; the coordinates (x, y) of its center C ; whether the polygon is inscribed in or circumscribed about a circle; and the radius R of that circle. In the example shown, the following values were chosen: $n=6$; $C=(200, 100)$; an inscribed circle $R=30$. The resulting polygon is displayed in Fig. 3.18.

This same figure can be generated using the *Polygon* macro in Visual Basic for Applications.

```
Sub Polygon()  
    'sending a command to AutoCAD to draw a hexagon  
    ThisDrawing.SendCommand "polygon 6 200,100 C 30 "  
End Sub
```

The syntax of this macro is based on the SendCommand method described in the previous subsection. The arguments required by the POLYGON command have already been discussed earlier.

Since two-dimensional geometric modeling involves not only points and lines but also regions representing areas of the plane, let us consider an additional example..

In technical drawings, sectional and cut views are used frequently. These are represented by applying hatching to closed contours. To perform hatching in AutoCAD, the user selects the command sequence *Draw* → *Hatch* (see the seventh item from the bottom in the drop-down menu in Fig. 3.11).

The corresponding dialog window is shown in Fig. 3.19, which is similar to Fig. 2.127 in Chapter 2..

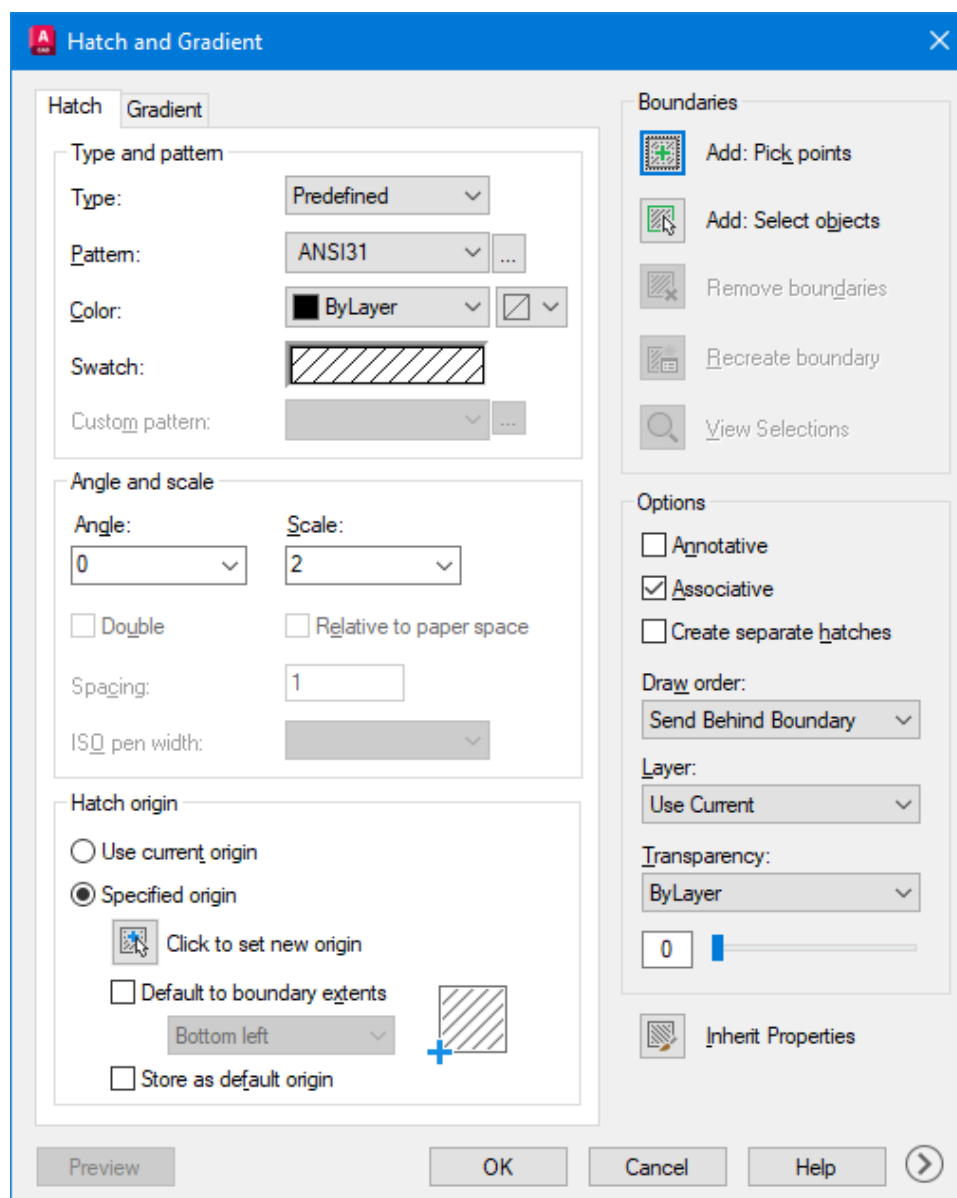


Fig. 3.19. Hatch command menu

In this example, the hatching pattern ANSI31 was selected, and the scale factor was set to 2. The resulting hatched figure is shown in Fig. 3.20.

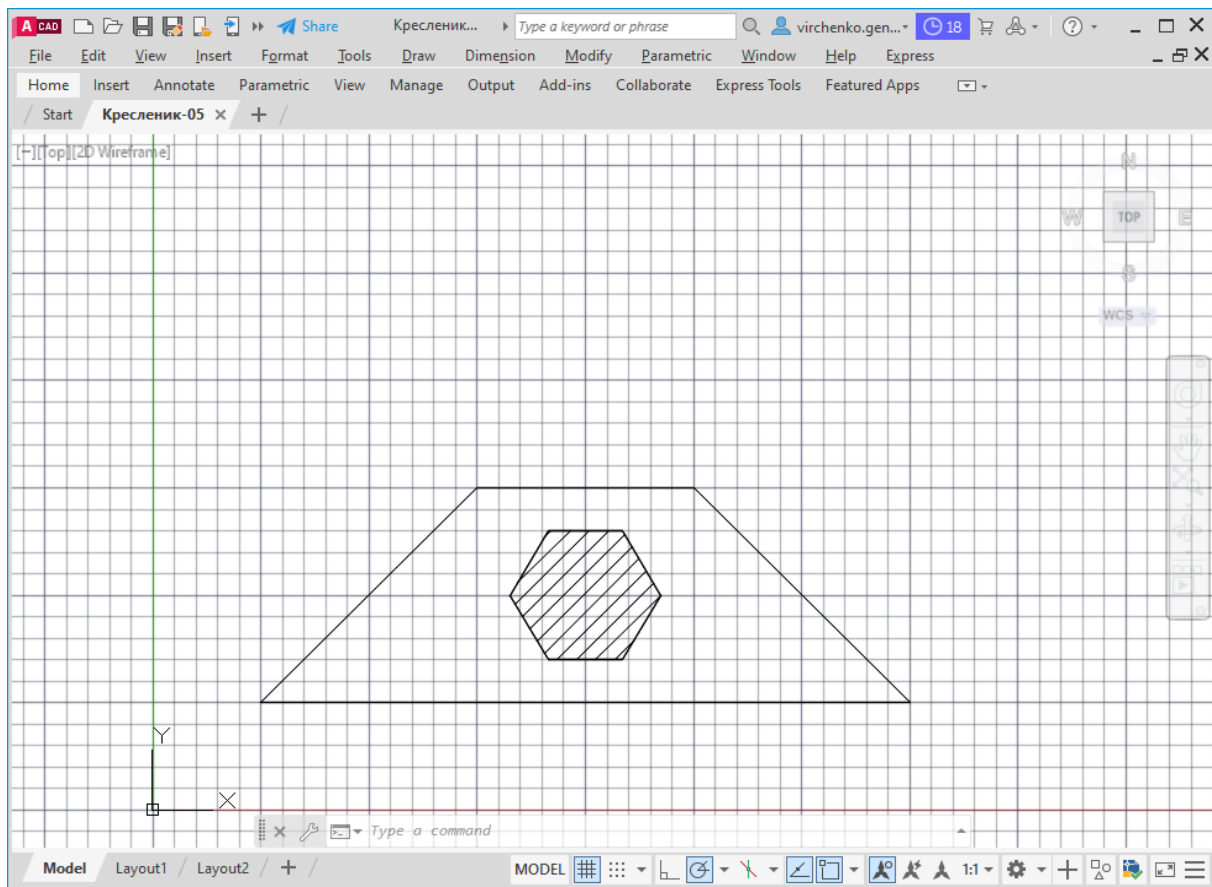


Fig. 3.20. Hatching of a hexagon

The hatched regular hexagon shown above can also be generated using the *Hatches* macro in the Visual Basic for Applications environment.

```
Sub Hatches()  
    Dim center(0 To 2) As Double      'x, y, z coordinates of the hexagon center  
    Dim radiusI As Double              'radius of the inscribed circle of the hexagon  
    Dim radiusD As Double              'radius of the circumscribed circle of the hexagon  
    Dim polyline(0 To 0) As AcadEntity 'hexagon polyline  
    Dim Points(0 To 20) As Double      'array of coordinates x, y, z of polyline  
    'vertices  
    Dim hatch As AcadHatch              'hatch  
    Dim patternName As String           'hatch pattern name  
    Dim patternType As Long             'hatch pattern type  
    Dim bAssociativity As Boolean        'hatch associativity  
    n = 6                               'number of sides of the regular polygon  
    center(0) = 200: center(1) = 100: center(2) = 0  
    radiusI = 30  
    pi = 4 * Atn(1)                     'constant value  
    radiusD = radiusI / Cos(pi / n)
```

```

For i = 0 To n * 3 + 2 Step 3
    'x, y, z coordinates of polyline vertices in plane z=0
    Points(i) = center(0) - radiusD * Cos(2 * pi / n * Int(i / 3)) 'x
    Points(i + 1) = center(1) - radiusD * Sin(2 * pi / n * Int(i / 3)) 'y
    Points(i + 2) = center(2) 'z
Next i
'polyline construction
Set polyline(0) = ThisDrawing.ModelSpace.AddPolyline(Points)
'hatch definition
patternName = "ANSI31"
patternType = 0
bAssociativity = True
'creating hatch object
Set hatch = ThisDrawing.ModelSpace.AddHatch(patternType, patternName, bAssociativity)
hatch.Rotate center, 0 'hatch rotation
hatch.AppendOuterLoop polyline 'outer boundary of hatch
hatch.LineWeight = acLnWt035 'hatch line thickness
hatch.PatternScale = 2 'scaling distance between hatch lines
hatch.Evaluate
ThisDrawing.Regen True 'model regeneration
End Sub

```

Let us clarify how this program operates. The embedded comments explain the logic of the macro, covering the coordinates of the figure's center, the radius of the inscribed circle, the geometric primitives involved (such as the polyline and its vertices, as well as the hatch object and its properties), the calculations needed to determine the required values, and the sequence of steps used to construct the final geometry.

Although the macro is not intended to be fully universal or capable of modeling all geometric shapes, it nevertheless demonstrates several useful features.

To make changes in shape, size, or orientation easier to observe, the resulting figures were placed over the background of the isosceles trapezoid shown in Fig. 3.17.

As a result of the modifications described below, the image shown in Fig. 3.21 was obtained.

Fig. 3.21. Formation of a hatched equilateral triangle

```

...
Dim Points(0 To 11) As Double 'array of coordinates x, y, z of polyline vertices
...
n = 3 'number of sides of the regular polygon
...
radiusI = 20

```

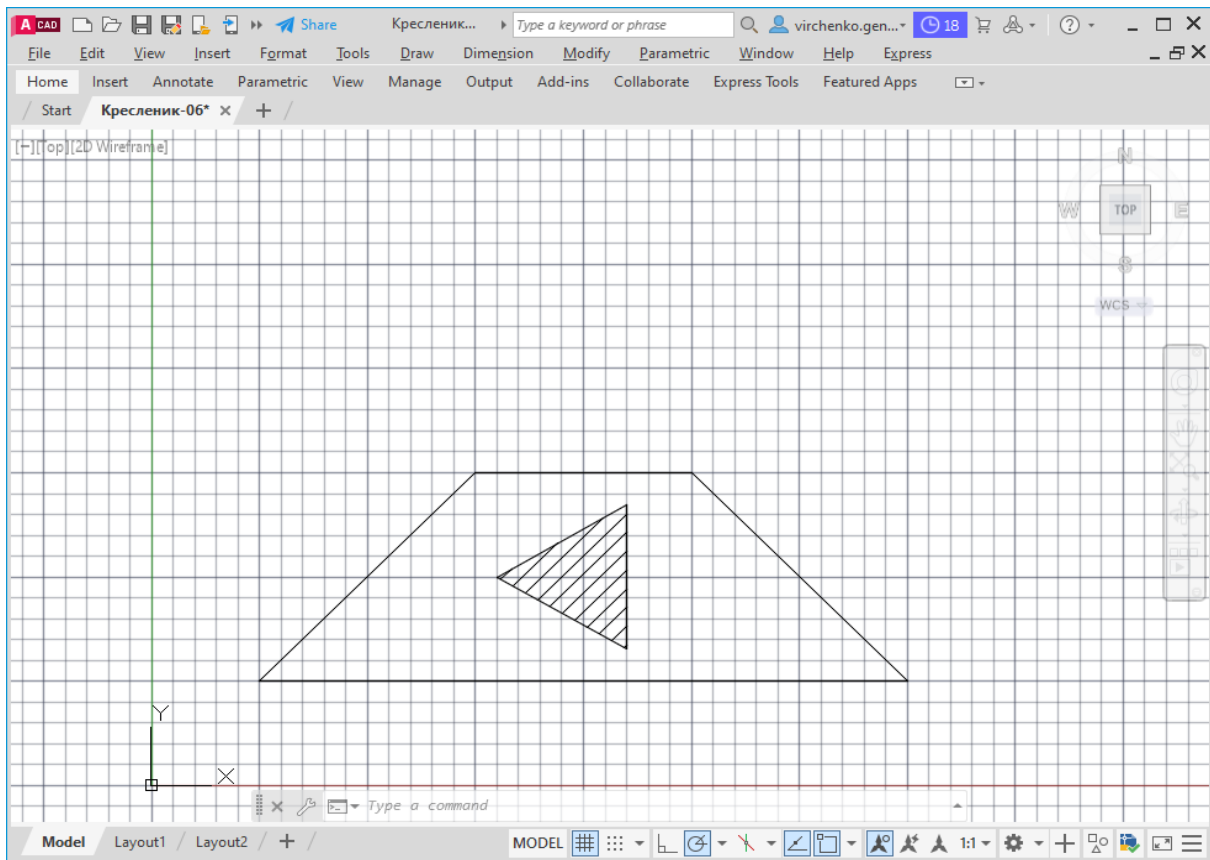


Fig. 3.21. Formation of a hatched equilateral triangle

The next set of modifications produces the result shown in Fig. 3.22.

```

...
Dim Points(0 To 14) As Double 'array of coordinates x, y, z of polyline vertices
...
n = 4                          'number of sides of the regular polygon
...
radiusI = 20
center(0) = 200: center(1) = 250: center(2) = 0

```

If a different hatching angle is required for the square, this can be achieved by modifying only a single line in the macro.

```

...
hatch.Rotate center, pi / 6 'hatch rotation

```

Applying the additional modifications shown below yields the image in Fig. 3.24.

```

...
Dim Points(0 To 17) As Double 'array of coordinates x, y, z of polyline
'vertices
...
n = 5                          'number of sides of the regular polygon
center(0) = 200: center(1) = 250: center(2) = 0
radiusI = 50
...
patternName = "HONEY"

```

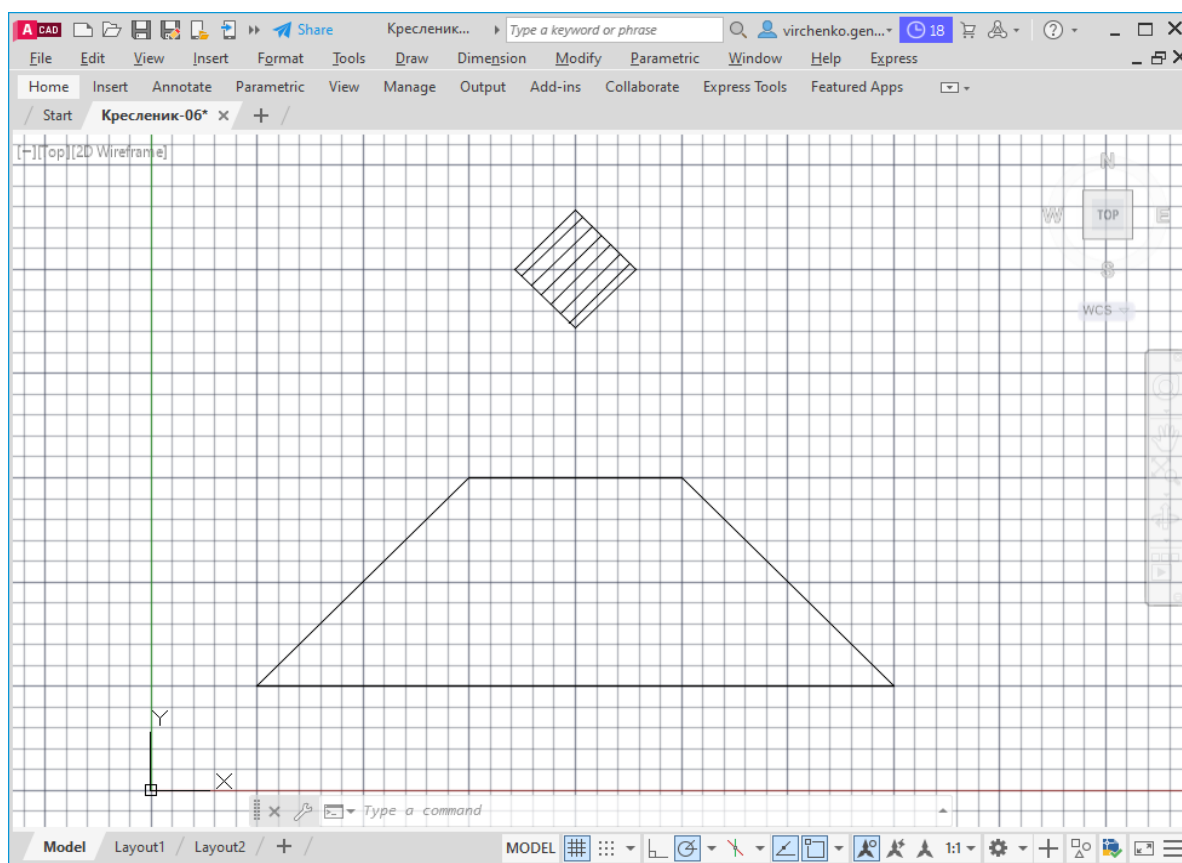


Fig. 3.22. Construction of a hatched square

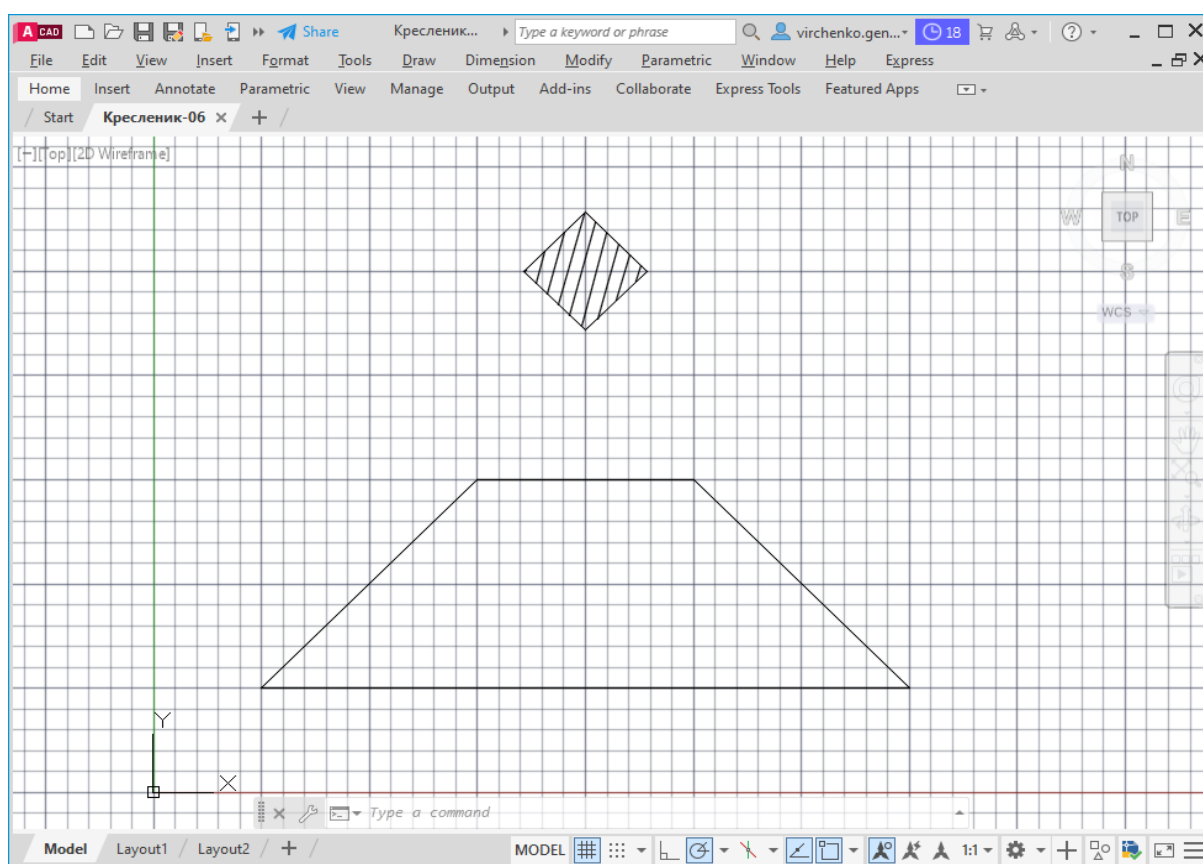


Fig. 3.23. Changing the hatching direction of a square

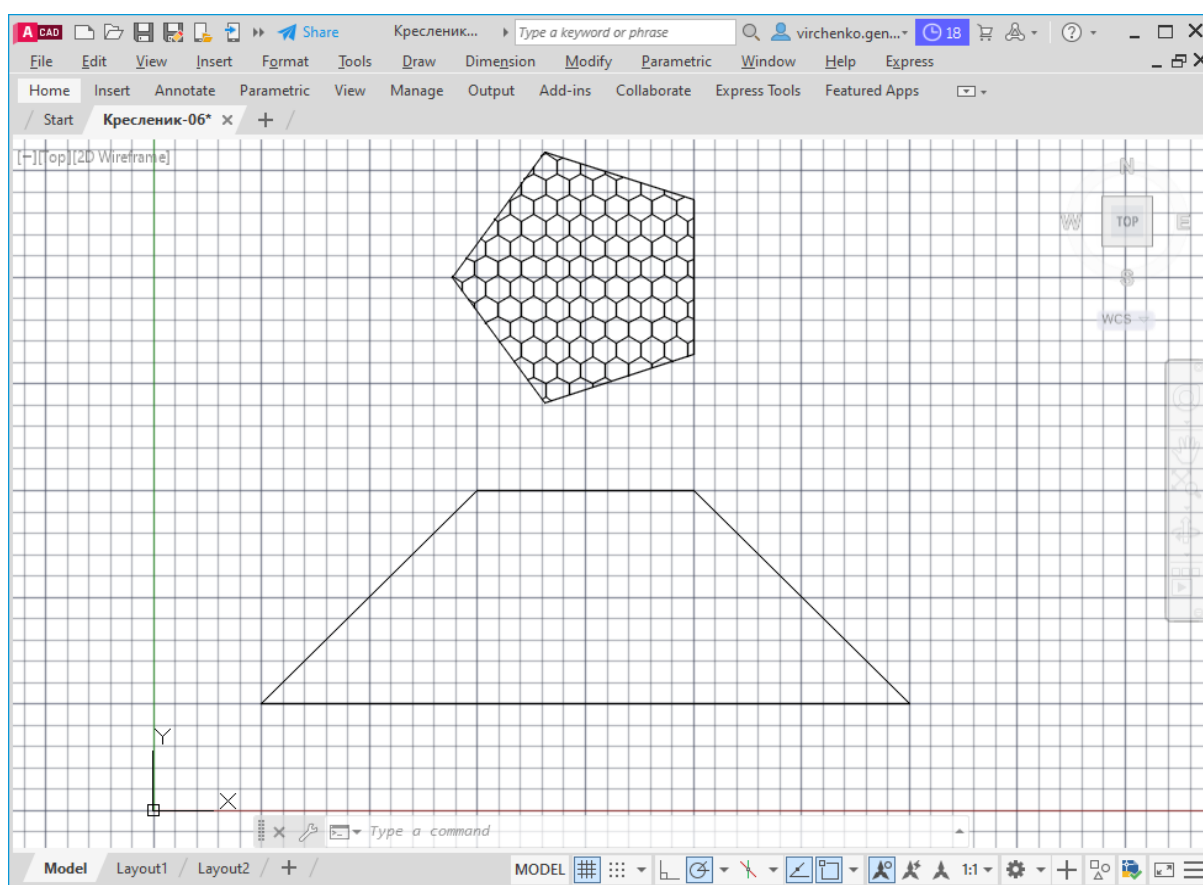


Fig. 3.24. A pentagon with HONEY hatching

These practical examples demonstrate the flexibility and efficiency of automated parametric shape generation using Visual Basic for Applications. Before moving on, let us consider one more useful capability offered by these tools.

In AutoCAD system, the *Properties* menu (as shown earlier in Fig. 3.16) can be used to obtain information such as the area and perimeter of a constructed polygon.

Access to this information is also available in Visual Basic for Applications. If the following lines are added at the end of the macro examined in this subsection:

```
Dim polyline2 As AcadPolyline
Dim strHandle As String
strHandle = polyline(0).Handle
Set polyline2 = ThisDrawing.HandleToObject(strHandle)
MsgBox "Polygon area: " & polyline2.Area & Chr(13) _
    & "Polygon perimeter: " & polyline2.Length
```

the macro will additionally display a message window with the computed results, as shown in Fig. 3.25.

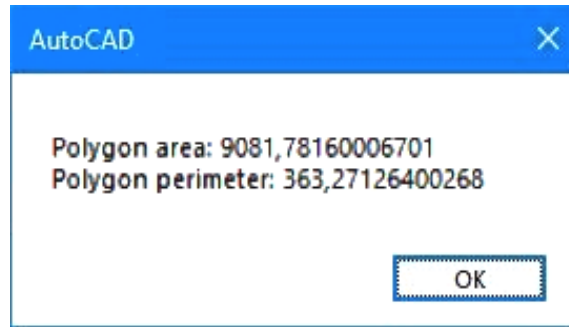


Fig. 3.25. Computed area and perimeter of the polygon

We now proceed to the construction of curved lines – specifically circles, ellipses, and splines.

3.2.5. Circles and Their Arcs

A circle in the rectangular Cartesian coordinate system Oxy , with center $C(x_C, y_C)$ and radius R , can be defined using the following parametric equations:

$$x = x_C + R \cos u; \quad y = y_C + R \sin u, \quad (3.3)$$

where $u \in [0, 2\pi]$.

These expressions are also convenient for constructing circular arcs of any angular extent simply by adjusting the interval of the parameter u .

In AutoCAD, a circle is created using the sequence *Draw* $\rightarrow$ *Circle* (Fig. 3.26). Several construction methods are available: by center and radius, by center and diameter, through two points, through three points, and others.

Note. In the following examples, the figures are shown inside a rectangular A3-format frame within the AutoCAD drawing area for improved clarity.

Circular arcs can also be created using *Draw* $\rightarrow$ *Arc*. As shown in Fig. 3.27, arcs may be defined in various ways: by three points; by start point, center, and endpoint; by start point, center, and angle; and so on.

Figure 3.28 illustrates several examples of circles and arcs.

The circles were created by specifying their center C and radius R . For the larger circle: $C=(200, 150)$, $R=100$; for the smaller circle $C=(100, 150)$, $R=50$. For the smaller one, the *Properties* menu (see subsection 3.2.3) was used to assign a red color, a dashed linetype, and a lineweight of 0.4 mm.

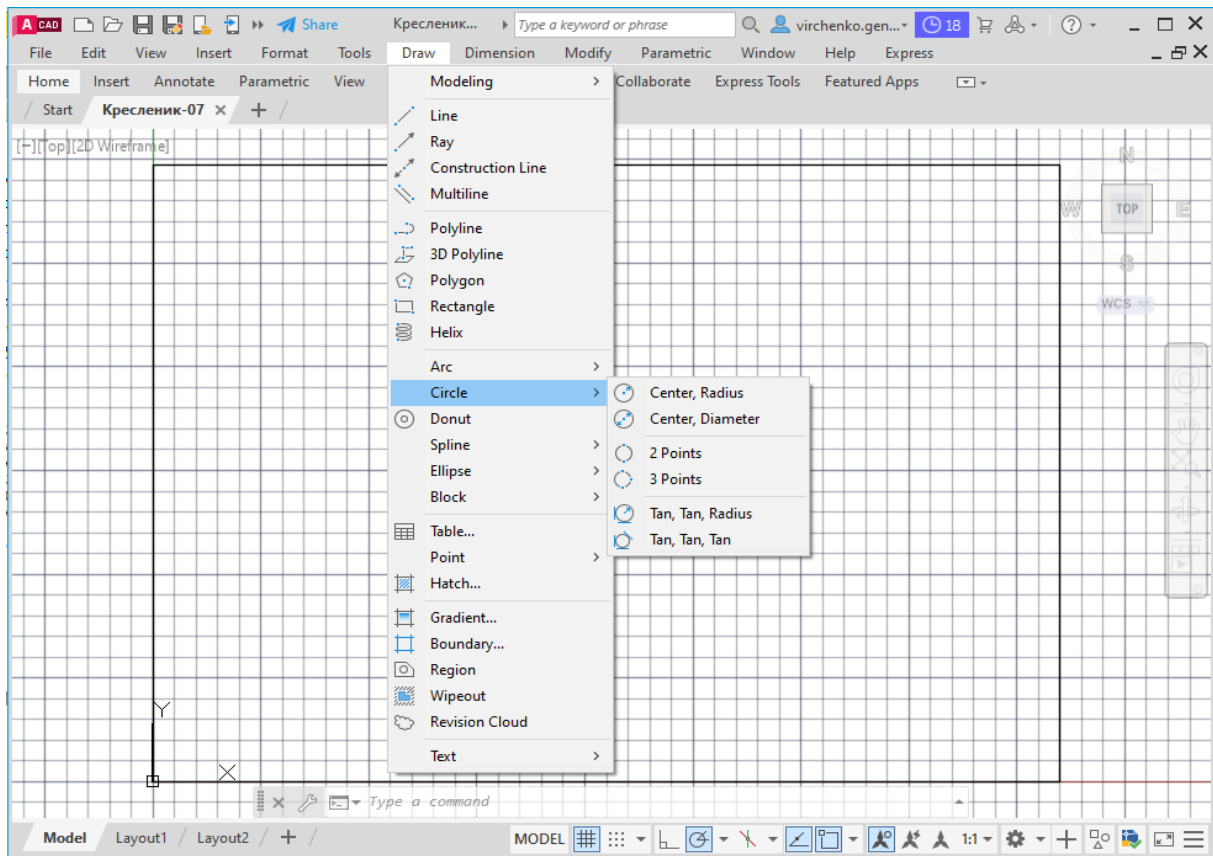


Fig. 3.26. AutoCAD menu for constructing circles

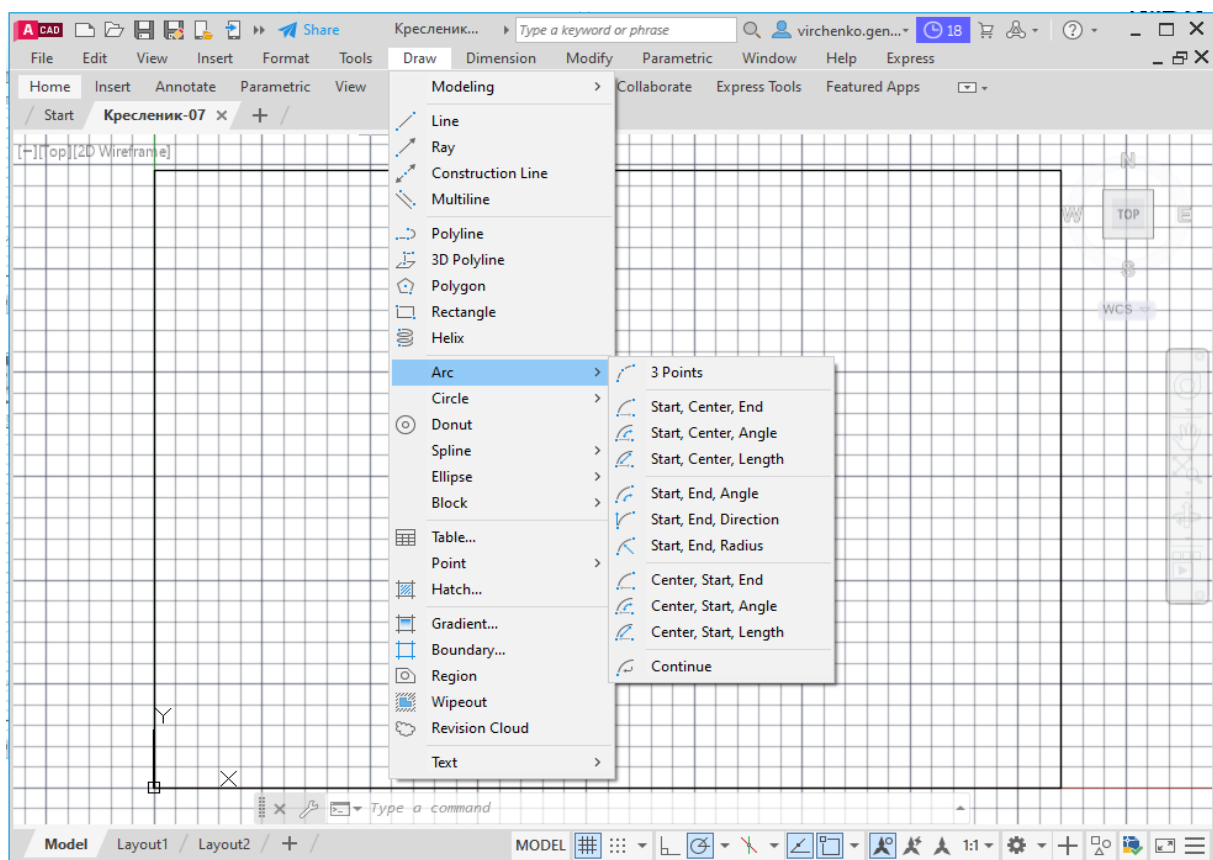


Fig. 3.27. AutoCAD menu for constructing circular arcs

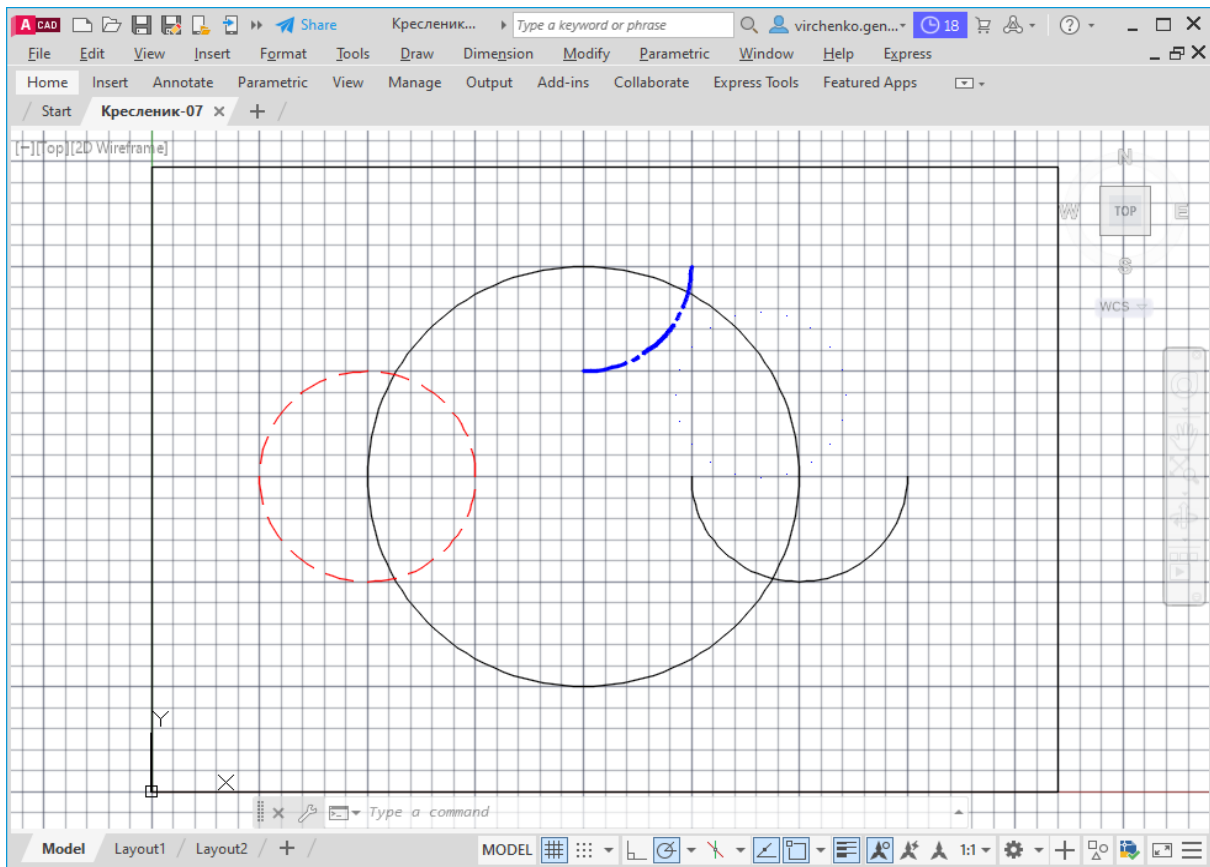


Fig. 3.28. Examples of creating circles and their arcs in AutoCAD

The arcs were constructed by specifying center C , start point P_0 , and endpoint P_1 . For the lower arc: $C=(300, 150)$, $P_0=(250, 150)$, $P_1=(350, 150)$. For the upper arc: $C=(200, 250)$, $P_0=(200, 200)$, $P_1=(250, 250)$. For the latter, a blue color, a center linetype, and a lineweight 1.2 mm were applied.

The geometric objects discussed above can also be created using a Visual Basic for Applications macro. The corresponding program is presented below.

```
Sub Circles()
    Dim circl As AcadCircle           'circle
    Dim center(0 To 2) As Double      'circle center
    Dim radius As Double              'circle radius
    Dim arc As AcadArc                'circle arc
    Dim startAngle As Double, endAngle As Double 'start and end angles of the
    ' circle arc
    'x, y, z coordinates of the center and circle radius
    center(0) = 200: center(1) = 150: center(2) = 0:
    radius = 100
    'drawing circle1
    Set circl = ThisDrawing.ModelSpace.AddCircle(center, radius)
    'x, y, z coordinates of the center and radius of circle2
    center(0) = 100: center(1) = 150: center(2) = 0:
    radius = 50
    'drawing circle2
    Set circl = ThisDrawing.ModelSpace.AddCircle(center, radius)
```

```

'setting circle2 to red color, dashed line and linewidth 0.4 mm
circl.color = acRed
circl.Linetype = "dashed"
circl.Lineweight = acLnWt040
circl.Update
pi = 4 * Atn(1)                                'constant value
'x, y, z coordinates of the center, radius and angles of arc1
center(0) = 300: center(1) = 150: center(2) = 0:
radius = 50: startAngle = -pi: endAngle = 0
'drawing arc1
Set arc = ThisDrawing.ModelSpace.AddArc(center, radius, startAngle, endAngle)
'x, y, z coordinates of the center, radius and angles of arc2
center(0) = 200: center(1) = 250: center(2) = 0:
radius = 50: startAngle = -pi / 2: endAngle = 0
'drawing arc2
Set arc = ThisDrawing.ModelSpace.AddArc(center, radius, startAngle, endAngle)
'setting arc2 to blue color, center line and linewidth 1.2 mm
arc.color = acBlue
arc.Linetype = "center2"
arc.Lineweight = acLnWt120
arc.Update
ThisDrawing.Regen True                        'model regeneration
End Sub

```

This macro is based on the AddCircle and AddArc methods, which construct a circle from its center and radius, and a circular arc from its center, radius, start angle, and end angle, respectively.

The comments included in the code explain how the macro operates.

It should also be noted that, before running the program, the linetypes DASHED and CENTER2 must be loaded into the drawing to ensure proper display of the constructed objects.

3.2.6. Ellipses and Their Arcs

The mathematical properties of *ellipses* are thoroughly discussed in a wide range of references. An ellipse centered at the origin of the rectangular coordinate system *Oxy* can be defined parametrically as:

$$x = a \cos u; \quad y = b \sin u, \quad (3.4)$$

where a and b are the semi-major and semi-minor axes of the ellipse ($a > b > 0$), $u \in [0, 2\pi]$ is the parameter.

A more general case is shown in Fig. 3.29, where the ellipse is centered at $C = (x_C, y_C)$ and its semi-major axis a is inclined at an angle γ relative to the positive direction of the Ox axis.

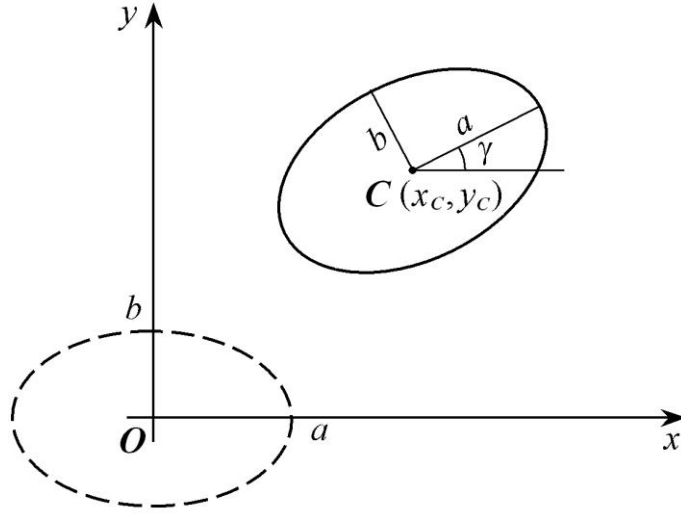


Fig. 3.29. Arbitrary placement of an ellipse in the Oxy plane

The equation of the rotated and translated ellipse can be obtained by first rotating the base ellipse (3.4) – shown with a dashed outline – around the origin point $O=(0, 0)$ by the angle γ , and then translating the result by the vector $T=(x_C, y_C)$.

Using homogeneous coordinates, the rotation matrix for angle γ is:

$$R(\gamma) = \begin{bmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (3.5)$$

and the translation matrix is:

$$T(x, y) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ x & y & 1 \end{bmatrix}. \quad (3.6)$$

Combining these transformations and applying (3.4) ... (3.6), we obtain:

$$\begin{aligned} [x \ y \ 1] \cdot R(\gamma) \cdot T(x, y) &= [x \ y \ 1] \cdot \begin{bmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ x_C & y_C & 1 \end{bmatrix} = \\ &= [a \cos u \ b \sin u \ 1] \cdot \begin{bmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ x_C & y_C & 1 \end{bmatrix} = \\ &= [x_C + a \cos u \cos \gamma - b \sin u \sin \gamma \ y_C + a \cos u \sin \gamma + b \sin u \cos \gamma \ 1]. \end{aligned} \quad (3.7)$$

Thus, based on expression (3.7), the resulting equation of the ellipse for the configuration shown in Fig. 3.29 becomes:

$$x = x_C + a \cos u \cos \gamma - b \sin u \sin \gamma; \quad y = y_C + a \cos u \sin \gamma + b \sin u \cos \gamma, \quad (3.8)$$

where $u \in [0, 2\pi]$.

Equations (3.5) and (3.8) are also convenient for generating an elliptical arc by restricting the interval of the parameter u .

In AutoCAD, ellipses and their arcs are created using the sequence *Draw* → *Ellipse* in the drop-down menu (Fig. 3.30). Two options are provided for constructing a full ellipse (center + axes, or axes only), and one option for constructing an elliptical arc.

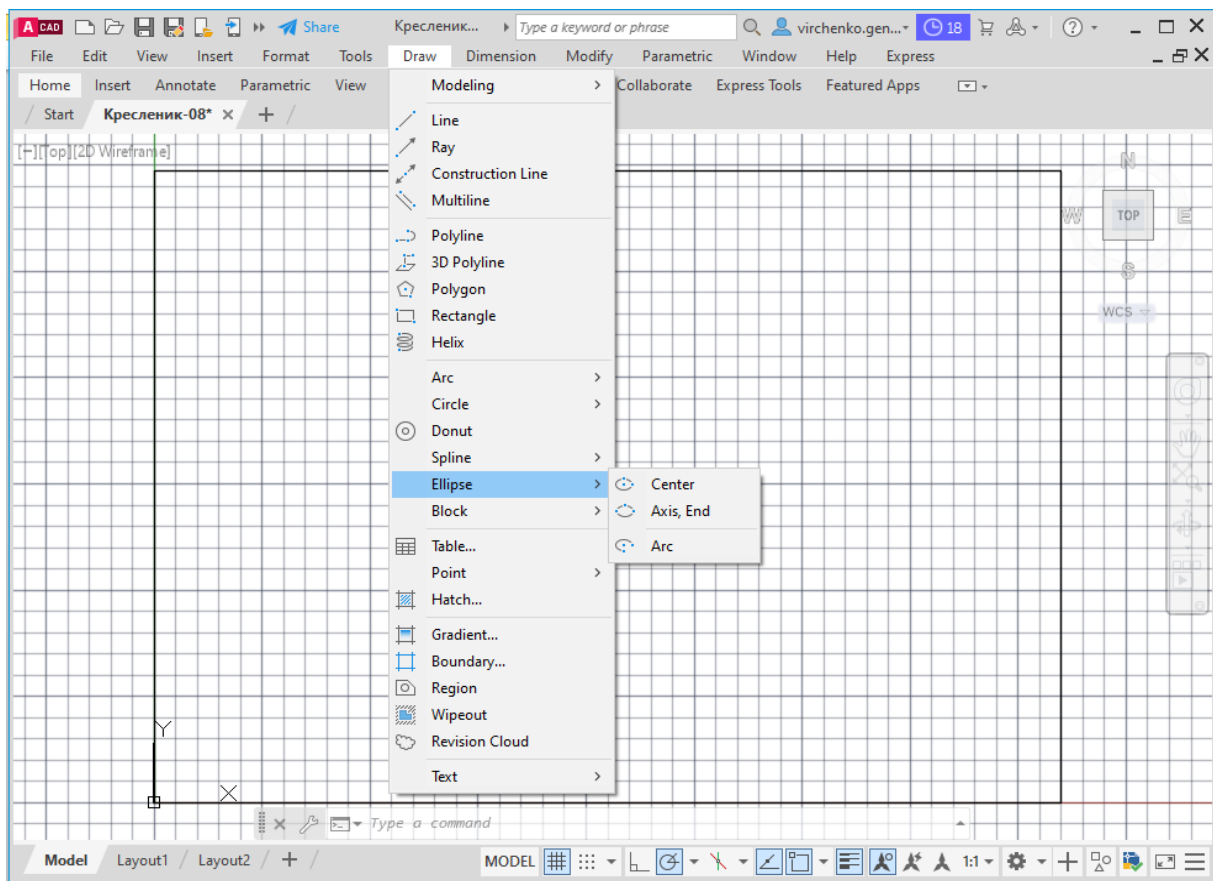


Fig. 3.30. AutoCAD menu for creating ellipses and their arcs

Examples of these constructions are shown in Fig. 3.31. The ellipse has center $C=(200, 150)$, the endpoint of its first axis at $(350, 150)$ and a semi-minor axis of 100. The arc is defined by the endpoints $(100, 150)$ and $(300, 150)$ of the first axis, the semi-minor axis length of 50, and the start and end angles of 0° and 270° , respectively.

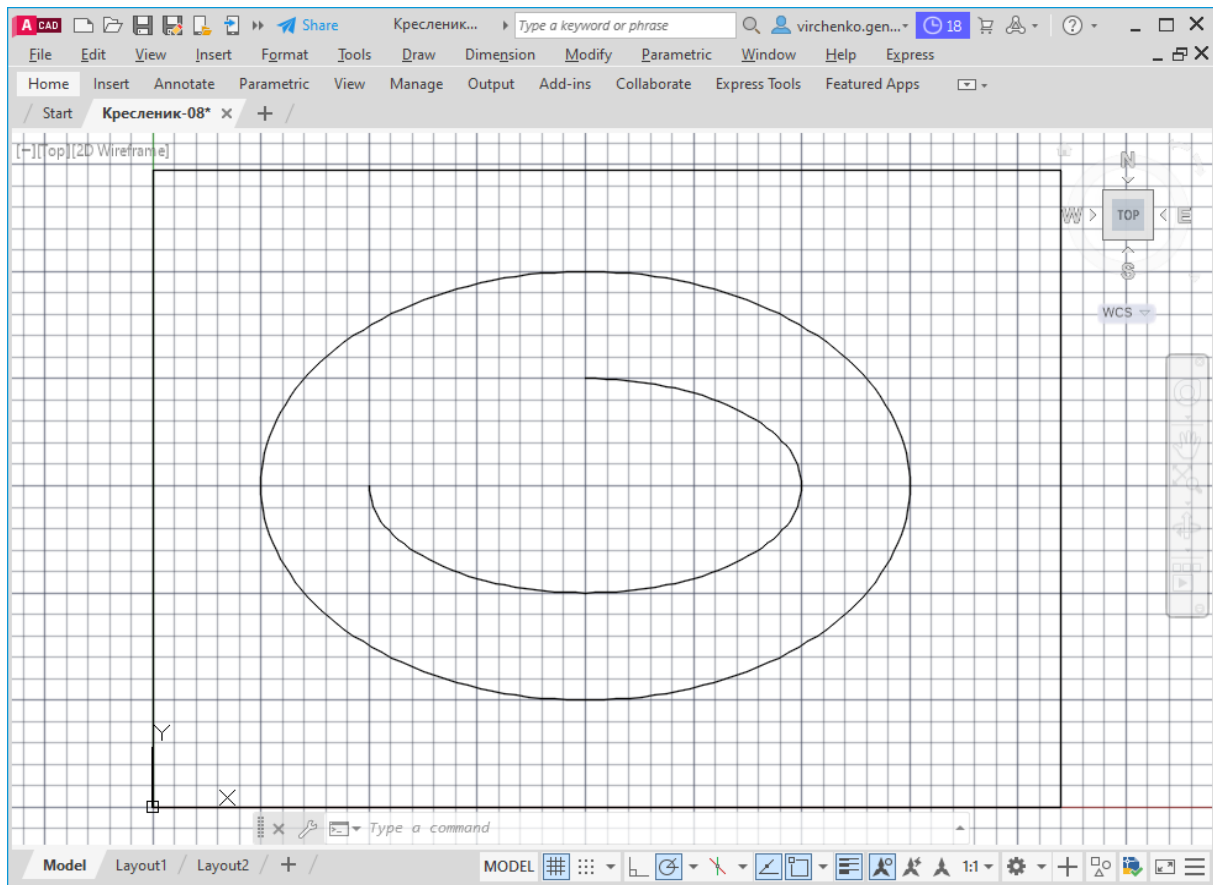


Fig. 3.31. Examples of constructing ellipses and elliptical arcs in AutoCAD

The geometric objects considered above are constructed using the following Visual Basic for Applications macro. The operation of this program is explained by the comments included within it.

```
Sub Ellipses()
    Dim ellipse As AcadEllipse      'ellipse
    Dim center(0 To 2) As Double    'x, y, z coordinates of the ellipse center
    Dim axis(0 To 2) As Double      'vector of the ellipse semi-axis
    Dim ratio As Double             'ratio of ellipse axes lengths
    Dim startAngle As Double, endAngle As Double 'start and end angles of the
    'ellipse arc
    center(0) = 200: center(1) = 150: center(2) = 0
    axis(0) = 150: axis(1) = 0: axis(2) = 0
    ratio = 2 / 3
    'drawing ellipse
    Set ellipse = ThisDrawing.ModelSpace.AddEllipse(center, axis, ratio)
    ellipse.Update
    'defining a whole ellipse
    center(0) = 200: center(1) = 150: center(2) = 0
    axis(0) = -100: axis(1) = 0: axis(2) = 0
    ratio = 0.5
    'drawing the whole ellipse
    Set ellipse = ThisDrawing.ModelSpace.AddEllipse(center, axis, ratio)
    ellipse.Update
End Sub
```

```

'transforming the whole ellipse into its arc
pi = 4 * Atn(1)           'value of the pi constant
ellipse.startAngle = 0
ellipse.endAngle = 1.5 * pi
ellipse.Update
End Sub

```

The operation of this program is explained in detail by the necessary comments provided in it.

3.2.7. Splines

Splines, originally used by draftsmen as flexible metal strips for drawing smooth curves, have long served as tools for constructing lines of complex shape. With the development of appropriate mathematical foundations, splines have become widely used in modern computer-aided geometric modeling across various CAD systems.

Among the most popular today are Non-Uniform Rational B-Splines (NURBS). These curves generalize points, straight-line segments, and second-order arcs, while also providing a powerful framework for representing composite parametric curves of higher degree.

In AutoCAD, splines are created using the sequence *Draw* → *Spline* (Fig. 3.32). Two construction methods are available: Fit points (where the spline interpolates a specified set of points) and Control Vertices (where the curve is formed based on given control points).

The first approach is illustrated in Figs. 3.33 and 3.34.

To begin, the required User Coordinate System (UCS) was created using *Tools* → *New UCS* → *Origin* with its origin set at $(x, y) = (10, 150)$.

Next, the sequence *Draw* → *Point* → *Multiple Point* was used to enter the following set of points: (0; 0), (10; 10,48), (30; 16,8), (60; 21,36), (80; 22,96), (120; 24), (160; 23,2), (200; 21,6), (240; 18,24), (280; 14,64), (320; 10,48), (360; 5,92), (400; 0).

These points represent a discrete approximation of the upper surface of an aerodynamic profile. Using the command *Draw* → *Spline* → *Fit points* a spline interpolating the given point set was generated (Fig. 3.34).

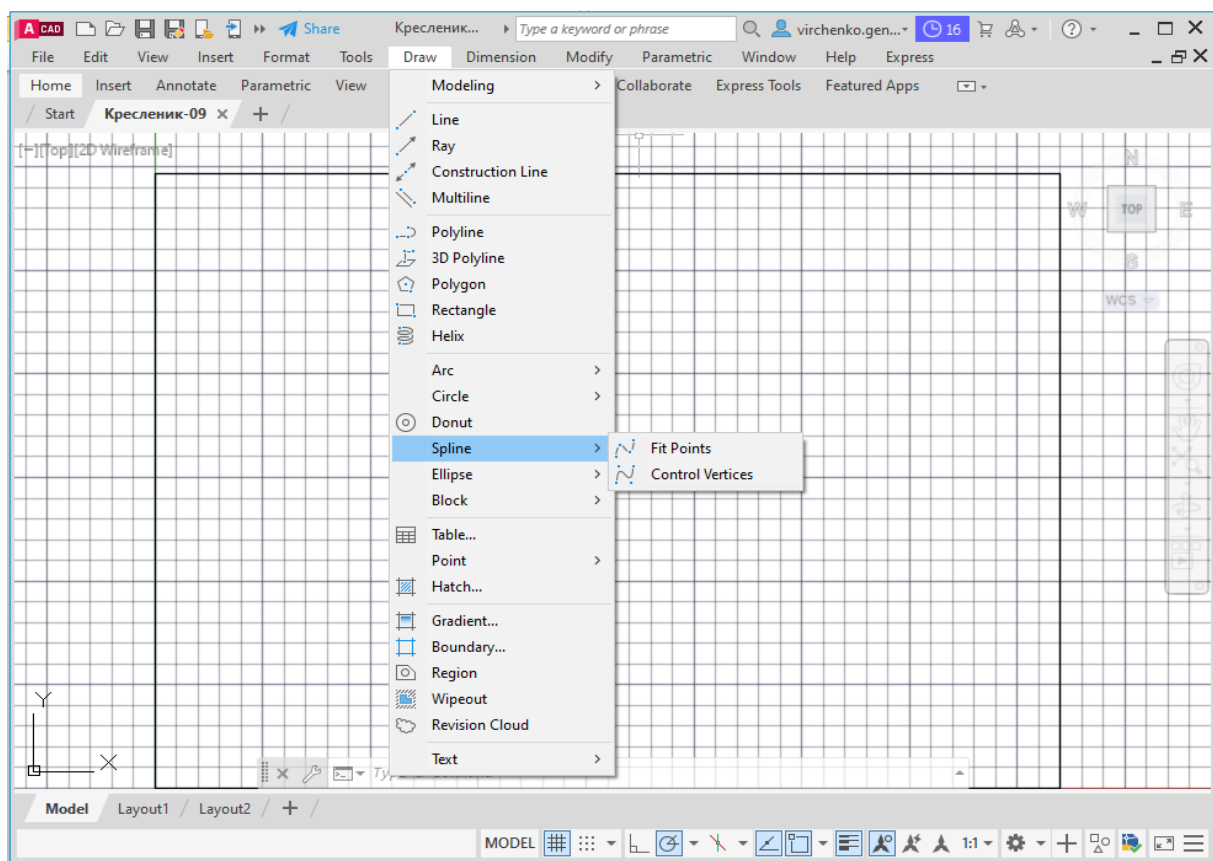


Fig. 3.32. AutoCAD menu for constructing splines

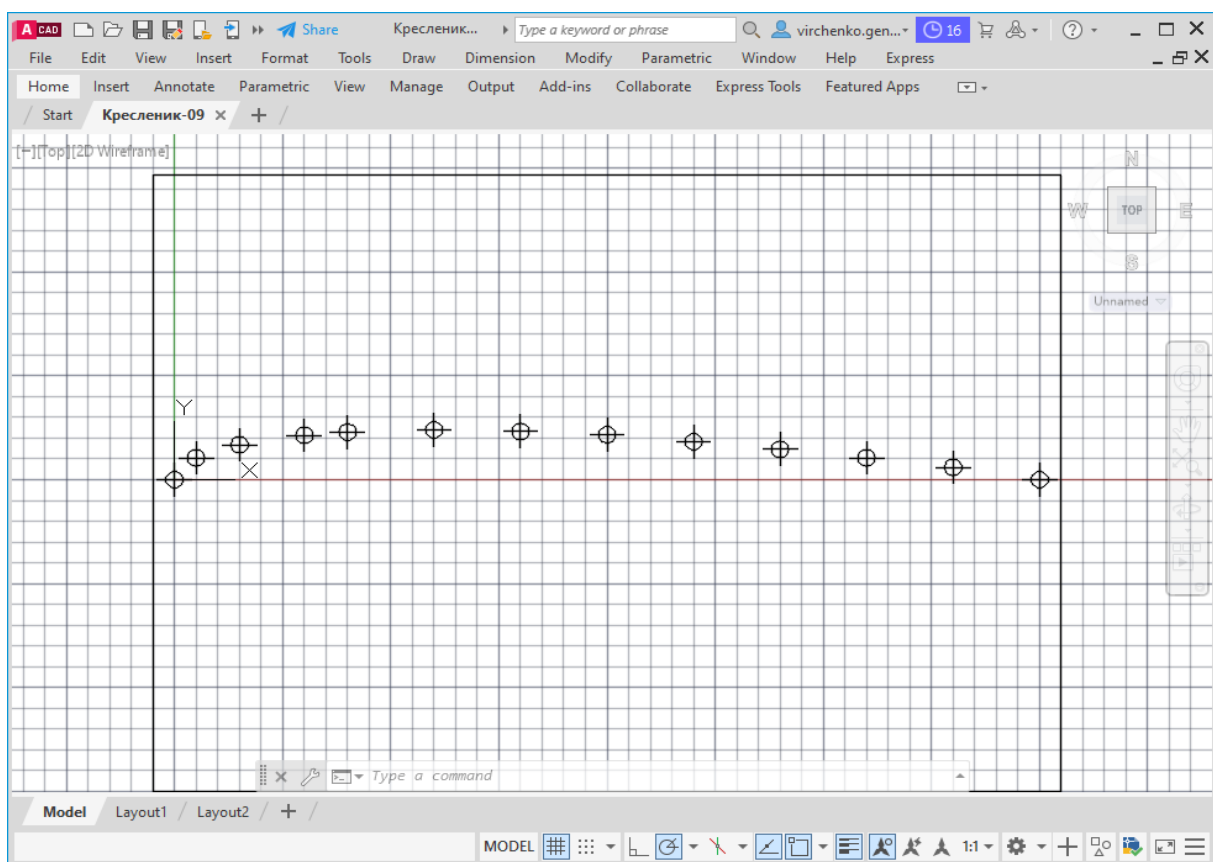


Fig. 3.33. Points for spline interpolation

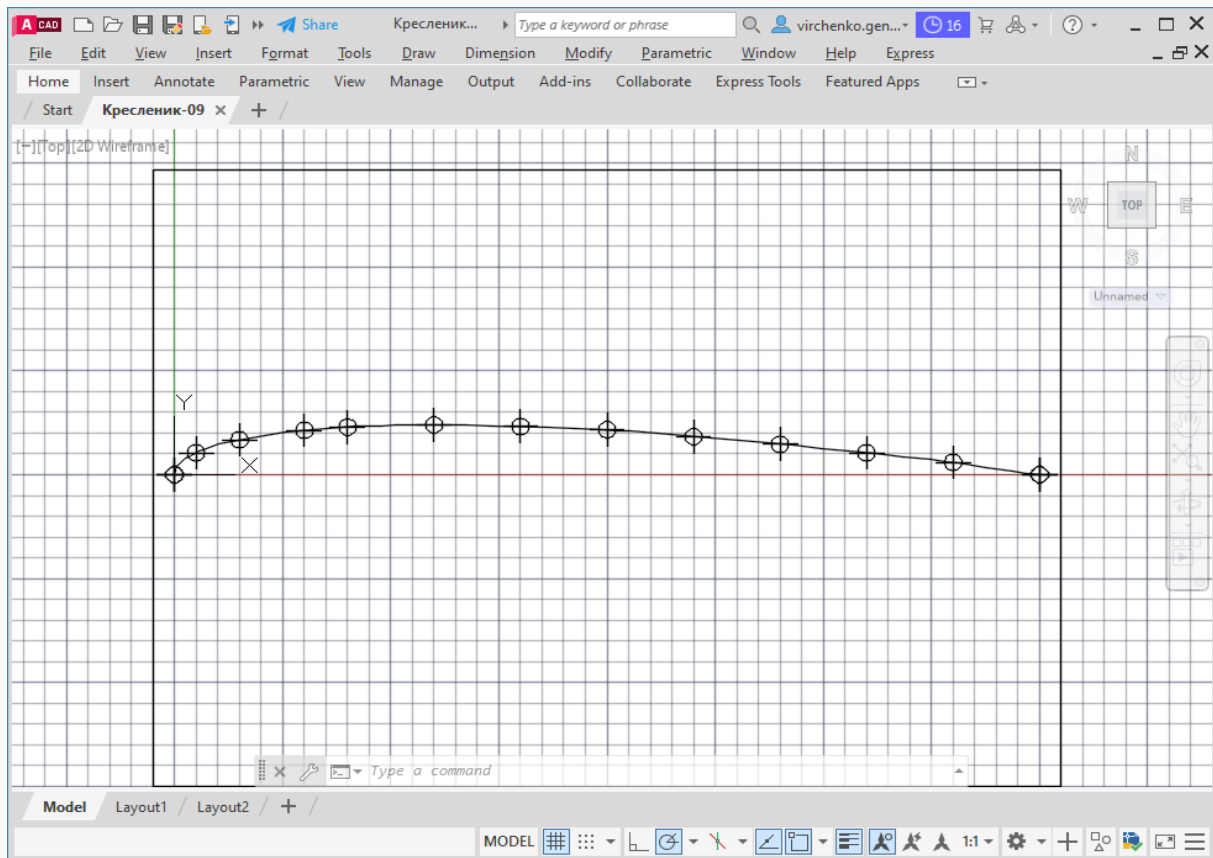


Рис.3.34. Spline constructed through fit points

The corresponding shape can also be produced using the Visual Basic for Applications macro provided below.

```
Sub Splines()
    Dim UCS As AcadUCS          'user coordinate system
    Dim origin(0 To 2) As Double, xAxis(0 To 2) As Double, _
    yAxis(0 To 2) As Double      'origin, x and y axes of the user coordinate
    'system
    Dim point As AcadPoint       'point
    Dim location(0 To 2) As Double 'array of coordinates x, y, z of a point
    Dim Points(0 To 38) As Double 'array of coordinates x, y, z of 13 points
    'aerodynamic profile (spline)
    Dim tangStart(0 To 2) As Double 'tangent vector at the start of the spline
    Dim tangEnd(0 To 2) As Double 'tangent vector at the end of the spline
    Dim UCSpnt As Variant, UCSpoints(0 To 38) As Double 'for transformation of
    'coordinates from acUCS to acWorld
    'definition of the user coordinate system
    origin(0) = 10: origin(1) = 150: origin(2) = 0
    xAxis(0) = 20: xAxis(1) = 150: xAxis(2) = 0
    yAxis(0) = 10: yAxis(1) = 160: yAxis(2) = 0
    Set UCS = ThisDrawing.UserCoordinateSystems.Add(origin, xAxis, yAxis, "UCS1")
    'display of user coordinate system and its activation
    ThisDrawing.ActiveViewport.UCSIconAtOrigin = True
    ThisDrawing.ActiveViewport.UCSIconOn = True
    ThisDrawing.ActiveUCS = UCS
End Sub
```

```

'x, y, z coordinates of 13 points of the aerodynamic profile (in plane z=0)
Points(0) = 0:    Points(1) = 0:    Points(2) = 0
Points(3) = 10:   Points(4) = 10.48: Points(5) = 0
Points(6) = 30:   Points(7) = 16.8:  Points(8) = 0
Points(9) = 60:   Points(10) = 21.36: Points(11) = 0
Points(12) = 80:  Points(13) = 22.96: Points(14) = 0
Points(15) = 120: Points(16) = 24:    Points(17) = 0
Points(18) = 160: Points(19) = 23.2:  Points(20) = 0
Points(21) = 200: Points(22) = 21.6:  Points(23) = 0
Points(24) = 240: Points(25) = 18.24: Points(26) = 0
Points(27) = 280: Points(28) = 14.64: Points(29) = 0
Points(30) = 320: Points(31) = 10.48: Points(32) = 0
Points(33) = 360: Points(34) = 5.92:  Points(35) = 0
Points(36) = 400: Points(37) = 0:    Points(38) = 0
'spline tangent vectors at its endpoints
tangStart(0) = 0: tangStart(1) = 2: tangStart(2) = 0
tangEnd(0) = 4: tangEnd(1) = -0.5: tangEnd(2) = 0
'point display format
ThisDrawing.SetVariable "PDMODE", 34
ThisDrawing.SetVariable "PDSIZE", -5
'forming 13 spline points
For i = 0 To 36 Step 3
    location(0) = Points(i)      'x
    location(1) = Points(i + 1)  'y
    location(2) = Points(i + 2)  'z
    'coordinate transformation: location from acUCS to acWorld
    UCSpnt = ThisDrawing.Utility.TranslateCoordinates(location, acUCS, acWorld, _
    False)
    'point creation
    Set point = ThisDrawing.ModelSpace.AddPoint(UCSpnt)
    'filling the UCSpoints array
    UCSpoints(i) = UCSpoint(0)    'x
    UCSpoints(i + 1) = UCSpoint(1) 'y
    UCSpoints(i + 2) = UCSpoint(2) 'z
Next i
'drawing the spline
Set spline = ThisDrawing.ModelSpace.AddSpline(UCSpoints, tangStart, tangEnd)
End Sub

```

A new element introduced in this macro is the use of a user-defined coordinate system, UCS1, which allows the aerodynamic profile to be constructed within its own local coordinate frame in the central area of the A3 drawing. The relevant clarifications are included within the macro comments. Unlike the spline shown in Fig. 3.34, the macro also defines tangent directions at the curve's endpoints.

Figures 3.35 ... 3.37 illustrate spline construction using control vertices. For this approach, four control points were defined using: *Draw* → *Point* → *Multiple Point* with the following coordinates: (0, 0), (0, 40), (200, 40), (400, 0).

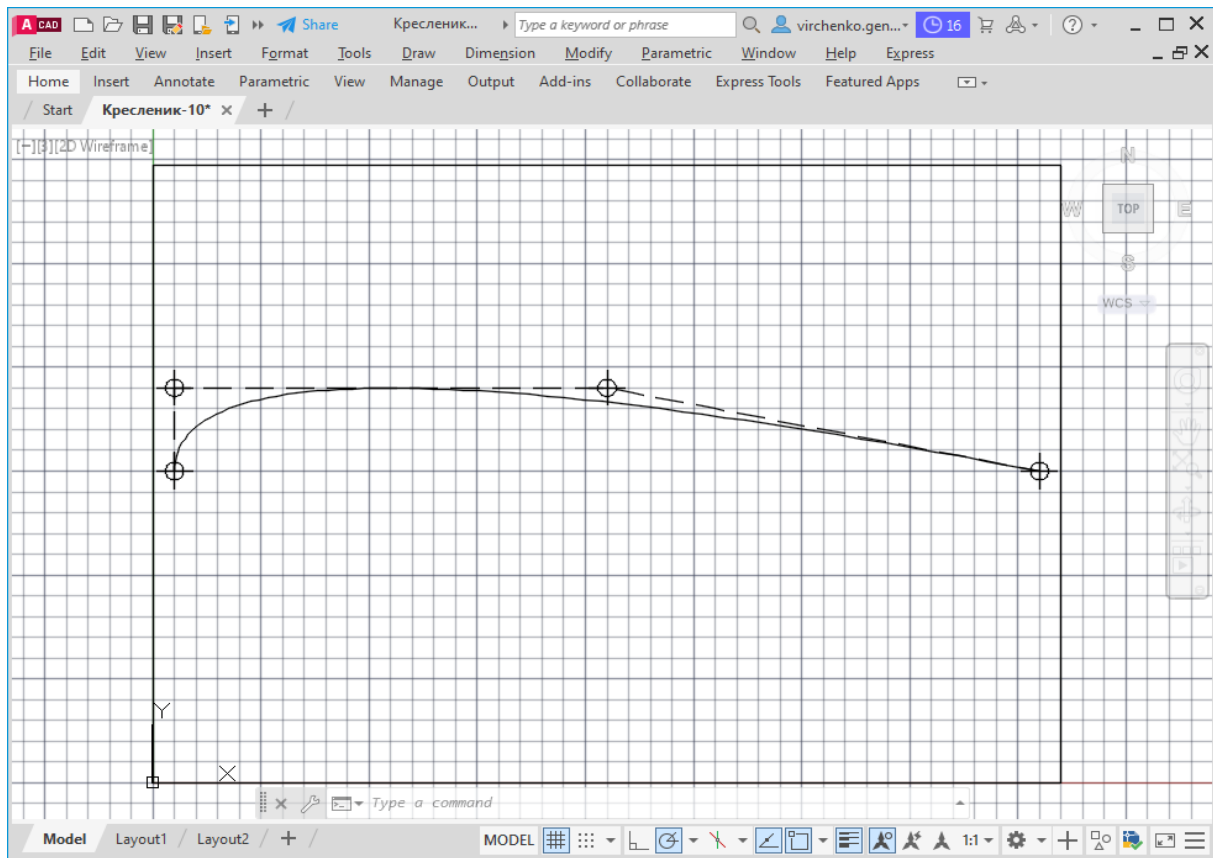


Fig. 3.35. Four control vertices and resulting first- and second-degree splines

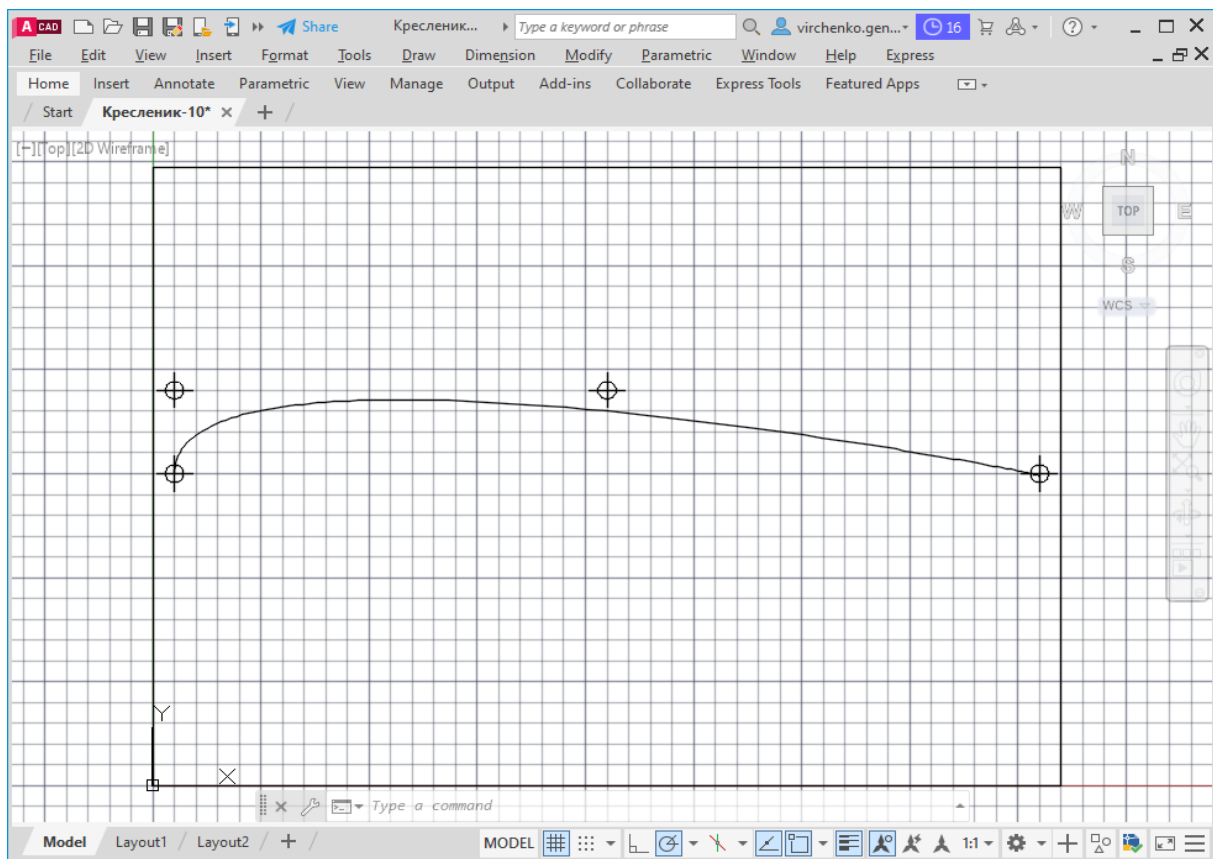


Fig. 3.36. Four control vertices and resulting third-degree spline

Next, using the sequence *Draw* → *Spline* → *Control Vertices*, and selecting the appropriate option within the command, splines of degree 1 and 2 (Fig. 3.35) and degree 3 (Fig. 3.36) were constructed. In AutoCAD, the term degree refers to the polynomial degree of the spline. The first curve (shown with dashed segments) is a polyline with three straight sections, the second is a combination of quadratic curves, and the third is a cubic spline. Clearly, changing the spline degree affects the resulting shape even when the control vertices remain fixed.

Fig. 3.37 illustrates how the shape of a third-degree spline can be modified by adjusting the Weight parameters of the second and third control vertices in the *Properties* menu (Fig. 3.38).

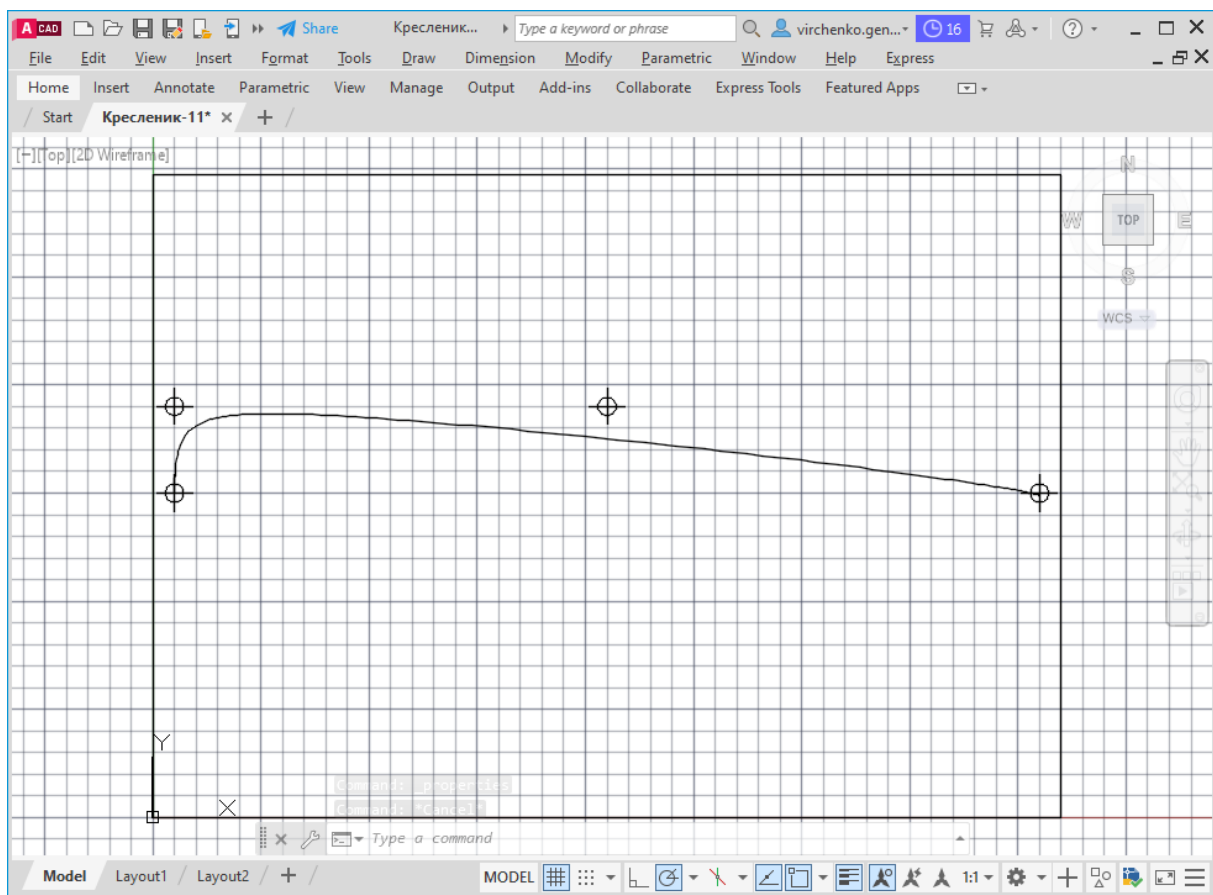


Fig. 3.37. Influence of vertex weights on spline shape

Compared with the previous figure, the curve moves closer to the second control point and farther from the third. Thus, the weight of the second control point has increased, while that of the third has decreased. In this example, the weight values were changed from 1 to 2 and 0.5, respectively.

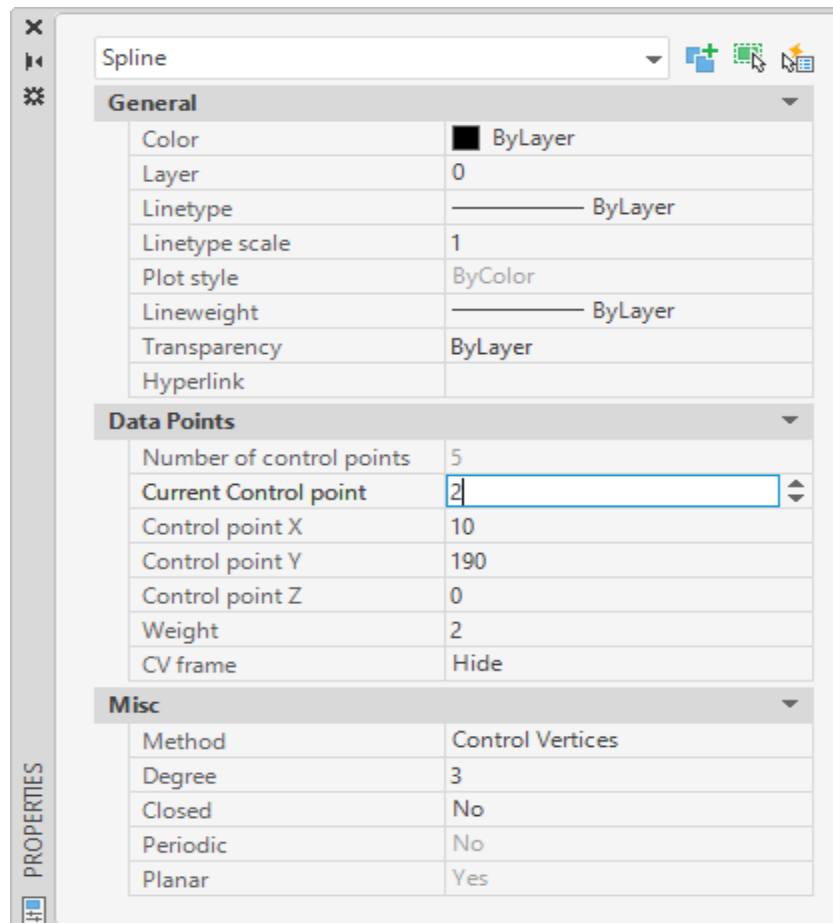


Fig. 3.38. Modifying spline vertex weights using the Properties menu

The following Visual Basic for Applications macro performs the spline constructions based on control vertices described above.

```
Sub Splines2()
    Dim UCS As AcadUCS          'user coordinate system
    Dim origin(0 To 2) As Double, xAxis(0 To 2) As Double, yAxis(0 To 2) As Double
    'origin, x and y axes of the user coordinate system
    Dim point As AcadPoint      'point
    Dim location(0 To 2) As Double 'array of coordinates x, y, z of a point
    Dim Points(0 To 11) As Double 'array of coordinates x, y, z of 4 control
    'vertices aerodynamic profile (spline)
    Dim UCSpnt As Variant, UCSpoints(0 To 11) As Double 'for coordinate
    'transformation from acUCS to acWorld
    origin(0) = 10: origin(1) = 150: origin(2) = 0
    xAxis(0) = 20: xAxis(1) = 150: xAxis(2) = 0
    yAxis(0) = 10: yAxis(1) = 160: yAxis(2) = 0
    Set UCS = ThisDrawing.UserCoordinateSystems.Add(origin, xAxis, yAxis, "UCS1")
    'display of the user coordinate system and its activation
    ThisDrawing.ActiveViewport.UCSIconAtOrigin = True
    ThisDrawing.ActiveViewport.UCSIconOn = True
    ThisDrawing.ActiveUCS = UCS
End Sub
```

```

'x, y, z coordinates of the array of 4 control vertices of the spline (in
'plane z=0)
Points(0) = 0:   Points(1) = 0:   Points(2) = 0
Points(3) = 0:   Points(4) = 40:  Points(5) = 0
Points(6) = 200: Points(7) = 40:  Points(8) = 0
Points(9) = 400: Points(10) = 0:  Points(11) = 0
'point display format
ThisDrawing.SetVariable "PDMODE", 34
ThisDrawing.SetVariable "PDSIZE", -5
'forming 4 control points
For i = 0 To 9 Step 3
    location(0) = Points(i)        'x
    location(1) = Points(i + 1)    'y
    location(2) = Points(i + 2)    'z
    'coordinate transformation: location from acUCS to acWorld
    UCSpnt = ThisDrawing.Utility.TranslateCoordinates(location, acUCS, acWorld, _
    False)
    'creating a point
    Set point = ThisDrawing.ModelSpace.AddPoint(UCSpnt)
    'filling the UCSpoints array
    UCSpoints(i) = UCSpoint(0)
    UCSpoints(i + 1) = UCSpoint(1)
    UCSpoints(i + 2) = UCSpoint(2)
Next i
'drawing the spline
ThisDrawing.SendCommand "SPLINE M CV D 3  0,0 0,40 200,40 400,0  "
'ThisDrawing.SendCommand "_SPLINE C YB 0,0 0,40 200,40 400,0  "
End Sub

```

This program is generally similar to the previous one; therefore, we emphasize only the key differences. The main distinction lies in the final block, where the spline is constructed using the previously discussed SendCommand method.

The arguments of the SPLINE command can be examined during its interactive execution in AutoCAD. Note that M CV selects the control-vertex construction mode, and D 3 specifies a third-degree curve. The remaining arguments are the coordinates of the vertices, and the command is completed by entering two final spaces.

Based on the above material, we have examined practical examples of spline generation in AutoCAD and Visual Basic for Applications. Thus, subsection 3.2 has presented methods for constructing basic geometric primitives – points, line segments, rectangles, polygons, circles, ellipses and their arcs, and splines – using both direct AutoCAD tools and Visual Basic for Applications macros.

The consistent generalization of these primitives provides a solid foundation for learning, enabling students to acquire strong practical skills for effective future use.

3.3. Modification of Geometric Objects

Editing previously created figures is an essential component of automated geometric modeling. Such operations include deleting and copying various primitives, translating, rotating, mirroring, scaling, splitting, joining, and modifying properties such as color, linetype, and lineweight.

Some of these modifications – such as reassigning color, linetype, and lineweight – have already been encountered in the previous subsection.

In this part of the textbook, we examine several plane coordinate transformations Translation, Rotation, and Scaling, applying them both directly in AutoCAD and through Visual Basic for Applications.

Figure 3.39 illustrates the elements of AutoCAD's Modify menu, which contains the commands used to perform the transformations and other editing operations mentioned above.

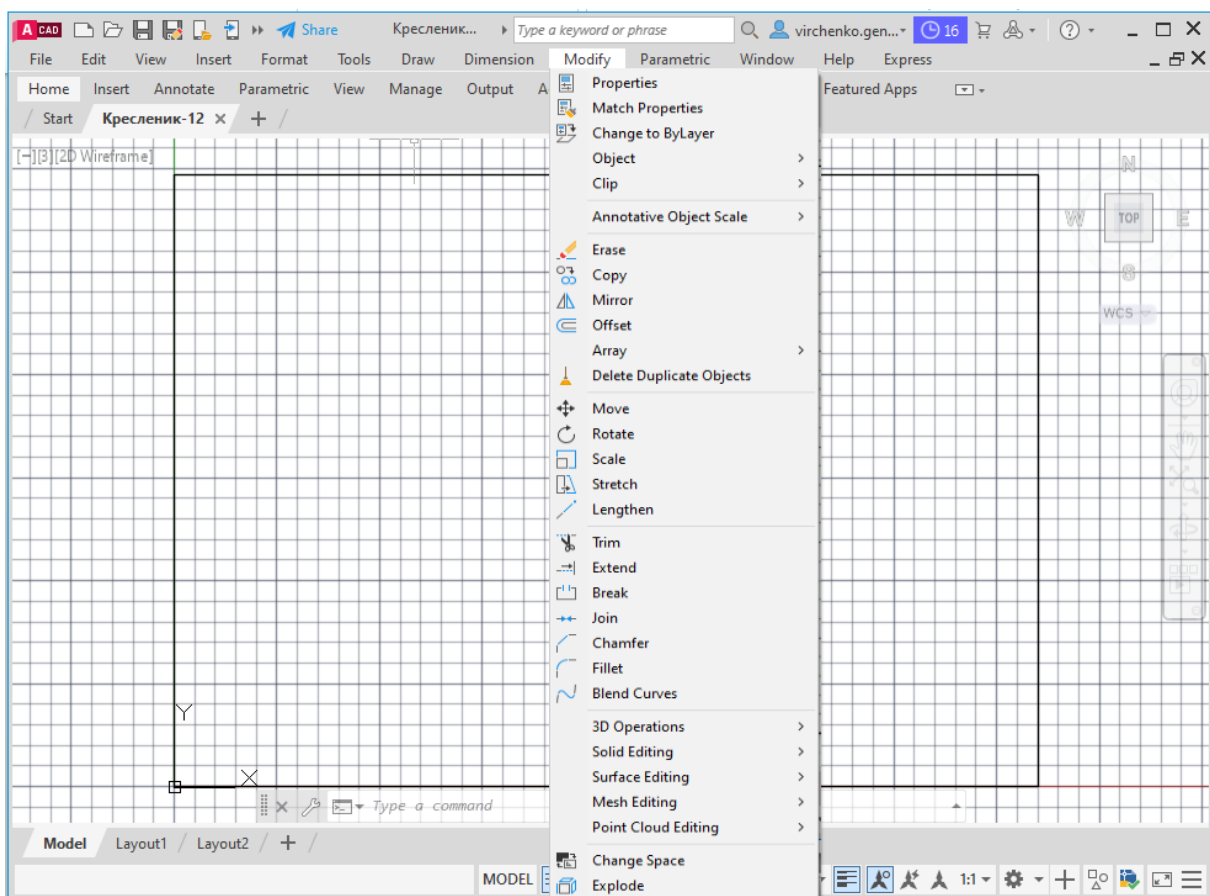


Fig. 3.39. AutoCAD Modify menu

3.3.1. Translation

Figure 3.40 shows an example of translating a circle of radius 30, originally centered at (100,100), by the vector (150, 100) using Visual Basic for Applications.

For clarity, the original circle is retained in the drawing and displayed with a dashed outline.

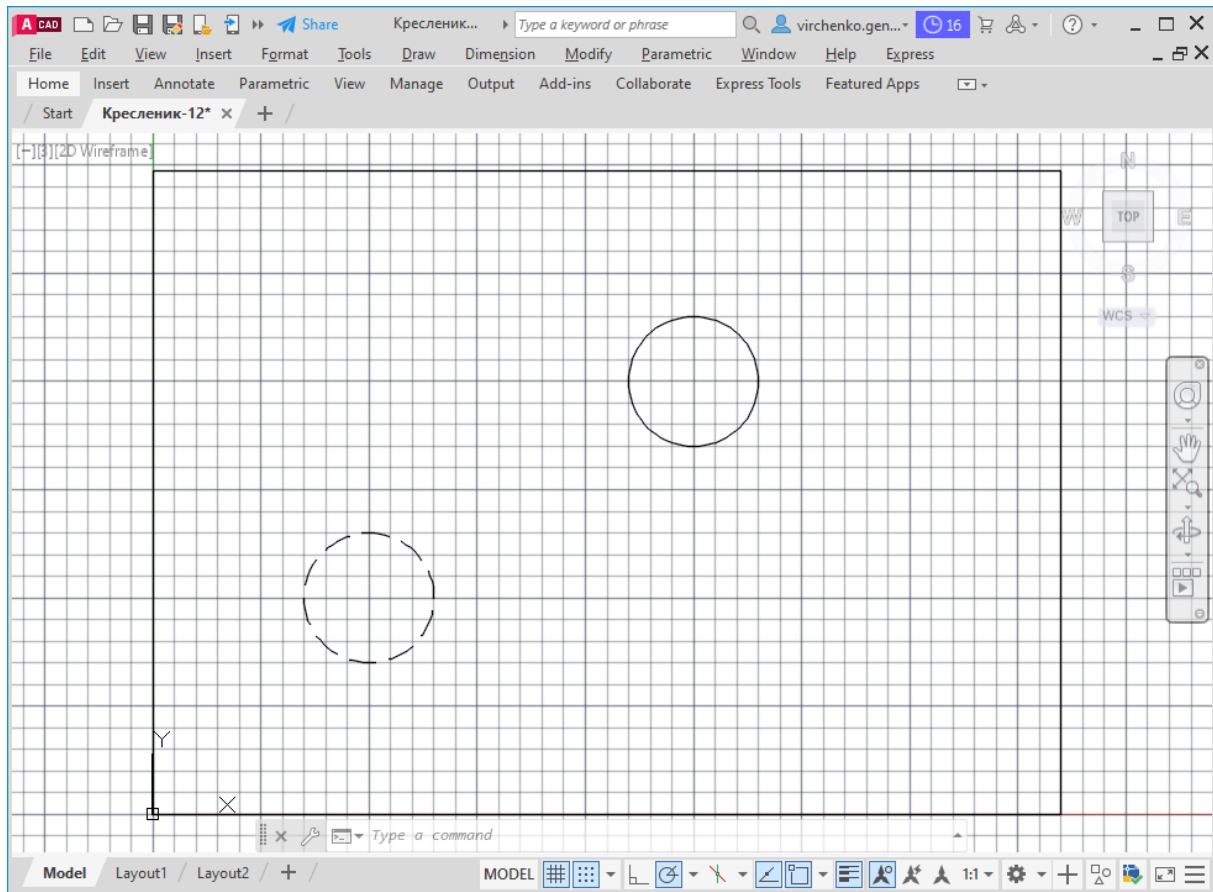


Fig. 3.40. Translation of a circle

This transformation is performed by the Visual Basic for Applications macro presented below.

```
Sub Move_Circle()  
    Dim circl As AcadCircle           'circle  
    Dim center(0 To 2) As Double      'circle center  
    Dim radius As Double              'circle radius  
    Dim point1(0 To 2) As Double, point2(0 To 2) As Double 'points defining the  
    'translation vector  
    'x, y, z coordinates of the center and circle radius  
    center(0) = 100: center(1) = 100: center(2) = 0:  
    radius = 30
```

```

'drawing the circle
Set circl = ThisDrawing.ModelSpace.AddCircle(center, radius)
circl.Update
'translation vector
point1(0) = 0:   point1(1) = 0:   point1(2) = 0
point2(0) = 150: point2(1) = 100: point2(2) = 0
'pause before performing the translation
MsgBox "Press OK to continue"
'circle translation and regeneration
circl.Move point1, point2
circl.Update
End Sub

```

The operation of the macro is based on concepts discussed earlier, and additional clarifications are provided in the comments within the code.

Before executing the translation, the macro displays a dialog window (Fig. 3.41), which introduces a brief pause for the user.



Fig. 3.41. Message displayed to the user

Another widely used geometric transformation is rotation, which is discussed in the following subsection.

3.3.2. Rotation

Figure 3.42 demonstrates the use of the *Modify* → *Rotate* command from the drop-down menu to rotate a rectangle with a length of 200 and a height of 100 about its center $O_1=(200, 150)$ by an angle of 90° counterclockwise. For clarity, the original position of the rectangle is shown with a dashed outline, allowing the reader to visually compare the initial and the rotated configurations.

This transformation of the geometric object can also be carried out using the Visual Basic for Applications macro provided below, which performs the same operation programmatically and illustrates how the rotation procedure can be automated.

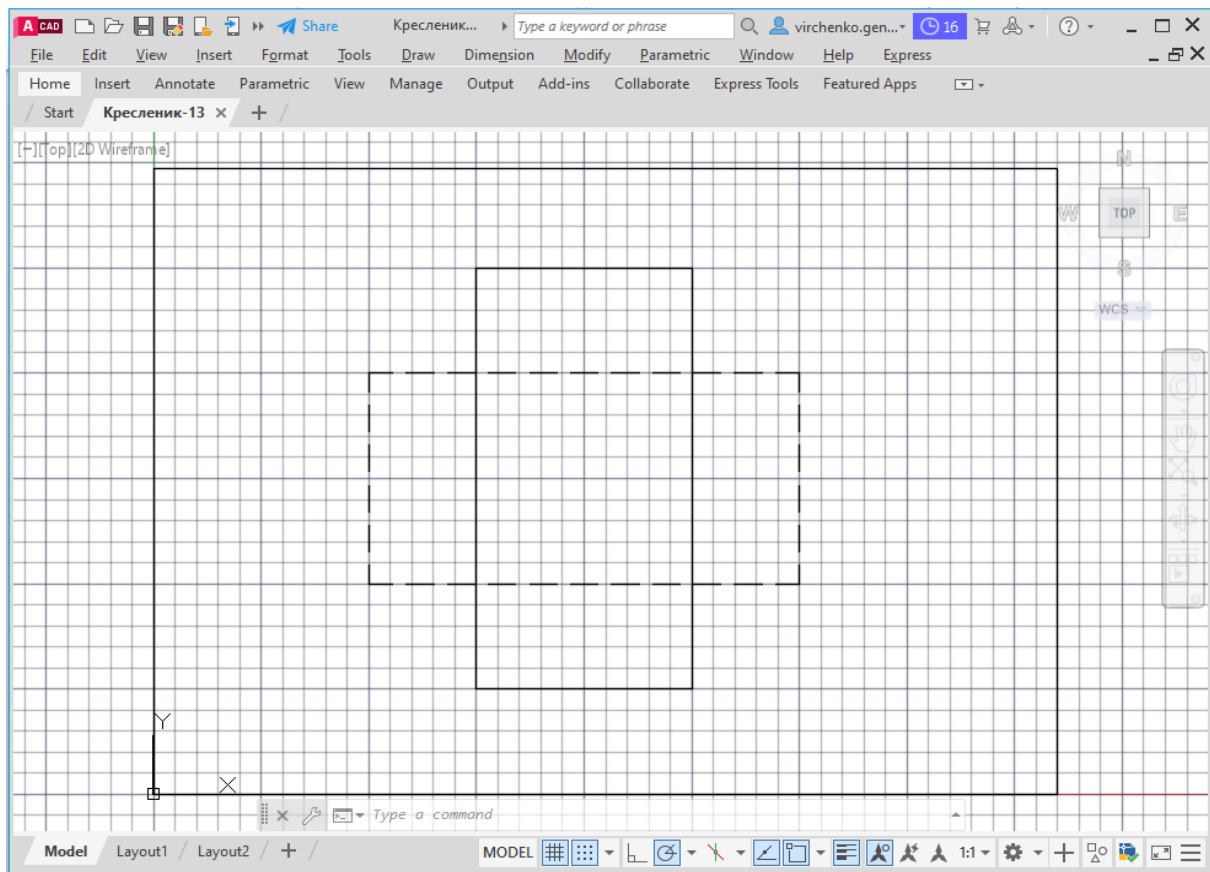


Fig. 3.42. Rotation of a rectangle using AutoCAD tools

```

Sub Rotate_Polyline()
    Dim polyline As AcadPolyline      'polyline
    Dim Points(0 To 11) As Double     'array of coordinates x, y, z of rectangle
    'vertices
    Dim basePoint(0 To 2) As Double   'coordinates x, y, z of the rotation center
    Dim rotationAngle As Double       'rotation angle
    'x, y, z initial coordinates of rectangle vertices in the plane z=0
    Points(0) = 100: Points(1) = 100: Points(2) = 0
    Points(3) = 300: Points(4) = 100: Points(5) = 0
    Points(6) = 300: Points(7) = 200: Points(8) = 0
    Points(9) = 100: Points(10) = 200: Points(11) = 0
    'drawing the rectangle
    Set polyline = ThisDrawing.ModelSpace.AddPolyline(Points)
    polyline.Closed = True
    polyline.Update
    'pause before rotating the rectangle
    MsgBox "Press OK to continue"
    'rotation center
    basePoint(0) = 200: basePoint(1) = 150: basePoint(2) = 0
    'rotation angle
    pi = 4 * Atn(1)                    'value of the pi constant
    rotationAngle = pi / 2              '90 degrees
    'rotating the rectangle and updating it
    polyline.Rotate basePoint, rotationAngle
    polyline.Update
End Sub

```

The presented program requires no additional explanation, as the included comments fully describe its operation.

The next common coordinate transformation for planar figures is scaling, which is discussed in the following section.

3.3.3. Scaling

Figure 3.43 illustrates the use of the *Modify* → *Scale* command from the drop-down menu to resize a rectangle originally measuring 200 units in length and 100 units in height. The scaling is performed with respect to its center point $O_1=(200, 150)$ using a scale factor of 1,5. For clarity, the initial position of the rectangle is shown with a dashed outline.

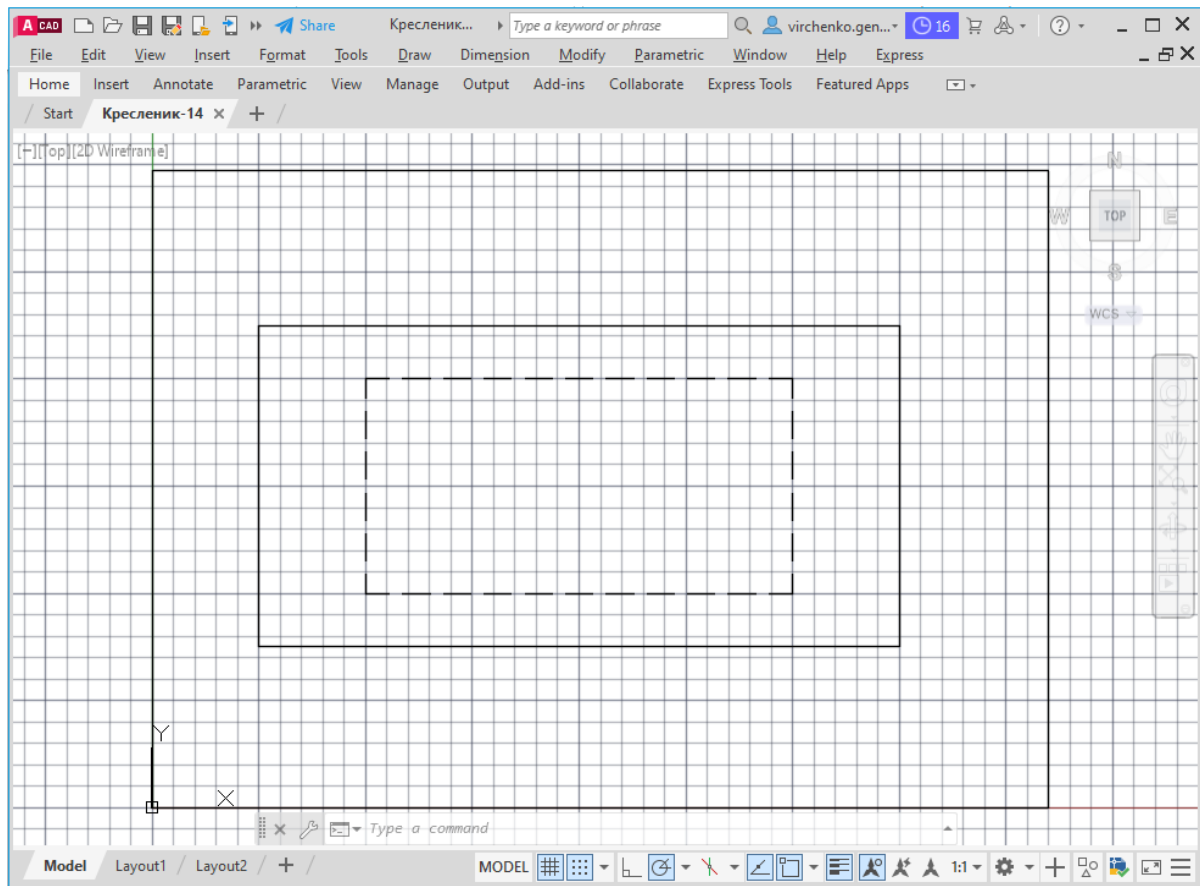


Fig. 3.43. Scaling a rectangle using AutoCAD

The coordinate grid displayed in the drawing area makes it easy to verify the correct execution of the described transformation.

The transformations applied to the analyzed figure can also be performed using the Visual Basic for Applications macro provided below.

```

Sub Scale_Polyline()
    Dim polyline As AcadPolyline      'polyline
    Dim Points(0 To 11) As Double     'array of coordinates x, y, z of rectangle
    'vertices
    Dim basePoint(0 To 2) As Double   'coordinates x, y, z of the scaling center
    Dim scaleFactor As Double          'scaling factor
    'x, y, z initial coordinates of rectangle vertices in the plane z=0
    Points(0) = 100: Points(1) = 100: Points(2) = 0
    Points(3) = 300: Points(4) = 100: Points(5) = 0
    Points(6) = 300: Points(7) = 200: Points(8) = 0
    Points(9) = 100: Points(10) = 200: Points(11) = 0
    'drawing the rectangle
    Set polyline = ThisDrawing.ModelSpace.AddPolyline(Points)
    polyline.Closed = True
    polyline.Update
    'pause before scaling the rectangle
    MsgBox "Press OK to continue"
    'scaling center
    basePoint(0) = 200: basePoint(1) = 150: basePoint(2) = 0
    'scaling factor
    scaleFactor = 1.5
    'scaling the rectangle and updating it
    polyline.ScaleEntity basePoint, scaleFactor
    polyline.Update
End Sub

```

The operation of the program is fully explained by the comments embedded in the code.

Using negative scale factors allows the user to perform a mirror transformation (*Mirror* in AutoCAD terminology). A corresponding example – reflecting the trapezoid shown in Figure 3.17 about its upper base – is presented in Figure 3.44.

The same result can be obtained by extending the macro described in Section 3.2.4 with the additional lines shown below.

```

polyline.Update
'axis of symmetry
Dim point1(0 To 2) As Double, point2(0 To 2) As Double
point1(0) = 150: point1(1) = 150: point1(2) = 0
point2(0) = 250: point2(1) = 150: point2(2) = 0
Dim mirrorPolyline As AcadPolyline 'mirrored trapezoid
'mirror operation
Set mirrorPolyline = polyline.Mirror(point1, point2)
mirrorPolyline.Update

```

If removal of the original trapezoid is required, this can be done using the command *polyline.Delete*.

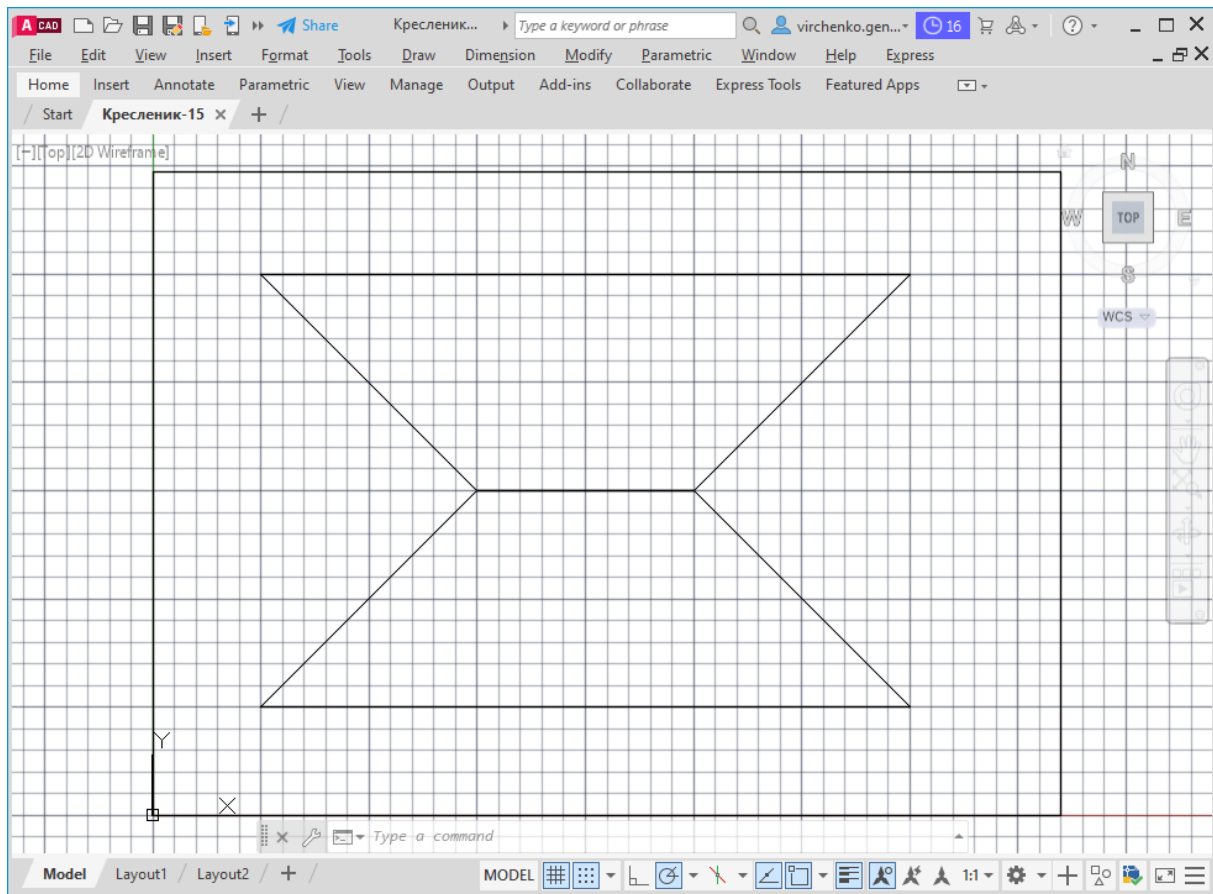


Fig. 3.44. Mirroring a trapezoid using AutoCAD

It should be noted that AutoCAD drop-down menu sequences such as *Modify* → *Rotate* та *Modify* → *Scale* allow the user to select whether the original objects should be deleted or retained after the transformation.

3.3.4. Compositions of Coordinate Transformations

In geometric modeling, operations performed on created objects often involve combinations of several transformations. When modifying a figure, such *combinations* may include translation, rotation, scaling, reflection, and other coordinate transformations.

For example, consider a primitive in a rectangular coordinate system Oxy . If it is required to perform the following sequence of operations: scale the object with factors (S_x, S_y) relative to a point (x_1, y_1) , rotate it by an angle γ about the same point, and then translate it by the vector (T_x, T_y) , the resulting modification M is represented by the following product of matrices in homogeneous coordinates:

$$M = T(-x_1, -y_1) \cdot S(Sx, Sy) \cdot R(\gamma) \cdot T(x_1 + Tx, y_1 + Ty). \quad (3.9)$$

This approach is illustrated in Figure 3.45. The initial square, shown with a dashed outline and defined by the vertex coordinates (50, 50), (250, 50), (250, 250), and (50, 250), is first scaled by a factor of 0.5 relative to its center at (150, 150). The scaled figure is then rotated by 45° counterclockwise, after which it undergoes a translation by the vector (200, 0).

For better visual perception, the successive objects in the figure are differentiated by line type and color.

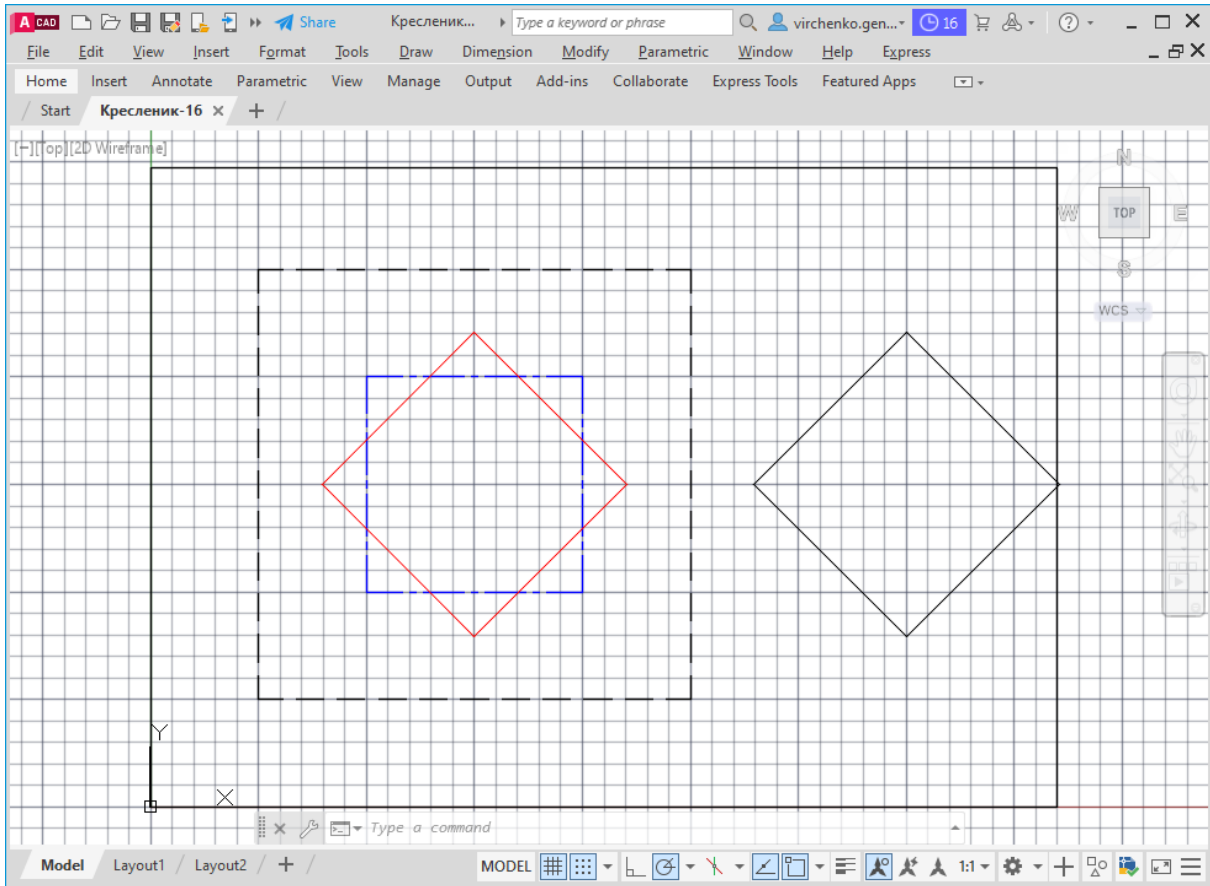


Fig. 3.45. Example of a composition of coordinate transformations

Based on expression (3.9), we compute for the analyzed case:

$$M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -x_1 - y_1 & 1 \end{bmatrix} \cdot \begin{bmatrix} Sx & 0 & 0 \\ 0 & Sy & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ x_1 + Tx & y_1 + Ty & 1 \end{bmatrix} =$$

$$\begin{aligned}
&= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -150 & -150 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0,5 & 0 & 0 \\ 0 & 0,5 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \sqrt{2}/2 & \sqrt{2}/2 & 0 \\ -\sqrt{2}/2 & \sqrt{2}/2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 150+200 & 150+0 & 1 \end{bmatrix} = \\
&= \begin{bmatrix} \sqrt{2}/4 & \sqrt{2}/4 & 0 \\ -\sqrt{2}/4 & \sqrt{2}/4 & 0 \\ 350 & -75\sqrt{2} + 150 & 1 \end{bmatrix}. \quad (3.10)
\end{aligned}$$

The Visual Basic for Applications macro presented below performs the described composition of coordinate transformations based on the matrix (3.10).

```

Sub Modification()
    Dim polyline As AcadPolyline 'polyline
    Dim Points(0 To 11) As Double 'array of coordinates x, y, z of
    'polyline vertices
    'x, y, z coordinates of square vertices in the plane z=0
    Points(0) = 50: Points(1) = 50: Points(2) = 0
    Points(3) = 250: Points(4) = 50: Points(5) = 0
    Points(6) = 250: Points(7) = 250: Points(8) = 0
    Points(9) = 50: Points(10) = 250: Points(11) = 0
    'drawing the initial square
    Set polyline = ThisDrawing.ModelSpace.AddPolyline(Points)
    polyline.Closed = True
    polyline.Update
    'square modification matrix
    Dim modMat(0 To 3, 0 To 3) As Double
    modMat(0, 0) = Sqr(2) / 4: modMat(0, 1) = -Sqr(2) / 4: _
    modMat(0, 2) = 0: modMat(0, 3) = 350
    modMat(1, 0) = Sqr(2) / 4: modMat(1, 1) = Sqr(2) / 4: _
    modMat(1, 2) = 0: modMat(1, 3) = -75 * Sqr(2) + 150
    modMat(2, 0) = 0: modMat(2, 1) = 0: modMat(2, 2) = 0.5: modMat(2, 3) = 0
    modMat(3, 0) = 0: modMat(3, 1) = 0: modMat(3, 2) = 0: modMat(3, 3) = 1
    'pause before modifying the square
    MsgBox "Press OK to continue"
    'square modification
    polyline.TransformBy modMat
    polyline.Update
End Sub

```

In the next subsection of the textbook, a systematised overview of the methods of modifying various figures in the AutoCAD and Visual Basic for Applications environments – already discussed earlier – is provided.

3.3.5. Properties and Methods of Geometric Objects in Visual Basic for Applications

From the object-oriented standpoint, each figure has a certain set of associated *properties* and *methods*. The former represent its characteristics – for example, color, linetype, length, area, etc.

The latter implement the actions that can be performed on the object, such as parallel translation, rotation, scaling, symmetry, copying, deletion, and so on.

In AutoCAD, access to the properties of a selected figure is provided through the *Properties* context menu (see Fig. 3.16). This menu allows not only viewing the required parameters and characteristics but also modifying them when necessary. The role of methods – translation, rotation, scaling, symmetry, deletion, and others – is performed by the corresponding AutoCAD commands.

In Visual Basic for Applications macros, properties and methods are grouped in a single context menu, which is invoked automatically by typing a period after the name of an object variable.

Specific examples of using such polygon properties as area and perimeter were examined in subsection 3.2.4; color, linetype, and lineweight for a circle – in subsection 3.2.5; and start and end angles for an ellipse arc – in subsection 3.2.6. Let us note that in the latter case, modifying the figure's properties changes its length. The use of Visual Basic for Applications methods was illustrated in subsections 3.3.1–3.3.3 through examples of translating a circle (method *Move*), rotating a rectangle (method *Rotate*) and scaling it (method *ScaleEntity*), as well as mirroring a trapezoid (method *Mirror*) and deleting it (method *Delete*). In many of the macros presented above, the *Update* method – used for regenerating constructed geometric objects – is widely employed.

The remaining AutoCAD commands and the full list of Visual Basic for Applications properties and methods for modifying various figures can be found in the literature sources provided at the end of the textbook and in other references.

3.4. Conclusions

This chapter of the textbook has presented information on constructing and modifying various basic geometric figures – points, straight-line segments, polygons, circles, ellipses, their arcs, splines, etc. – using Visual Basic for Applications macros in the AutoCAD system. Such operations (methods) as

translation, rotation, scaling, and their compositions have been analyzed. Relevant examples were considered both directly in AutoCAD and through Visual Basic for Applications macros. The necessary generalisations concerning this programming environment have been made. The correctness of understanding the material can be verified using the control questions listed below.

Control Questions

1. Provide general information on the Visual Basic for Applications software tools in the AutoCAD.
2. Analyze the construction of points using Visual Basic for Applications.
3. Examine the structure of a Visual Basic for Applications program and the purpose of its components for creating line segments.
4. Explain the construction of line segments using the AutoCAD system menu and the Visual Basic for Applications environment.
5. Describe the shape generation of polygons using AutoCAD and Visual Basic for Applications.
6. Analyze the execution of hatching using the AutoCAD menu and Visual Basic for Applications macros.
7. Demonstrate the flexibility and efficiency of computer geometric modeling based on specialised software.
8. Discuss the modeling of circles using AutoCAD commands and Visual Basic for Applications tools.
9. Compare the advantages and disadvantages of creating ellipses and their arcs directly in AutoCAD and by using Visual Basic for Applications.
10. Describe spline modeling in AutoCAD using interpolation points and control vertices.
11. Justify the importance of effective geometric modifications for the automated design of various technical objects and processes.
12. Characterize the use of Visual Basic for Applications properties and methods for modifying geometric objects.
13. Illustrate, with examples, the compositions of transformation operations such as translation, rotation, and scaling.
14. Explain the purpose and main functions of the AutoCAD Properties menu for modifying geometric figures.

15. Describe such methods of geometric objects in Visual Basic for Applications as translation, rotation, scaling, and symmetry.

16. Using specific program examples, explain the application of the methods for copying, deleting, and regenerating geometric figures using Visual Basic for Applications.

17. Characterize the most promising directions for the further development of computer graphics as a component of automated design of various technical objects and processes.

GENERAL CONCLUSIONS

Computer-aided engineering graphics is a modern technical discipline that is continuously and rapidly developing and is characterized by a substantial volume of material and considerable conceptual depth. Mastery of this course is an essential component of the professional training of many technical specialists, ensuring their successful future work with automated information systems.

A distinctive feature of this textbook is the extensive use of the systems approach in presenting the material. This applies to the integration of such areas as: the technical and software tools of computer graphics; the mathematical foundations of geometric object modeling; direct shape generation in the environment of a specific system (AutoCAD); and the automation of constructions using auxiliary programming tools, illustrated through Visual Basic for Applications.

Thus, in particular, the *first chapter* provides general information on computer-aided engineering graphics (technical tools used, types of images, color reproduction models, etc.).

The *second chapter* is devoted to computer-based shape generation in the AutoCAD system. It examines tools for two-dimensional and three-dimensional geometric modeling, including the use of coordinate systems, layer and viewport management, and commands for constructing such primitives as points, lines, circles, polygons, splines, parallelepipeds, cylinders, cones, pyramids, etc.; their editing; and the definition of parametric dependencies. Tools for drawing documentation and for creating solid models of parts are also presented.

The *third chapter* provides information on automating constructions in the AutoCAD system using Visual Basic for Applications macros for various basic geometric figures (points, segments, polygons, circles, ellipses, their arcs, splines, etc.) and their modification.

The described approach allows the proposed instructional material to be mastered comprehensively, offering a deep understanding of its essence and enabling students to acquire stronger theoretical knowledge and practical skills for effective future application. These factors ensure that students can productively master other computer-aided design systems in addition to AutoCAD.

LIST OF REFERENCES

1. AUTODESK: available at URL: [Getting started with AutoCAD | Autodesk](#) (Accessed: 07 November 2025).
2. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About the Quick Access Toolbar | Autodesk](#) (Accessed: 07 November 2025).
3. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About the Ribbon | Autodesk](#) (Accessed: 07 November 2025).
4. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About the Start tab | Autodesk](#) (Accessed: 07 November 2025).
5. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About the Status Bar | Autodesk](#) (Accessed: 07 November 2025).
6. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Using Task-Based Workspaces | Autodesk](#) (Accessed: 07 November 2025).
7. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Drawings and Templates | Autodesk](#) (Accessed: 07 November 2025).
8. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Units of Measurement | Autodesk](#) (Accessed: 07 November 2025).
9. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Panning and Zooming in the Current View | Autodesk](#) (Accessed: 07 November 2025).
10. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About the Navigation Bar | Autodesk](#) (Accessed: 07 November 2025).
11. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About the ViewCube | Autodesk](#) (Accessed: 07 November 2025).
12. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Viewing 3D Objects | Autodesk](#) (Accessed: 07 November 2025).
13. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About 3D Navigation Tools | Autodesk](#) (Accessed: 07 November 2025).
14. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Parallel and Perspective Views | Autodesk](#) (Accessed: 07 November 2025).
15. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About the User Coordinate System \(UCS\) | Autodesk](#) (Accessed: 07 November 2025).

16. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Coordinate Entry | Autodesk](#) (Accessed: 07 November 2025).
17. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Tracking Points Automatically Using Object Snaps | Autodesk](#) (Accessed: 07 November 2025).
18. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Adjusting the Grid and Grid Snap | Autodesk](#) (Accessed: 07 November 2025).
19. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Orthogonal Locking \(Ortho Mode\) | Autodesk](#) (Accessed: 07 November 2025).
20. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Polar Tracking and PolarSnap | Autodesk](#) (Accessed: 07 November 2025).
21. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Specifying Distances, Lengths, and Angles | Autodesk](#) (Accessed: 07 November 2025).
22. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Using Object Snaps | Autodesk](#) (Accessed: 07 November 2025).
23. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Using Object Snaps in 3D | Autodesk](#) (Accessed: 07 November 2025).
24. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Finding Distances, Angles, and Point Locations | Autodesk](#) (Accessed: 07 November 2025).
25. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Finding Area and Mass Properties Information | Autodesk](#) (Accessed: 07 November 2025).
26. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Specifying Equal Intervals on Objects | Autodesk](#) (Accessed: 07 November 2025).
27. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About 2D Isometric Drawing | Autodesk](#) (Accessed: 07 November 2025).
28. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Lines | Autodesk](#) (Accessed: 07 November 2025).
29. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Polylines | Autodesk](#) (Accessed: 07 November 2025).

30. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Curved Objects | Autodesk](#) (Accessed: 07 November 2025).
31. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Arcs | Autodesk](#) (Accessed: 07 November 2025).
32. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Circles | Autodesk](#) (Accessed: 07 November 2025).
33. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Donuts | Autodesk](#) (Accessed: 07 November 2025).
34. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Ellipses | Autodesk](#) (Accessed: 07 November 2025).
35. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Splines | Autodesk](#) (Accessed: 07 November 2025).
36. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Helixes | Autodesk](#) (Accessed: 07 November 2025).
37. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Regions | Autodesk](#) (Accessed: 07 November 2025).
38. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Parametric Drawing and Constraints | Autodesk](#) (Accessed: 07 November 2025).
39. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Geometric Constraints | Autodesk](#) (Accessed: 07 November 2025).
40. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Applying and Deleting Geometric Constraints | Autodesk](#) (Accessed: 07 November 2025).
41. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Displaying and Verifying Geometric Constraints | Autodesk](#) (Accessed: 07 November 2025).
42. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Modifying Geometrically Constrained Objects | Autodesk](#) (Accessed: 07 November 2025).
43. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Inferring Geometric Constraints | Autodesk](#) (Accessed: 07 November 2025).
44. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Dimensional Constraints | Autodesk](#) (Accessed: 07 November 2025).

45. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Applying Dimensional Constraints | Autodesk](#) (Accessed: 07 November 2025).
46. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Controlling the Display of Dimensional Constraints | Autodesk](#) (Accessed: 07 November 2025).
47. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Modifying Objects With Dimensional Constraints Applied | Autodesk](#) (Accessed: 07 November 2025).
48. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Controlling Geometry With the Parameters Manager | Autodesk](#) (Accessed: 07 November 2025).
49. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Parametric Formulas and Equations | Autodesk](#) (Accessed: 07 November 2025).
50. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Selecting Objects Individually | Autodesk](#) (Accessed: 07 November 2025).
51. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Selecting Multiple Objects | Autodesk](#) (Accessed: 07 November 2025).
52. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Moving Objects | Autodesk](#) (Accessed: 07 November 2025).
53. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Rotating Objects | Autodesk](#) (Accessed: 07 November 2025).
54. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Resizing or Reshaping Objects | Autodesk](#) (Accessed: 07 November 2025).
55. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Trimming and Extending Objects | Autodesk](#) (Accessed: 07 November 2025).
56. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Copying Objects | Autodesk](#) (Accessed: 07 November 2025).
57. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Using the Clipboard | Autodesk](#) (Accessed: 07 November 2025).
58. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Mirroring Objects | Autodesk](#) (Accessed: 07 November 2025).
59. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Offsetting Objects | Autodesk](#) (Accessed: 07 November 2025).
60. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help |](#)

[About Arrays | Autodesk](#) (Accessed: 07 November 2025).

61. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Fillets and Rounds | Autodesk](#) (Accessed: 07 November 2025).

62. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Chamfers and Bevels | Autodesk](#) (Accessed: 07 November 2025).

63. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Breaking and Joining Objects | Autodesk](#) (Accessed: 07 November 2025).

64. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Editing with Grips | Autodesk](#) (Accessed: 07 November 2025).

65. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Making Multiple Copies With Grips | Autodesk](#) (Accessed: 07 November 2025).

66. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Modifying Splines | Autodesk](#) (Accessed: 07 November 2025).

67. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Exploding Compound Objects | Autodesk](#) (Accessed: 07 November 2025).

68. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Object Properties | Autodesk](#) (Accessed: 07 November 2025).

69. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Object Properties Tools | Autodesk](#) (Accessed: 07 November 2025).

70. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Layers | Autodesk](#) (Accessed: 07 November 2025).

71. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Setting the Color of Objects | Autodesk](#) (Accessed: 07 November 2025).

72. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Linetypes | Autodesk](#) (Accessed: 07 November 2025).

73. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Lineweights | Autodesk](#) (Accessed: 07 November 2025).

74. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Annotation Objects | Autodesk](#) (Accessed: 07 November 2025).

75. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Creating Notes Using Text | Autodesk](#) (Accessed: 07 November 2025).

76. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Multiline Text Format | Autodesk](#) (Accessed: 07 November 2025).

77. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help |](#)

[About Text Styles | Autodesk](#) (Accessed: 07 November 2025).

78. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Changing Text Scale and Justification | Autodesk](#) (Accessed: 07 November 2025).

79. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Leader Objects | Autodesk](#) (Accessed: 07 November 2025).

80. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Creating Leaders | Autodesk](#) (Accessed: 07 November 2025).

81. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Modifying Leaders With Grips | Autodesk](#) (Accessed: 07 November 2025).

82. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About the Types of Dimensions | Autodesk](#) (Accessed: 07 November 2025).

83. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Associative Dimensions | Autodesk](#) (Accessed: 07 November 2025).

84. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Setting the Scale for Dimensions | Autodesk](#) (Accessed: 07 November 2025).

85. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Dimension Styles | Autodesk](#) (Accessed: 07 November 2025).

86. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Modifying Dimension Geometry | Autodesk](#) (Accessed: 07 November 2025).

87. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Modifying Dimension Text | Autodesk](#) (Accessed: 07 November 2025).

88. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Hatch Patterns and Fills | Autodesk](#) (Accessed: 07 November 2025).

89. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Modeling 3D Objects | Autodesk](#) (Accessed: 07 November 2025).

90. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Modifying the Properties of 3D Objects | Autodesk](#) (Accessed: 07 November 2025).

91. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Creating Basic 3D Solids and Walls | Autodesk](#) (Accessed: 07 November 2025).

92. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | To Work With Creating Solid Boxes | Autodesk](#) (Accessed: 07 November 2025).

93. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | To Work With Creating Solid Cones | Autodesk](#) (Accessed: 07 November 2025).

94. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | To Work With Creating Solid Cylinders | Autodesk](#) (Accessed: 07 November 2025).
95. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | To Work With Creating Solid Spheres | Autodesk](#) (Accessed: 07 November 2025).
96. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | To Work With Creating Solid Pyramids | Autodesk](#) (Accessed: 07 November 2025).
97. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | To Create a Solid Torus | Autodesk](#) (Accessed: 07 November 2025).
98. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | To Work With Creating Solid Wedges | Autodesk](#) (Accessed: 07 November 2025).
99. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Converting Surfaces and Meshes to 3D Solids | Autodesk](#) (Accessed: 07 November 2025).
100. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Creating 3D Surfaces | Autodesk](#) (Accessed: 07 November 2025).
101. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Offsetting Surfaces | Autodesk](#) (Accessed: 07 November 2025).
102. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Creating NURBS Surfaces | Autodesk](#) (Accessed: 07 November 2025).
103. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Constructing Solids and Surfaces From 2D Geometry | Autodesk](#) (Accessed: 07 November 2025).
104. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Constructing Meshes From Other Objects | Autodesk](#) (Accessed: 07 November 2025).
105. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Using 3D Gizmos | Autodesk](#) (Accessed: 07 November 2025).
106. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Moving, Rotating, and Scaling 3D Subobjects | Autodesk](#) (Accessed: 07 November 2025).
107. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Selecting Subobjects | Autodesk](#) (Accessed: 07 November 2025).
108. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Modifying Faces on 3D Solids | Autodesk](#) (Accessed: 07 November 2025).

109. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Modifying Edges on 3D Objects | Autodesk](#) (Accessed: 07 November 2025).
110. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Modifying Vertices on 3D Objects | Autodesk](#) (Accessed: 07 November 2025).
111. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Shelling 3D Solids | Autodesk](#) (Accessed: 07 November 2025).
112. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Pressing or Pulling Areas, Objects, and Faces | Autodesk](#) (Accessed: 07 November 2025).
113. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Modifying Surfaces | Autodesk](#) (Accessed: 07 November 2025).
114. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Editing NURBS Surfaces | Autodesk](#) (Accessed: 07 November 2025).
115. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Section Objects | Autodesk](#) (Accessed: 07 November 2025).
116. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Rendering | Autodesk](#) (Accessed: 07 November 2025).
117. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Materials | Autodesk](#) (Accessed: 07 November 2025).
118. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Creating and Modifying Materials | Autodesk](#) (Accessed: 07 November 2025).
119. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Applying Materials to Objects and Faces | Autodesk](#) (Accessed: 07 November 2025).
120. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Textures | Autodesk](#) (Accessed: 07 November 2025).
121. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Lighting | Autodesk](#) (Accessed: 07 November 2025).
122. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Creating Preview Animations | Autodesk](#) (Accessed: 07 November 2025).
123. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Camera Motion Path Animations | Autodesk](#) (Accessed: 07 November 2025).
124. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Model Space and Paper Space | Autodesk](#) (Accessed: 07 November 2025).
125. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Printing From Paper Space | Autodesk](#) (Accessed: 07 November 2025).

126. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Printing From Model Space | Autodesk](#) (Accessed: 07 November 2025).
127. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Layouts | Autodesk](#) (Accessed: 07 November 2025).
128. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Layout Viewports | Autodesk](#) (Accessed: 07 November 2025).
129. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Named Page Setups | Autodesk](#) (Accessed: 07 November 2025).
130. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Setting the Lineweight Scale for a Layout | Autodesk](#) (Accessed: 07 November 2025).
131. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | To Modify a Standard Paper Size | Autodesk](#) (Accessed: 07 November 2025).
132. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | To Set the Paper Size for a Layout | Autodesk](#) (Accessed: 07 November 2025).
133. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | To Work With Printer or Plotter Configurations | Autodesk](#) (Accessed: 07 November 2025).
134. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | About Plotting | Autodesk](#) (Accessed: 07 November 2025).
135. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | Autodesk](#) (Accessed: 07 November 2025).
136. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | LINE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
137. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | POINT \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
138. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | POLYGON \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
139. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | CIRCLE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
140. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | ARC \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
141. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | DONUT \(Command\) | Autodesk](#) (Accessed: 07 November 2025).

142. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | ELLIPSE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
143. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | HATCH \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
144. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | HELIX \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
145. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | LAYER \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
146. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | LIST \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
147. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | MEASUREGEOM \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
148. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | PLINE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
149. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | PROPERTIES \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
150. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | RECTANG \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
151. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | SPLINE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
152. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | STYLE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
153. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | TEXT \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
154. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | UCS \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
155. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | CONE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
156. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | CYLINDER \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
157. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | EXTRUDE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
158. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | PYRAMID \(Command\) | Autodesk](#) (Accessed: 07 November 2025).

159. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | SPHERE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
160. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | SWEEP \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
161. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | TORUS \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
162. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | COPY \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
163. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | ERASE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
164. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | EXPLODE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
165. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | EXTEND \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
166. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | FILLET \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
167. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | JOIN \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
168. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | MIRROR \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
169. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | MIRROR3D \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
170. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | MOVE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
171. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | OFFSET \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
172. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | ROTATE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
173. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | ROTATE3D \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
174. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | SCALE \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
175. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | TRIM \(Command\) | Autodesk](#) (Accessed: 07 November 2025).

176. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | UNION \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
177. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | RIBBON \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
178. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | OSNAP \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
179. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | ORTHO \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
180. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | DIMLINEAR \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
181. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | DIMANGULAR \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
182. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | DIMARC \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
183. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | DIMRADIUS \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
184. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | Plot Dialog Box | Autodesk](#) (Accessed: 07 November 2025).
185. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | PRESSPULL \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
186. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | CHAMFER \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
187. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | AREA \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
188. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | ARRAY \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
189. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | BOX \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
190. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Help | BREAK \(Command\) | Autodesk](#) (Accessed: 07 November 2025).
191. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Defining the Components in a Project \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
192. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023](#)

[Developer and ObjectARX Help | About Using the Code Window \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

193. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | To Create a New Macro \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

194. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | To Run a Macro from the Macros Dialog Box \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

195. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | To Run a Macro from the VBA IDE \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

196. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | To Edit a Macro \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

197. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Viewing Project Information \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

198. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | To Access the Code Associated with a Form \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

199. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | To Open the VBA IDE on Demand \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

200. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Exercise: An Introduction to VBA \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

201. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AutoCAD VBA Project Terms Reference \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

202. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AutoCAD VBA Commands Reference \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

203. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Referencing Objects in the Object Hierarchy \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

204. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Accessing a Collection \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
205. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Adding a New Member to a Collection Object \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
206. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Iterating through a Collection Object \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
207. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Deleting a Member of a Collection Object \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
208. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Converting Arrays to Variants \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
209. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Interpreting Variant Arrays \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
210. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Controlling the Drawing Windows \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
211. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Setting and Returning System Variables \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
212. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Updating the Geometry in the Document Window \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
213. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Resetting Active Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
214. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Adjusting Snap and Grid Alignment \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
215. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Using Ortho Mode \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

216. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Calculating a Defined Area \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
217. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Requesting for a String \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
218. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Requesting a Point \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
219. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Requesting a Keyword \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
220. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Controlling User Input \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
221. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Accessing the AutoCAD Command Line \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
222. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Exporting to Other File Formats \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
223. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Determining the Container Object \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
224. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Lines \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
225. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Curved Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
226. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Point Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
227. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Solid-Filled Areas \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

228. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Assigning the Hatch Pattern Type and Name \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
229. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Defining the Hatch Boundaries \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
230. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating a Selection Set \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
231. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Adding Objects to a Selection Set \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
232. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Using Filter Lists to Define Selection Set Rules \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
233. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Specifying Multiple Criteria in a Selection Set Filter List \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
234. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Adding Complexity to Your Filter List Conditions \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
235. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Using Wild-Card Patterns in Selection Set Filter Criteria \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
236. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Displaying Information About a Selection Set \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
237. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Removing Objects from a Selection Set \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
238. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Purging Named Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
239. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Renaming Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

240. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Copying Multiple Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

241. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Offsetting Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

242. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Mirroring Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

243. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Rectangular Arrays \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

244. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Polar Arrays \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

245. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Moving Objects Along a Vector \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

246. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Rotating Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

247. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Erasing or Deleting Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

248. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Scaling Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

249. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Transforming Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

250. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Extending and Trimming Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

251. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Exploding Objects \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

252. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Editing Polylines \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

253. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Editing Splines \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

254. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Editing Hatch Boundaries \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

255. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Editing Hatch Patterns \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

256. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating and Modifying Text Styles \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

257. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Assigning Fonts \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

258. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Setting Text Height \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

259. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Setting Obliquing Angle \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

260. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Formatting Line Text \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

261. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Aligning Line Text \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

262. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Changing Line Text \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

263. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Formatting Multiline Text \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

264. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Sorting Layers and Linetypes \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
265. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating and Naming Layers \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
266. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Making a Layer Active \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
267. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Turning Layers on and off \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
268. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Freezing and Thawing Layers \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
269. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Locking and Unlocking Layers \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
270. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Assigning Color to a Layer \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
271. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Assigning a Linetype to a Layer \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
272. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Deleting Layers \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
273. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Working with Colors \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
274. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Changing an Object's Layer \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
275. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Changing an Object's Color \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

276. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Changing an Object's Linetype \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
277. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About How AutoCAD Saves Layer Settings \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
278. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Saving Layer Settings \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
279. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Restoring Layer Settings \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
280. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About the Parts of a Dimension \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
281. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Defining the Dimension System Variables \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
282. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Setting Dimension Text Styles \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
283. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Leader Lines \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
284. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Linear Dimensions \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
285. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Radial Dimensions \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
286. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Angular Dimensions \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
287. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Overriding Dimension Text \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

288. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating, Modifying, and Copying Dimension Styles \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
289. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Overriding the Dimension Style \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
290. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Dimensioning in Model Space and Paper Space \(VBA/activeX\) | Autodesk](#) (Accessed: 07 November 2025).
291. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Leader Lines \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
292. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Adding the Annotation to a Leader Line \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
293. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Leader Associativity \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
294. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About the MenuBar and MenuGroups Collections \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
295. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Loading Menu Groups \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
296. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Inserting Menus in the Menu Bar \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
297. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Removing Menus from the Menu Bar \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
298. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Specifying 3D Coordinates \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
299. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Defining a User Coordinate System \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

300. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Converting Coordinates \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
301. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Wireframes \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
302. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Meshes \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
303. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Polyface Meshes \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
304. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Creating Solids \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
305. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Rotating in 3D \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
306. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Arraying in 3D \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
307. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Mirroring Objects Along a Plane \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
308. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Editing 3D Solids \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
309. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Model Space and Paper Space \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
310. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Paper Size and Units \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).
311. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Adjusting the Plot Origin \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

312. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Setting the Plot Area \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

313. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Setting the Plot Scale \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

314. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Setting the Plot Device \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

315. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Plotting from Model Space and Paper Space \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

316. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Attaching and Scaling a Raster Image \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

317. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | About Manage Raster Images \(VBA/ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

318. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Reference Guide \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

319. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Object Model \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

320. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | All Drawing Objects \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

321. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Circle Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

322. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddCircle Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

323. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Line Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

324. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddLine Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

325. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Arc Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

326. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddArc Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

327. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Hatch Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

328. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddHatch Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

329. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Point Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

330. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddPoint Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

331. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Spline Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

332. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddSpline Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

333. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Text Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

334. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddText Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

335. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | 3DPolyline Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

336. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Add3DPoly Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

337. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | DimDiametric Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

338. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddDimDiametric Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

339. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | DimAligned Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

340. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddDimAligned Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

341. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | DimAngular Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

342. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddDimAngular Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

343. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Layer Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

344. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | ActiveLayer Property \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

345. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | NurbSurface Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

346. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | UCS Object \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

347. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Activate Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

348. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Add Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

349. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddBox Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

350. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddCone Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

351. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddEllipse Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

352. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddSphere Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

353. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | AddTorus Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

354. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Close Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

355. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Copy Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

356. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Delete Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

357. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Explode Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

358. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Mirror Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

359. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Mirror3D Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

360. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Move Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

361. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Offset Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

362. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Open Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

363. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Quit Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

364. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Rotate Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

365. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Rotate3D Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

366. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | Select Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

367. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | SendCommand Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

368. AUTODESK AutoCAD 2023: available at URL: [AutoCAD 2023 Developer and ObjectARX Help | ScaleEntity Method \(ActiveX\) | Autodesk](#) (Accessed: 07 November 2025).

INDEX OF TERMS

- Arrays of geometric objects 75
- CMY method 14
- Command window 32
- Computer graphics 7
- Computer mice 11
- Context menu 38
- Coordinate grid 196
- Dimension styles 81
- Drawing area 26
- Drawing limits 44
- Graphic displays 11
- Interface 25
- Keyboards 11
- Layer management 47
- Lines 17
- Method:
 - AddArc 180
 - AddCircle 180
 - AddEllipse 183
 - AddLine 164
 - AddPoint 161
 - AddPolyline 169
 - AddSpline 187
 - Delete 197
 - Mirror 197
 - Move 194
 - Rotate 195
 - ScaleEntity 197
 - SendCommand 167, 170, 191
 - TransformBy 200
 - Update 180, 183, 184
- Multifunctional devices 12
- Object properties:
 - Area 176
 - Color 180
 - Length 176
 - Linetype 180
 - Lineweight 180
- Object selection 51
- Parametric constraints 96
- Parametric modeling 18
- Points 53, 160
- Printers 12
- Quick Access Toolbar 30
- Raster graphics 15
- RGB model 14
- Scanners 12
- Solid models 16
- Solids 17
- Surface models 16
- Template files 85
- Text styles 77
- Vector graphics 13
- Visual Basic environment 153
- Visual Basic Help 154
- Wireframe models 16

APPENDIX A. BASIC REQUIREMENTS FOR DRAWING LAYOUT

Sheet Formats and Title Blocks

Technical drawings are produced on standardized sheet formats regulated by DSTU 3321:2003. Each format is defined by the size of its outer rectangular frame (see Table A.1).

Table A.1

Designation of standard formats and their dimensions

<i>Format</i>	<i>Dimensions, mm</i>
<i>A0</i>	<i>1189x841</i>
<i>A1</i>	<i>594x841</i>
<i>A2</i>	<i>594x420</i>
<i>A3</i>	<i>297x420</i>
<i>A4</i>	<i>297x210</i>

All formats originate from the base sheet A0, which has an area of approximately 1 m².

Smaller formats are obtained by dividing the sheet in half parallel to its shorter side (Fig. A.1).

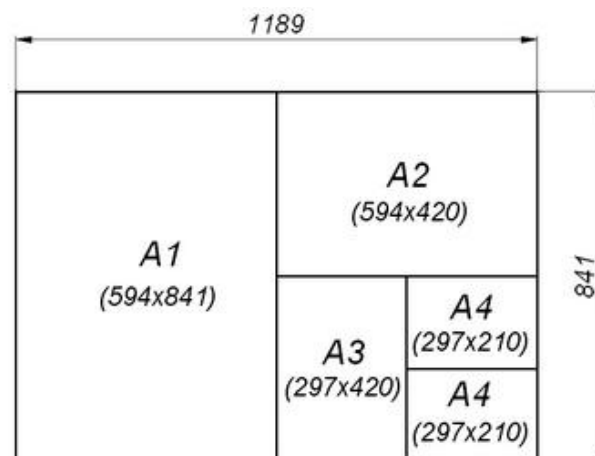


Fig. A.1. Generation of standard sheet formats

Additional, extended formats may be used when required; they are created by proportionally increasing the shorter sides of the standard formats.

Every drawing sheet includes an inner frame, and a title block placed in the lower-right corner (Fig. A.2).

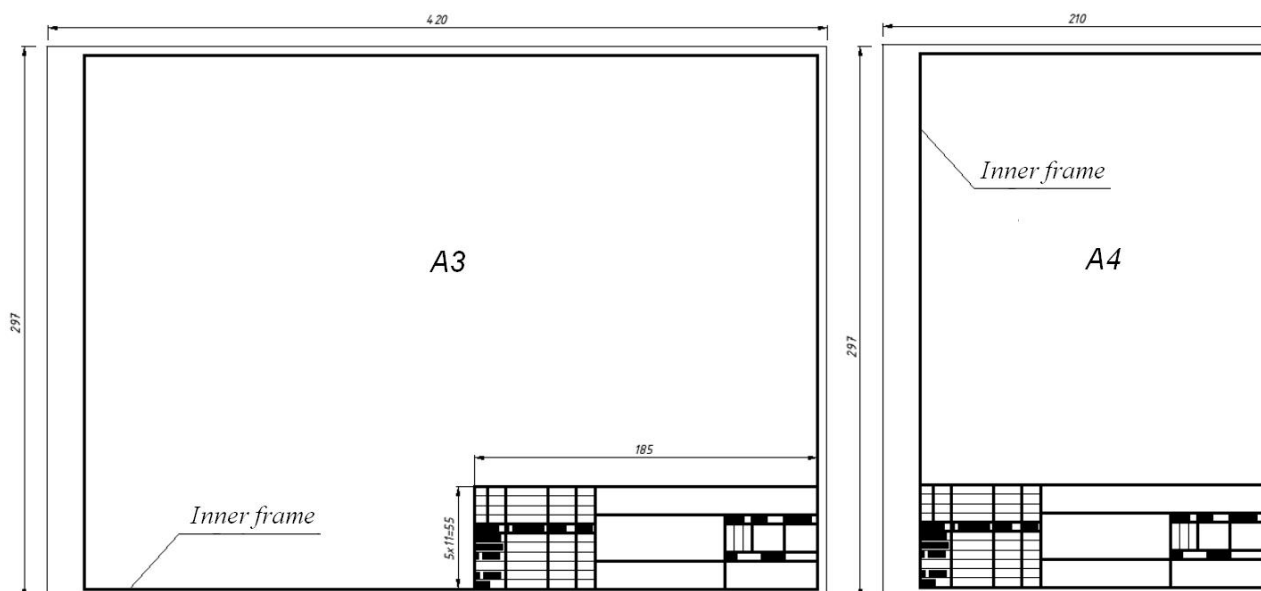


Fig. A.2. Inner frame and location of the title block

For A4 format, the title block is permitted only along the shorter edge of the sheet.

The form, dimensions, and the procedure for filling out the title block (Fig. A.3) are regulated by DSTU GOST 2.104:2006.

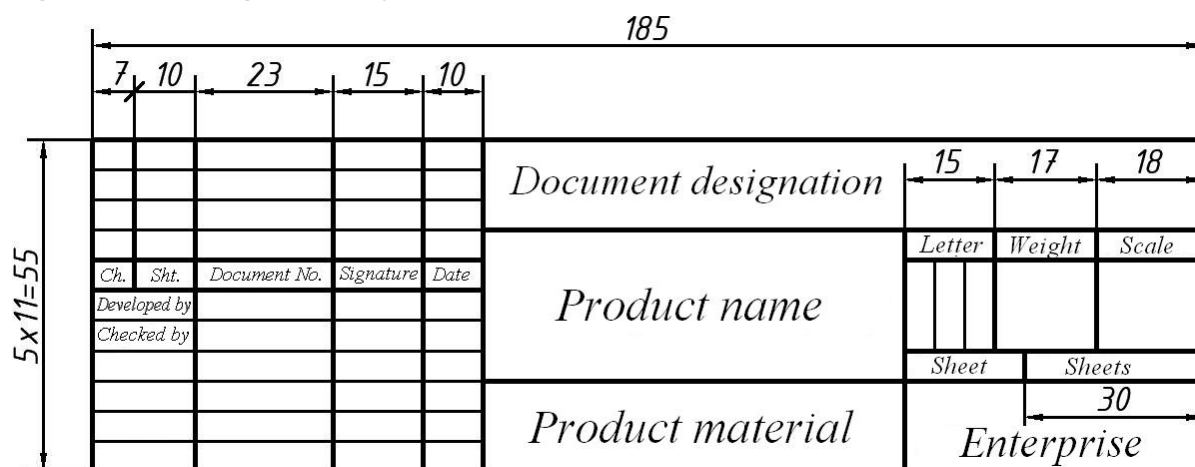


Fig. A.3. Standardized title block (format and structure)

Drawing Scales

A drawing scale represents the ratio between the dimensions depicted in the drawing (without distortion) and the actual dimensions of the object (DSTU 3321:2003).

A 1:1 scale corresponds to full-size representation.

Common reduction scales: 1:2; 1:2,5; 1:4; 1:5; 1:10; 1:15; 1:20; 1:25 ...

Common enlargement scales: 2:1; 2,5:1; 4:1; 5:1; 10:1; 20:1 ...

All dimensions are always written in true (actual) size, regardless of the scale applied (Fig. A.4).

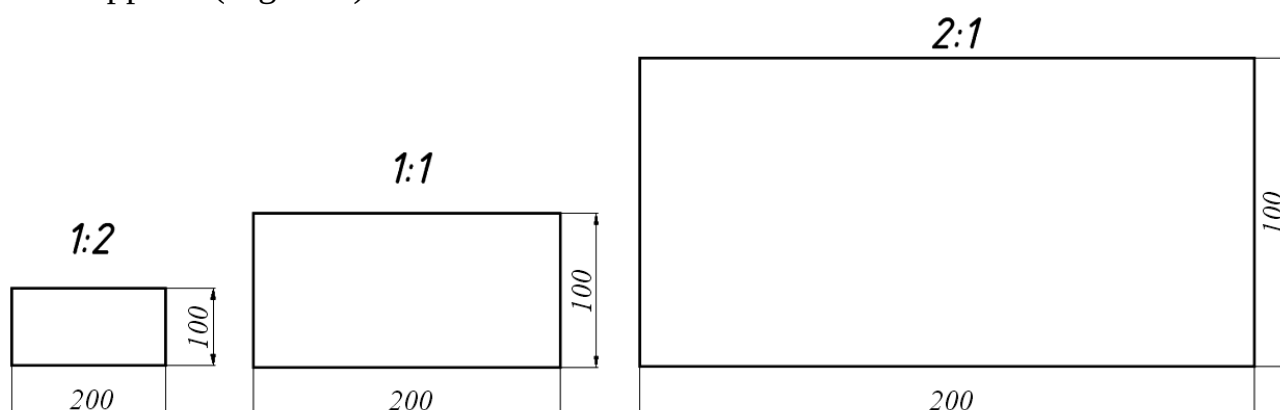


Fig. A.4. True-size dimensions independent of drawing scale

Line Types

Linework on technical drawings follows DSTU ISO 128-20:2003. The standard defines a set of line types, their thicknesses, and applications (Table A.2).

Table A.2

Standard line types defined by DSTU ISO 128-20:2003

<i>Image</i>	<i>Thickness, mm</i>	<i>Line name</i>
	$S=0,5 \dots 1,4$	<i>Solid thick</i>
	$S/3 \dots S/2$	<i>Solid thin</i>
		<i>Solid wavy</i>
		<i>Dashed</i>
		<i>Dash-dot thin</i>
	$S \dots 3S/2$	<i>Open</i>

Figure A.5 provides an example illustrating the use of several line types, including: 1 – continuous thick lines, 2 – continuous thin lines, 5 – thin chain (dash-dot) lines.

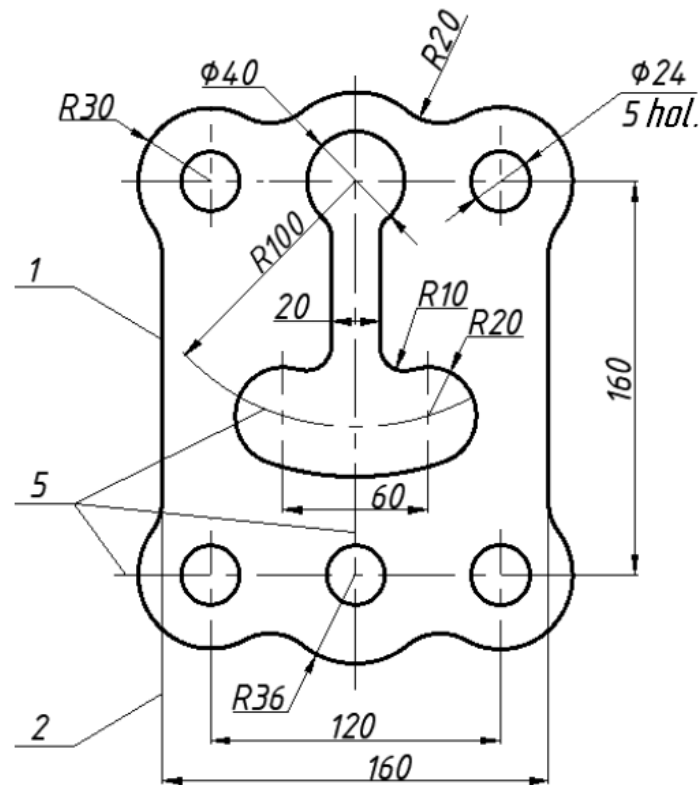


Fig. A.5. Illustration of line type usage in technical drawings

Technical Lettering

All text on engineering drawings is written using a standardized technical font. Acceptable font heights follow the series: 2,5; 3,5; 5; 7; 10; 14; 20, etc. Figure A.6 shows the slanted version of the font traditionally used in mechanical drafting.

ABCDEFGHIJKLMNO
PQRSTUVWXYZ
1234567890
R3 ∅5 □7 ≧1:10 ▷1:5 s4*
abcdefghijklmno
pqrstuvwxyz

Fig. A.6. Standard technical lettering style

Dimensioning Principles

Dimensions in technical drawings (Fig. A.7) are applied using dimension lines, which run parallel to the feature being measured. Each dimension line terminates in arrowheads and meets extension lines that extend from the object's outline or centerlines. The dimension value is placed above the dimension line – preferably outside the object's contour – at a distance of 1 ... 2 mm near the center of the line. Extension lines extend 2 ... 5 mm beyond the arrowheads, while the dimension line should be placed at least 10 mm away from the outline.

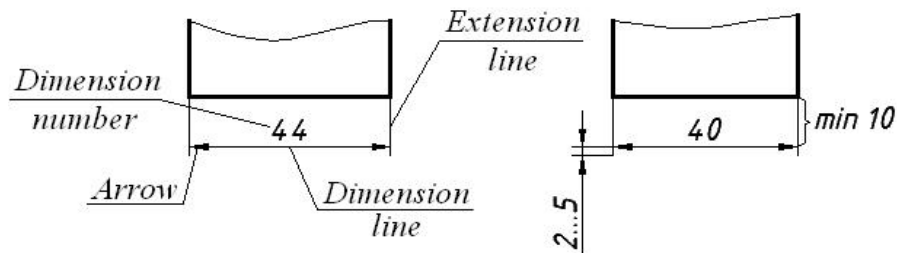


Fig. A.7. Dimensioning a line segment

Angular dimensions (Fig. A.8) use an arc-shaped dimension line, centered at the vertex of the angle, with extension lines drawn radially.

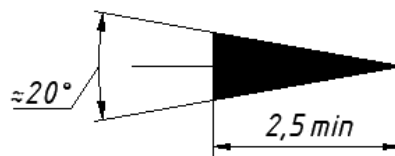


Fig. A.8. Dimension arrow

Such an arrangement ensures that the measured angle is visually clear and unambiguously associated with its geometric elements. Parallel dimension lines must be spaced no less than 8 mm apart (Fig. A.9), with staggered placement of dimension values.

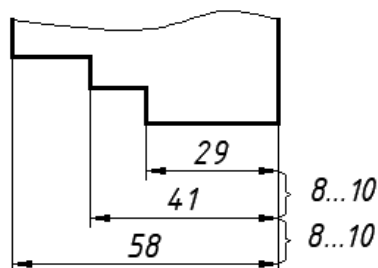


Fig. A.9. Parallel dimension lines

Dimension lines should not intersect whenever possible.

Diameters are indicated with the symbol $\varnothing$, matching the height of the dimension digits (Fig. A.10). Radii are indicated with R if the arc is 180° or less; for larger arcs, the $\varnothing$ symbol is used. Radii originating from the same center must not visually align as extensions of one another.

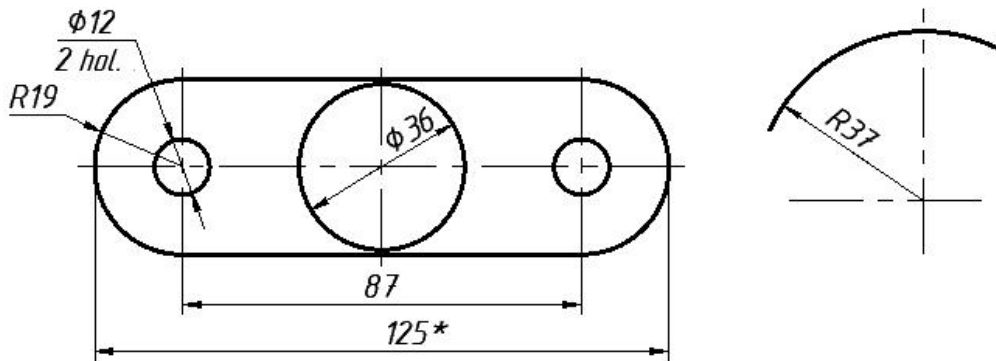


Fig. A.10. Diameter and radius indications

Figure A.11 illustrates the recommended placement of dimension values. In hatched areas, the value is written on a leader line shelf.

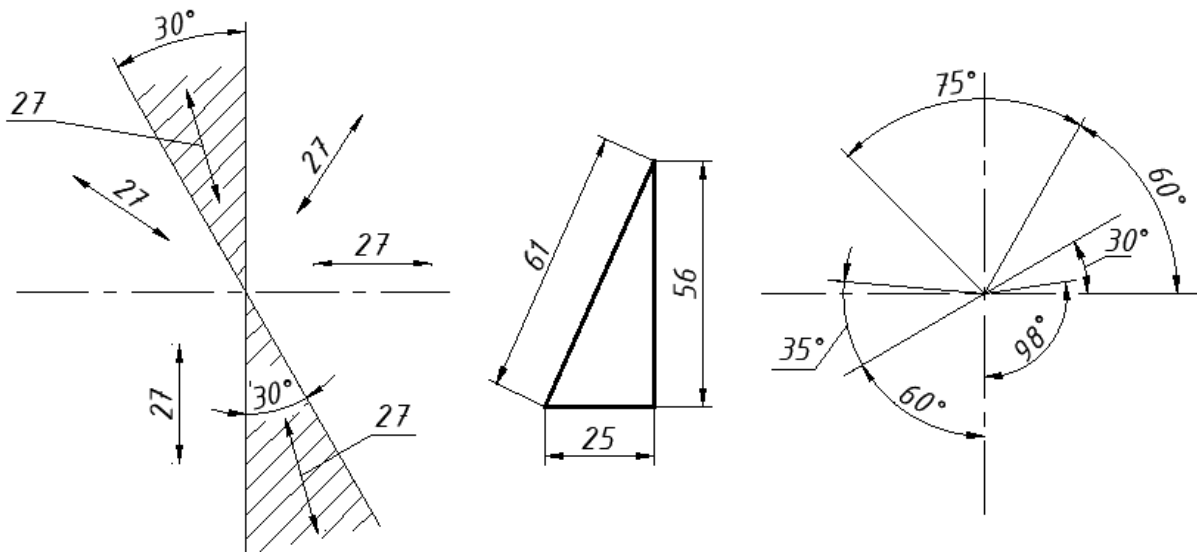


Fig. A.11. Dimensioning scheme

For symmetric parts, dimensions defining symmetrically located features are applied only once, without indicating quantity – for example, repeated radii $R10$ in Fig. A.12. Identical repeated elements are dimensioned once, with their quantity specified on a leader line shelf.

For holes, the number precedes the diameter (e.g., $6 \times \varnothing 10$) or is placed beneath the dimension line (Fig. A.12).

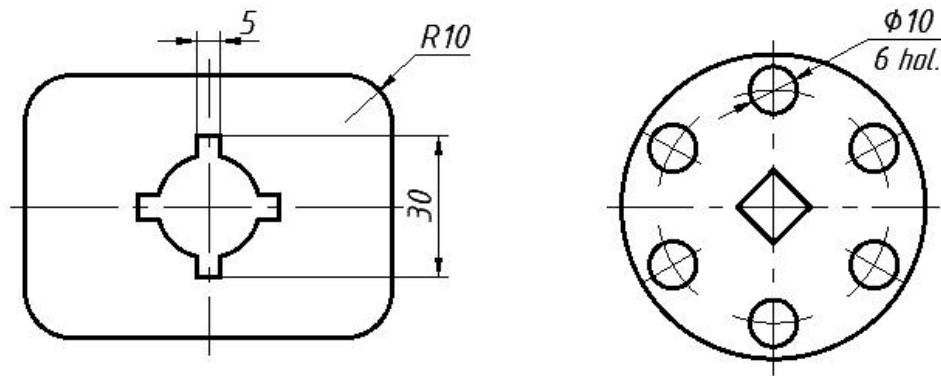


Fig. A.12. Dimensioning identical elements

When holes are evenly spaced on a bolt circle, the angular spacing between centers is omitted. It is acceptable to show only one hole fully, while the others may be represented only by their center marks.

Views

The rules for representing objects in technical drawings are defined by DSTU 3321:2003. Drawings are created using the method of orthographic projection. The six principal views are obtained by projecting the object onto the faces of a cube surrounding it (Fig. A.13).

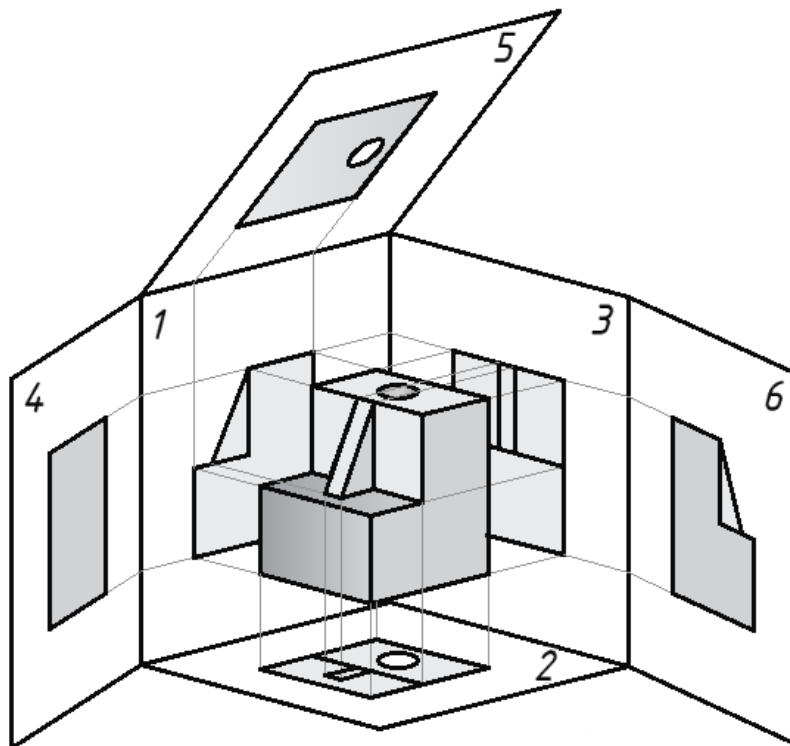


Fig. A.13. Projection onto the faces of a cube

The resulting views are shown in Fig. A.14.

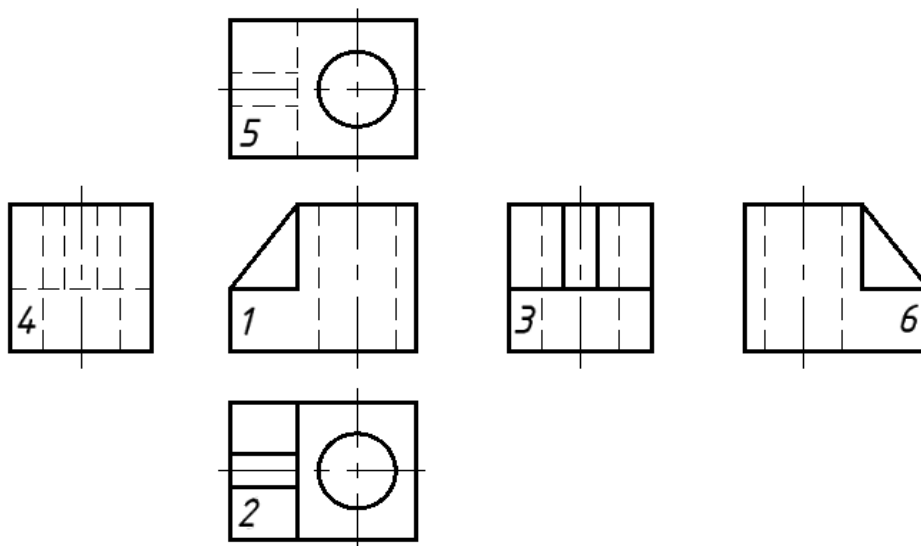


Fig. A.14. Six principal views:
1 – front (principal); 2 – top; 3 – left;
4 – right; 5 – bottom; 6 – rear.

According to the standard, images are classified into *views*, *sections*, and *cross-sections*. A *view* is an image of the visible surface of an object as oriented toward the observer.

The principal view is taken on the rear frontal face of the cube. The object is oriented such that this view provides the most complete representation of its shape and dimensions. If views are not in direct projection alignment, the direction of observation is shown using an arrow placed near the corresponding image, with a Cyrillic capital letter next to the arrow and the same letter above the resulting view (Fig. A.15).

This convention also applies when views are separated by other images or placed on different sheets.

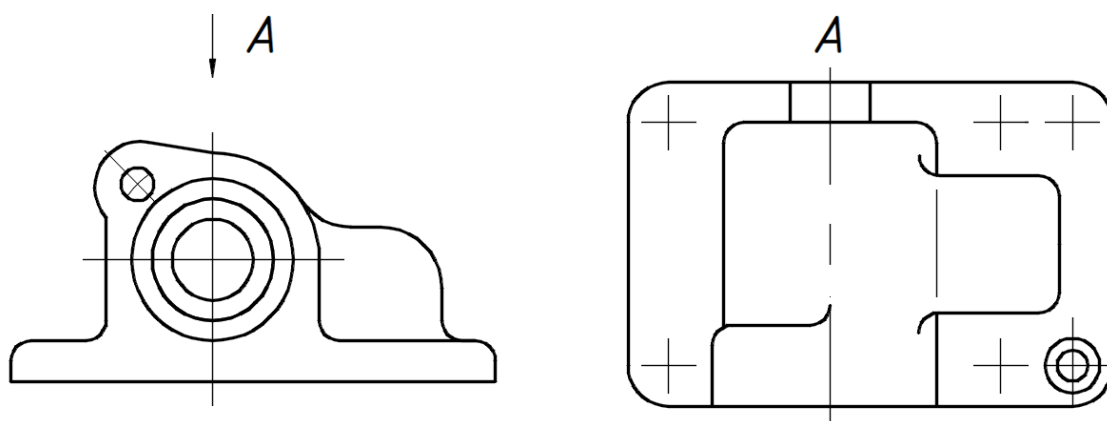


Fig. A.15. Indication of views not in projection alignment

In addition to principal views, auxiliary and local views are used.

An *auxiliary (additional) view* is a view of a surface projected onto a plane not parallel to any of the principal projection planes (Fig. A.16). It is used to obtain an undistorted image of a surface when this is not possible on any principal view.

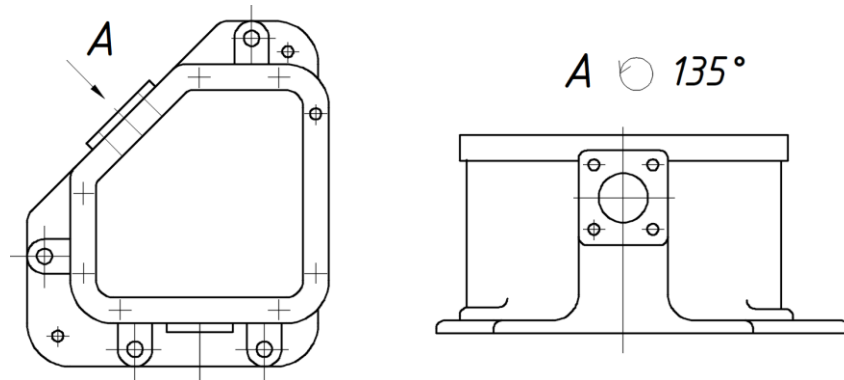


Fig. A.16. Example of an auxiliary view

A *local view* is an image of a limited portion of the object's surface. It may be bounded by a break line (placed at the smallest possible size) or by contour lines of the element (Fig. A.17).

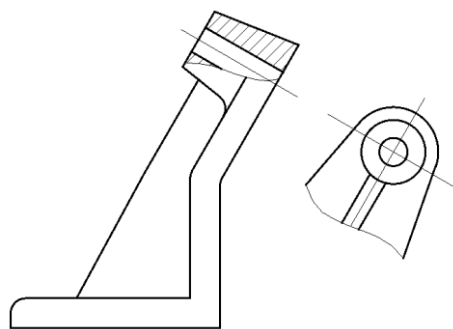


Fig. A.17. Local view

Sections

Sections are used to reveal the internal geometry of parts.

A *section* is an orthographic projection of an object that is imagined to be cut by one or several planes to expose internal features.

Depending on the orientation of the cutting plane, sections are classified as:

- *Horizontal* – the cutting plane is parallel to the horizontal projection plane.
- *Vertical (frontal, profile)* – the cutting plane is perpendicular to the horizontal projection plane.
- *Inclined* – the cutting plane forms an angle with the horizontal projection plane that is different from a right angle.

Based on the number of cutting planes involved, sections are divided into:

- *Simple sections* – produced by a single cutting plane.
- *Complex sections* – produced by two or more cutting planes.

The location of the cutting plane is indicated on the drawing by a cutting-plane line (Fig. A.18). The beginning and end of this line must not intersect the outline of the corresponding view. A pair of identical capital Cyrillic letters is placed next to the arrowheads on the outside of the line. These letters follow alphabetical order and are not repeated within the same drawing. The resulting section view is labeled using the same letters, connected with a dash (e.g., A–A).

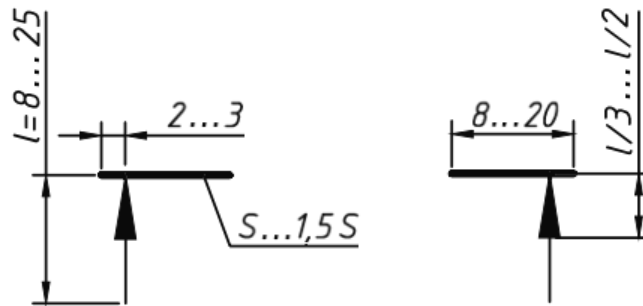


Fig. A.18. Cutting-plane line

A section view includes both the material cut by the plane and the features located behind it (Fig. A.19). The internal outline of the part is shown with a continuous thick line, while the regions that lie within the cutting plane and do not represent cavities are hatched to indicate solid material, ensuring a clear distinction between the cut surfaces and the visible contours located beyond the cutting plane.

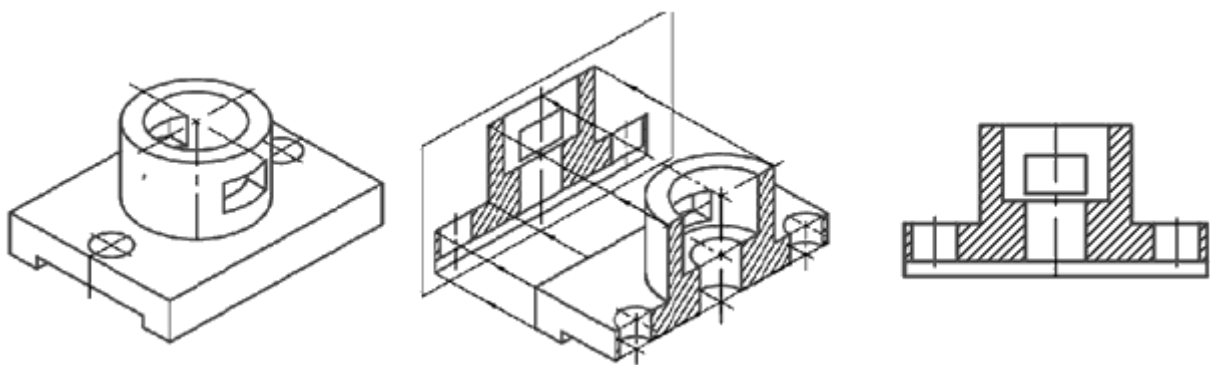


Fig. A.19. Construction of a section

If the cutting plane coincides with a plane of symmetry and the corresponding views are placed on the same sheet in direct projection alignment without being separated by other images, the cutting plane is not shown and the section is not labeled. To reduce the number of separate views on the drawing,

part of a view and the corresponding portion of a section may be combined, with the two areas divided by a continuous freehand (wavy) line. If the figures being joined are symmetrical, the dividing line is the axis of symmetry itself (Fig. A.20). However, when the axis of symmetry coincides with a visible or hidden contour of the part – for example, an edge – the boundary between the view and the section must be drawn as a continuous wavy line.

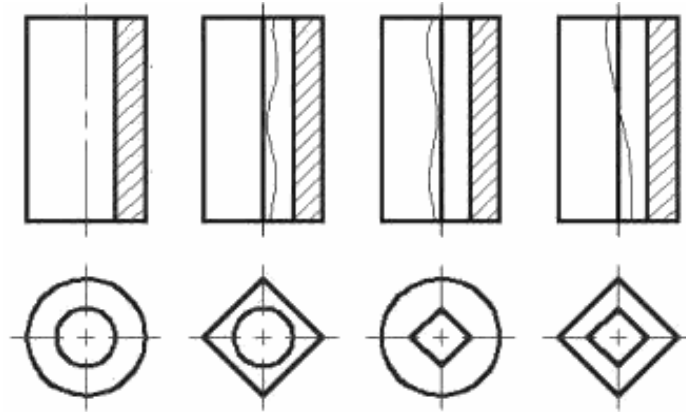


Fig. A.20. Formation of the section view

A section created to reveal the construction of a specific, limited area of a part is referred to as a *local section*. It is distinguished on the view by a continuous freehand (wavy) boundary line (Fig. A.21).

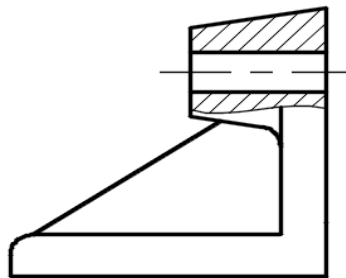


Fig. A.21. Local section

Note that the hatching pattern must remain consistent across all sectioned representations. It is also important to remember that dimensions relating to the external features of the part are applied on the view, whereas those defining internal geometry are placed on the corresponding section.

Dimensions on a Part Drawing

A part drawing contains the following categories of dimensions (Fig. A.22):

- Overall (bounding) dimensions (a, b, c).
- Feature size and location dimensions ($\varnothing, f, l$).

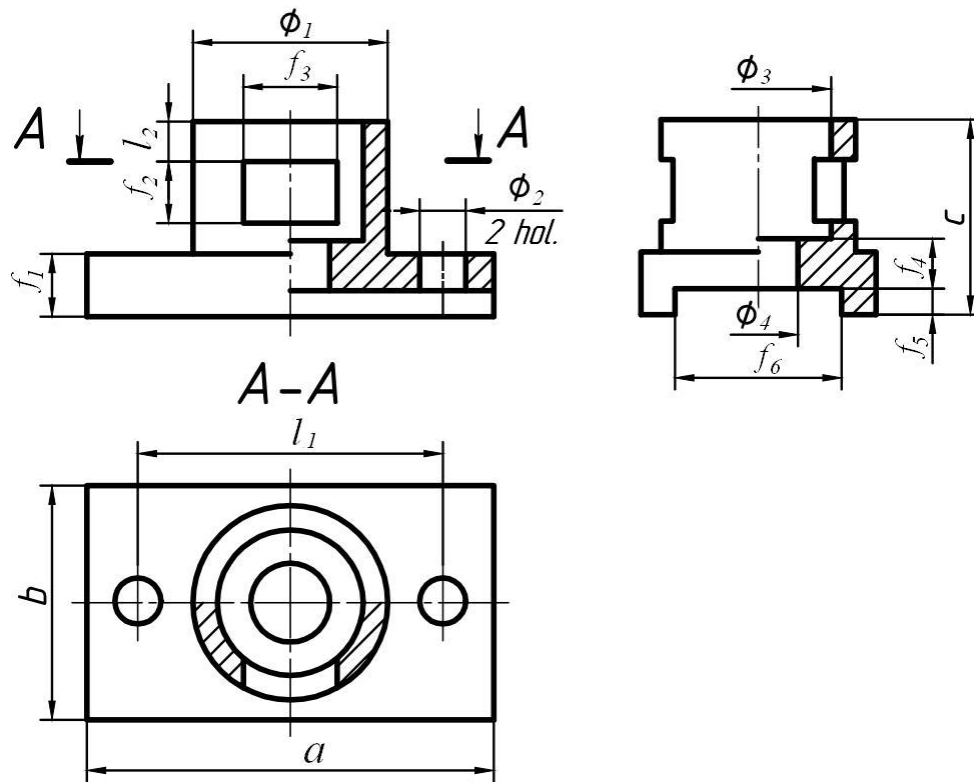


Fig. A.22. Example of dimensioning a part

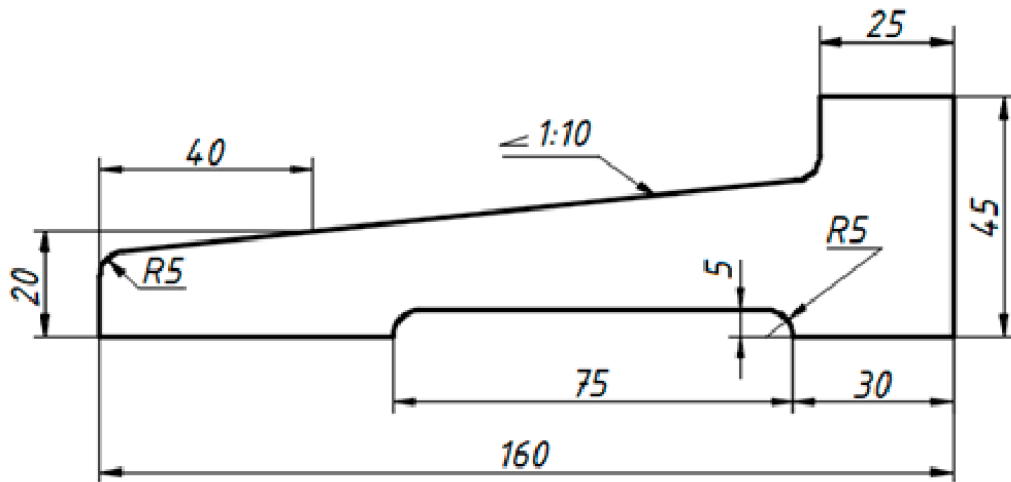
In this example, the dimensions a , b , f_1 define the basic rectangular-parallelepiped envelope of the part, within which a through cutout of size $f_5 \times f_6$ is formed. The diameters $\phi_1 \dots \phi_4$ specify the geometry of the vertical cylindrical features, while f_2 and f_3 define the cross-section of the longitudinal prismatic hole. Its position is determined by the location dimension l_2 . The spacing of the two cylindrical holes ϕ_2 is governed by l_1 , and the height of the hole ϕ_4 is defined by f_4 .

Thus, this appendix provides the fundamental guidelines for preparing technical drawings. It covers sheet formats, title blocks, scales, line types, and lettering styles; explains the principles of dimensioning and the use of views; and introduces the construction of sectional and auxiliary drawings. Together, these conventions ensure the creation of accurate and properly structured drawings within the AutoCAD environment.

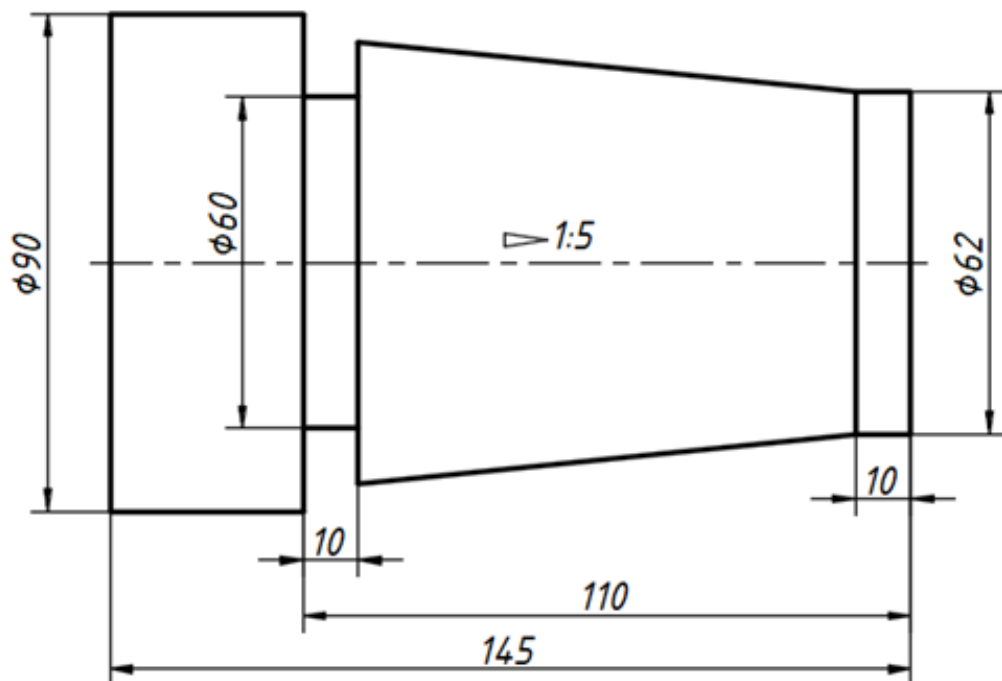
APPENDIX B. INDEPENDENT WORK ASSIGNMENTS

B.1. Part Outline

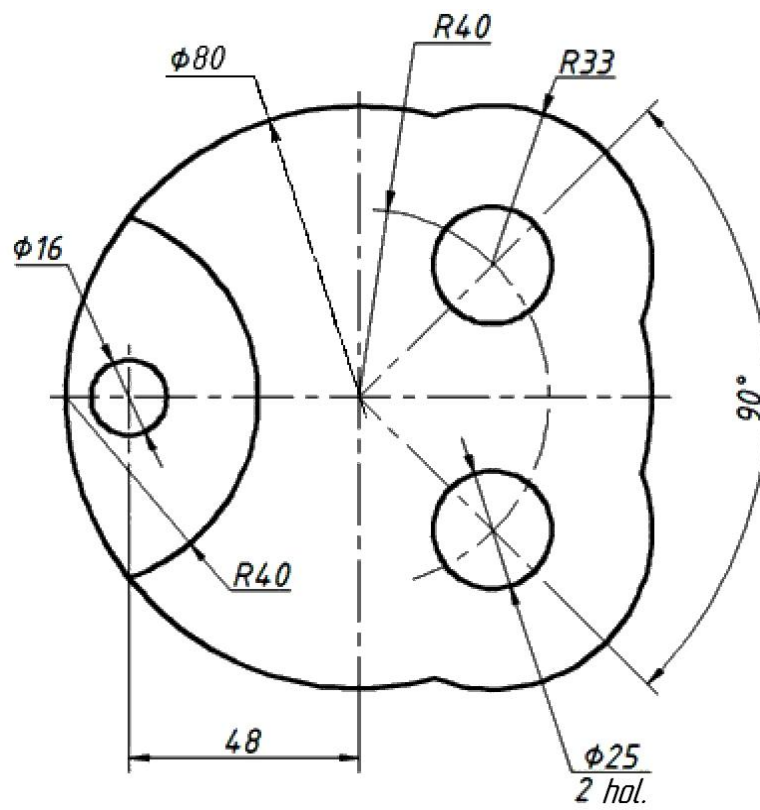
Draw the outline of the part using layers. Apply all necessary dimensions. Complete the drawing according to the drafting standards.



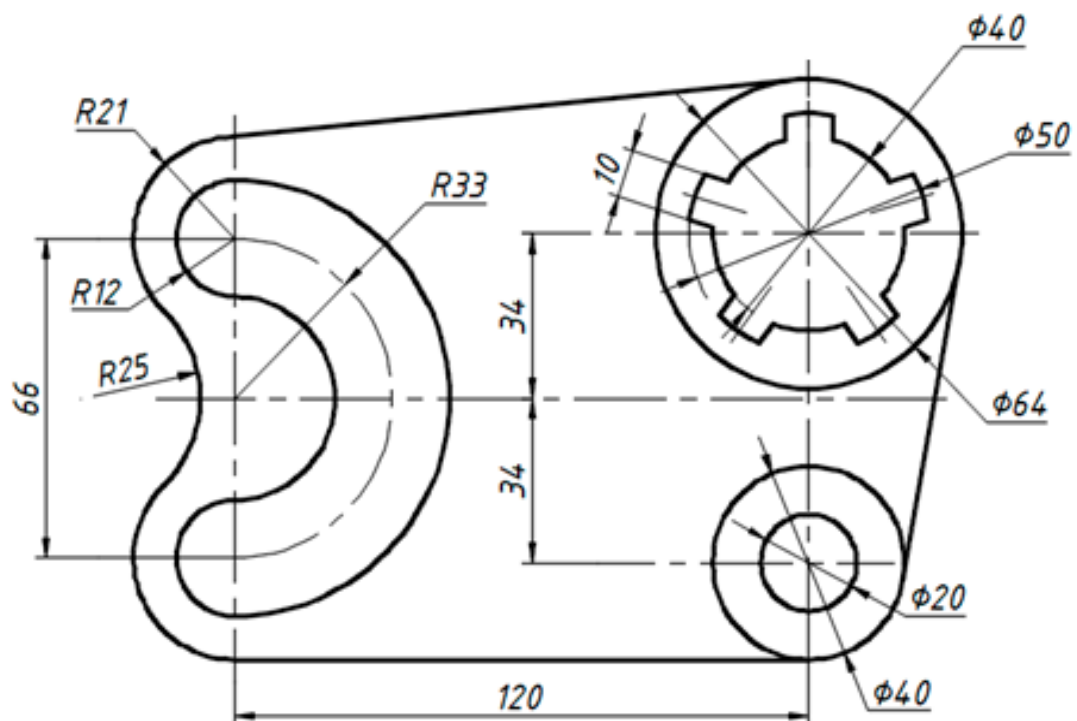
Variant 1



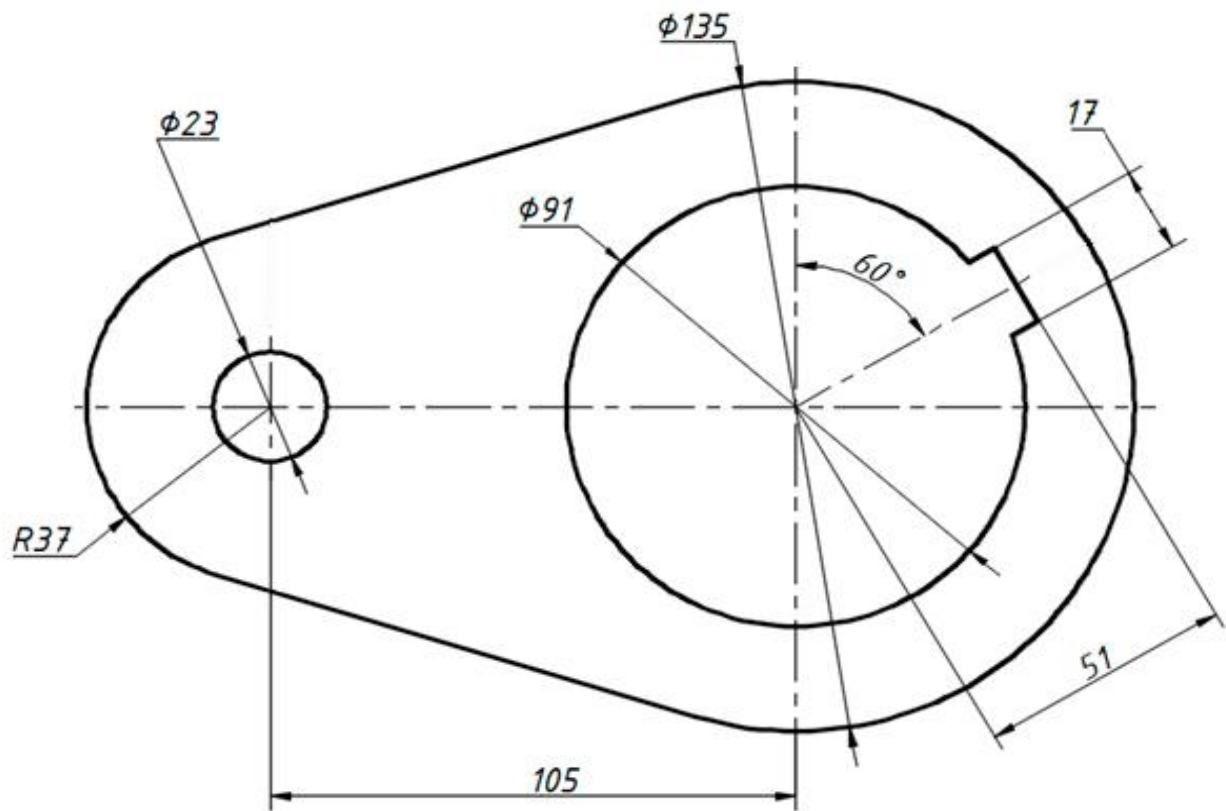
Variant 2



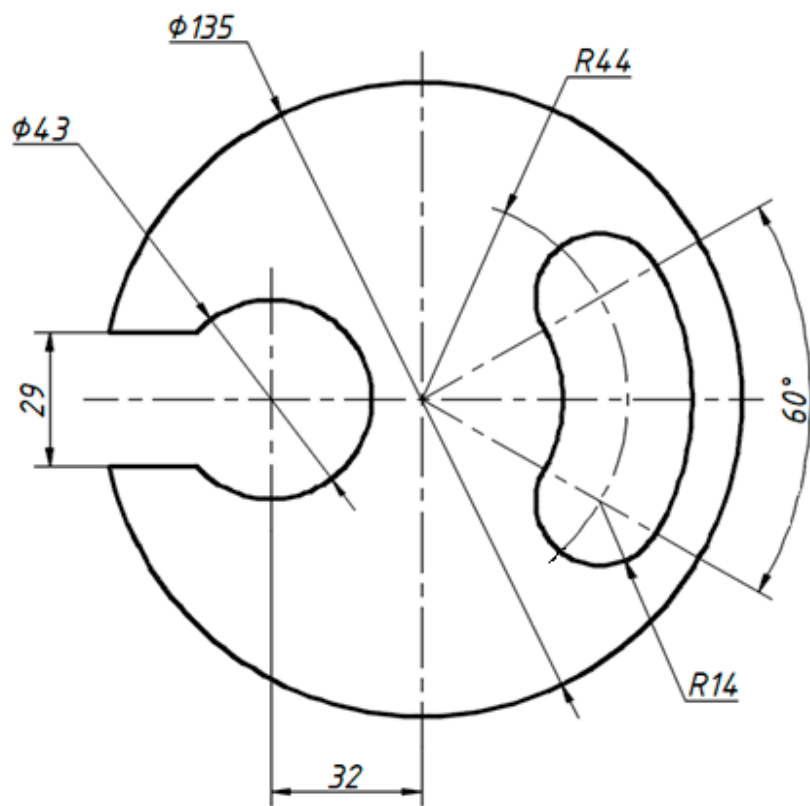
Variant 3



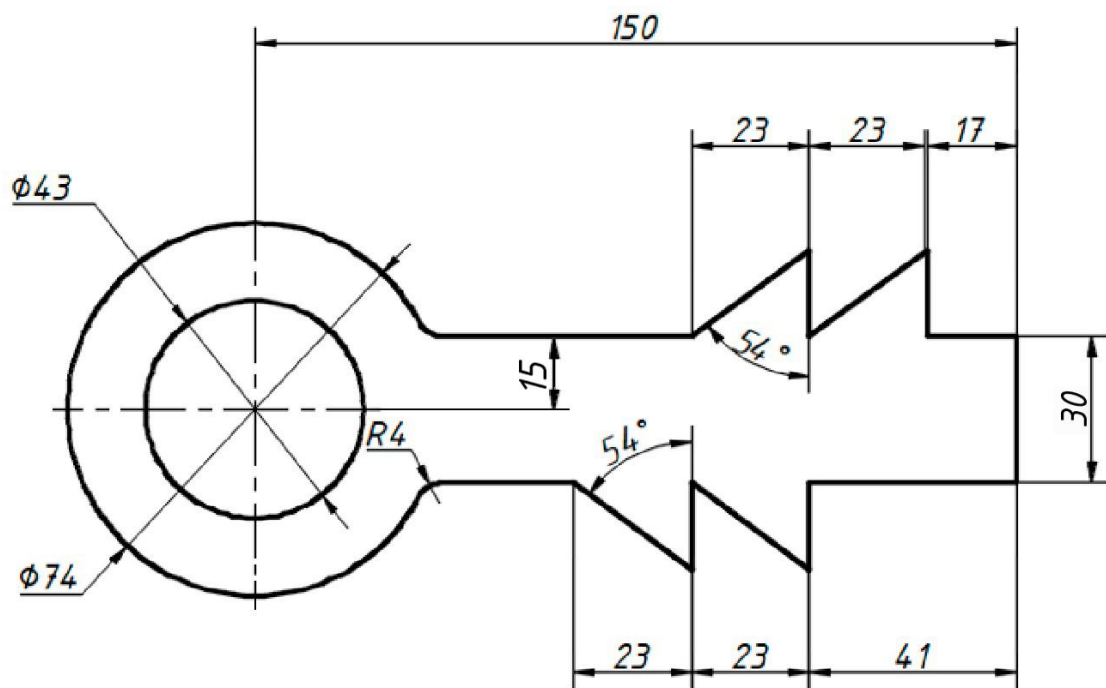
Variant 4



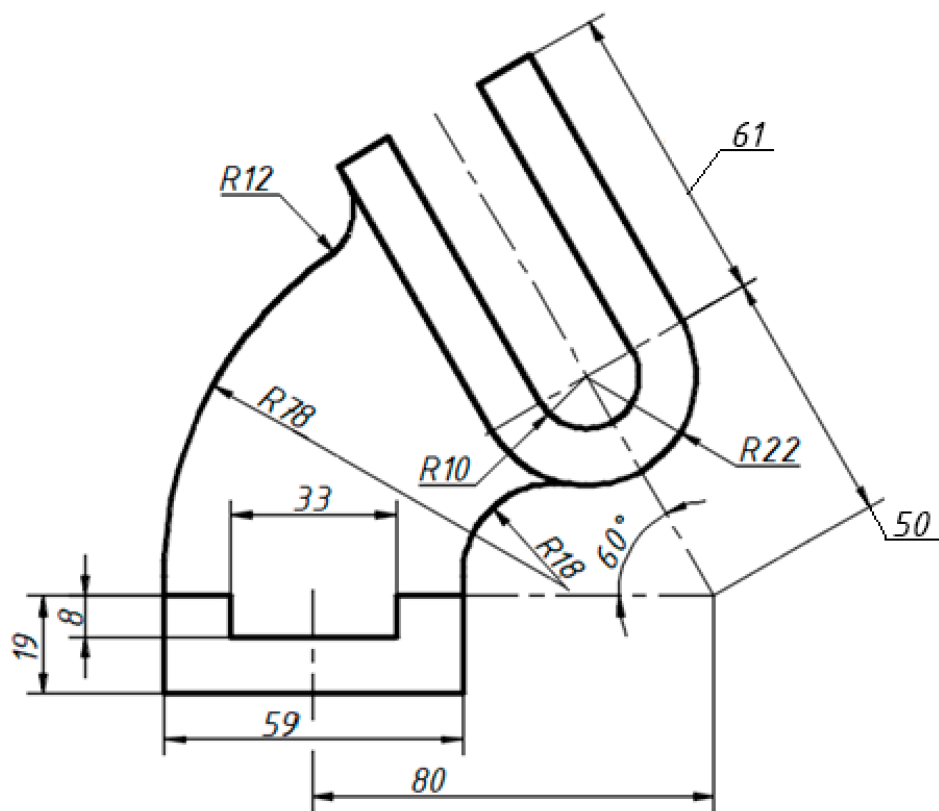
Variant 5



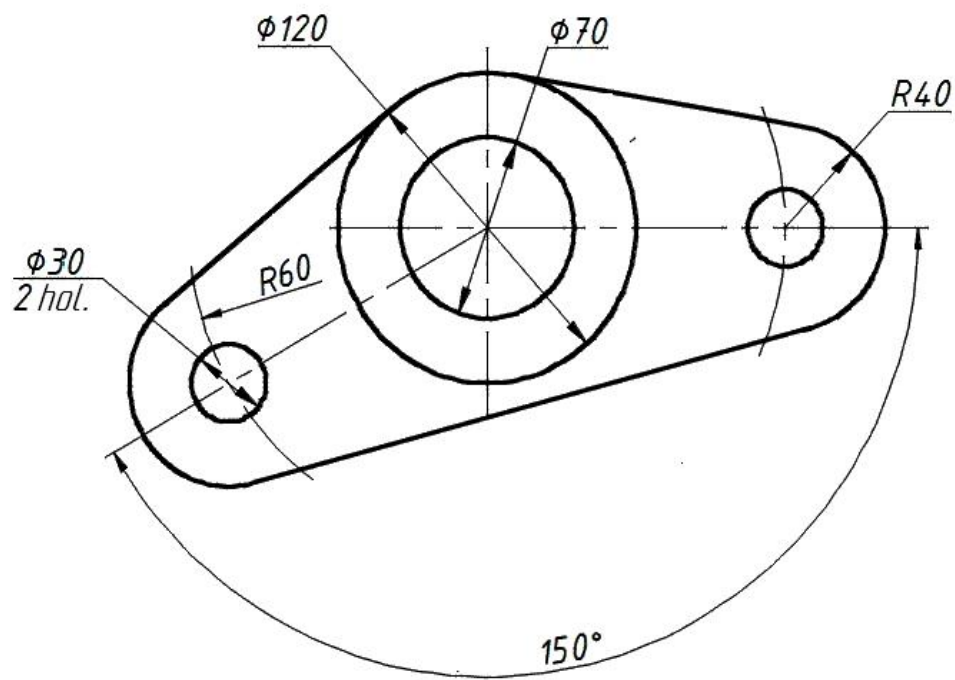
Variant 6



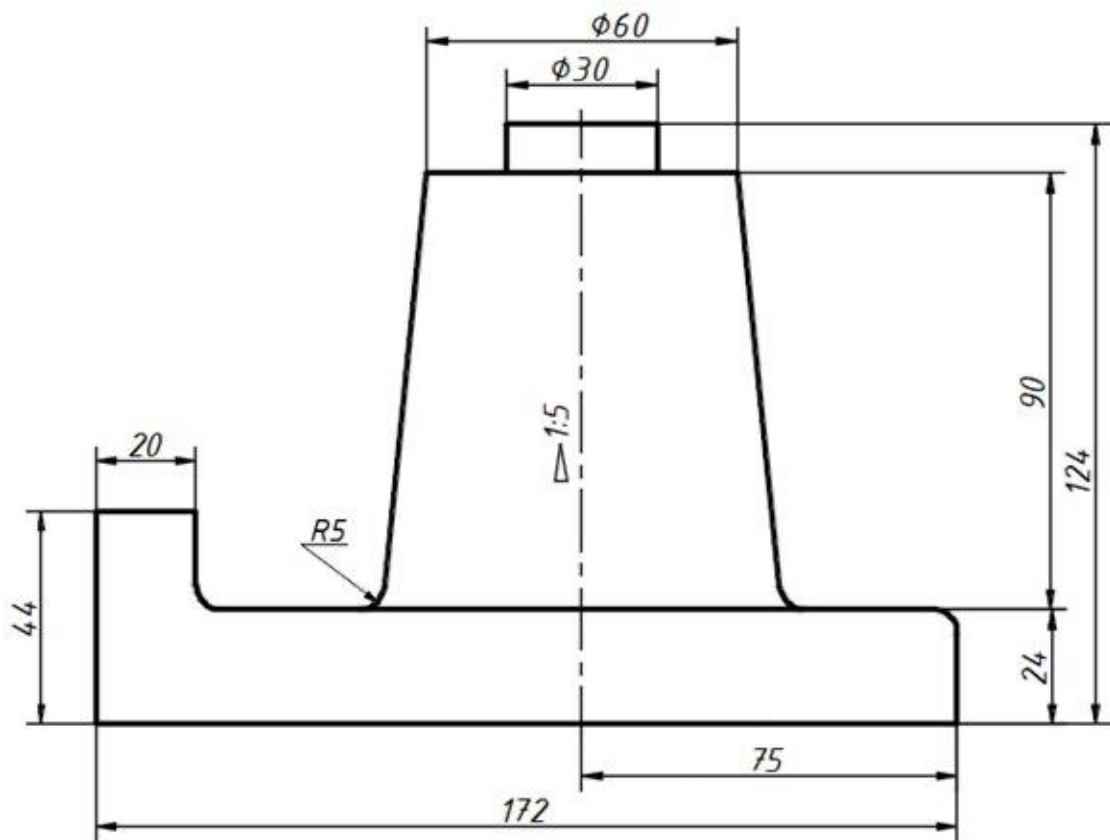
Variant 7



Variant 8

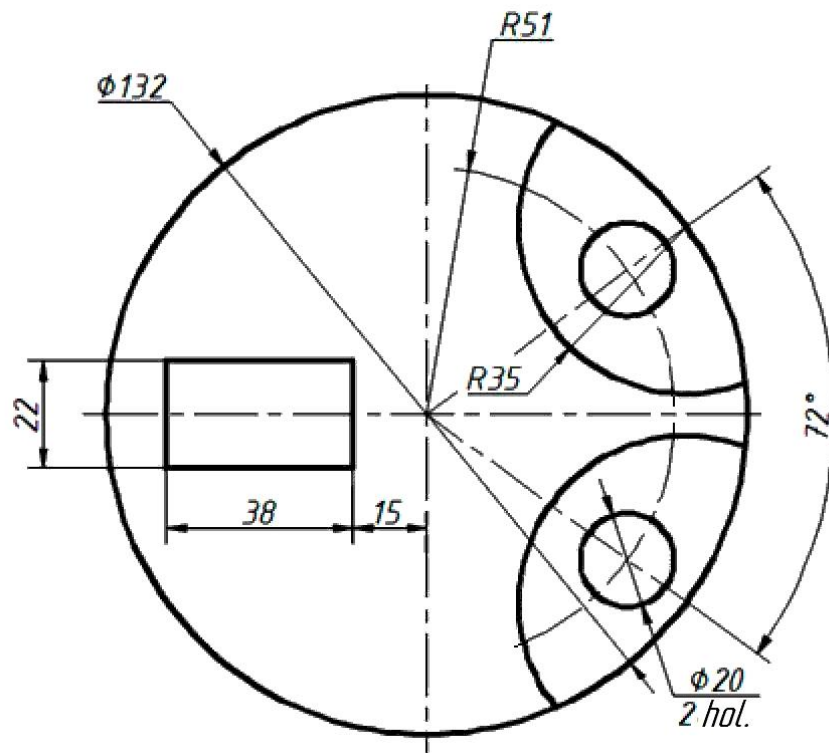


Variant 9

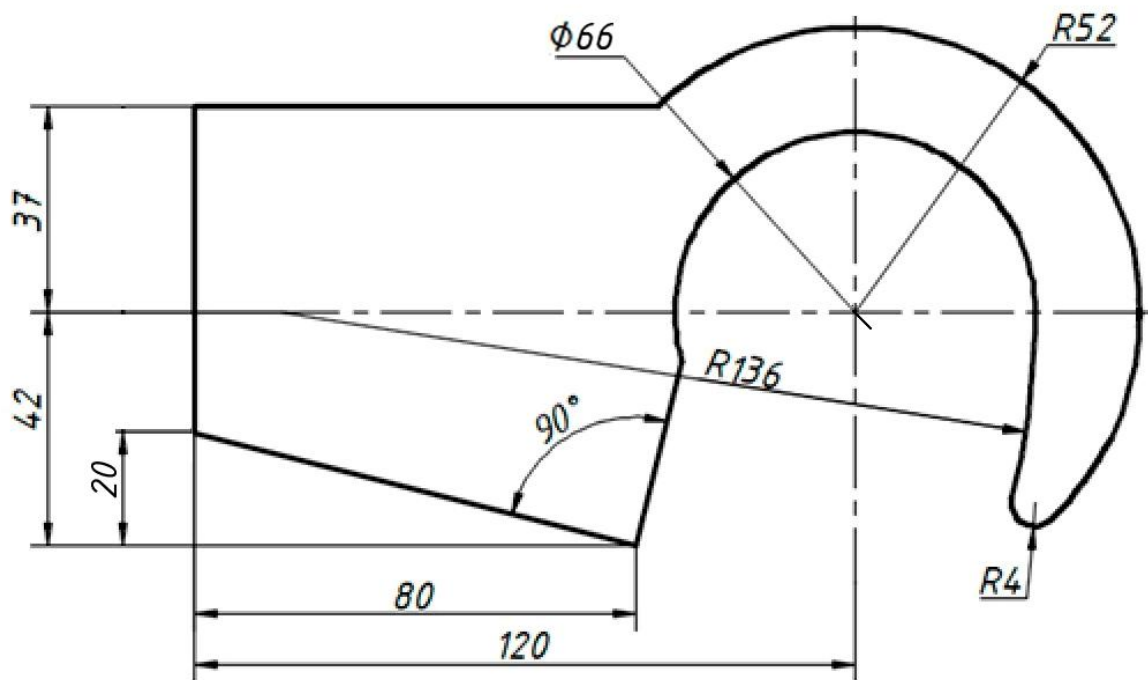


Variant 10

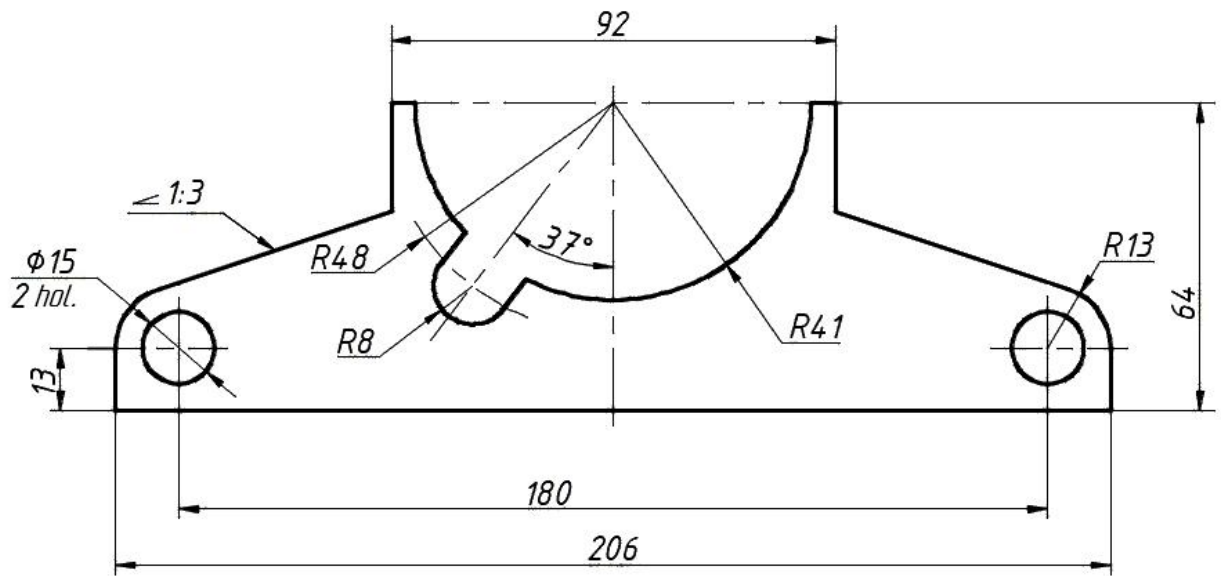




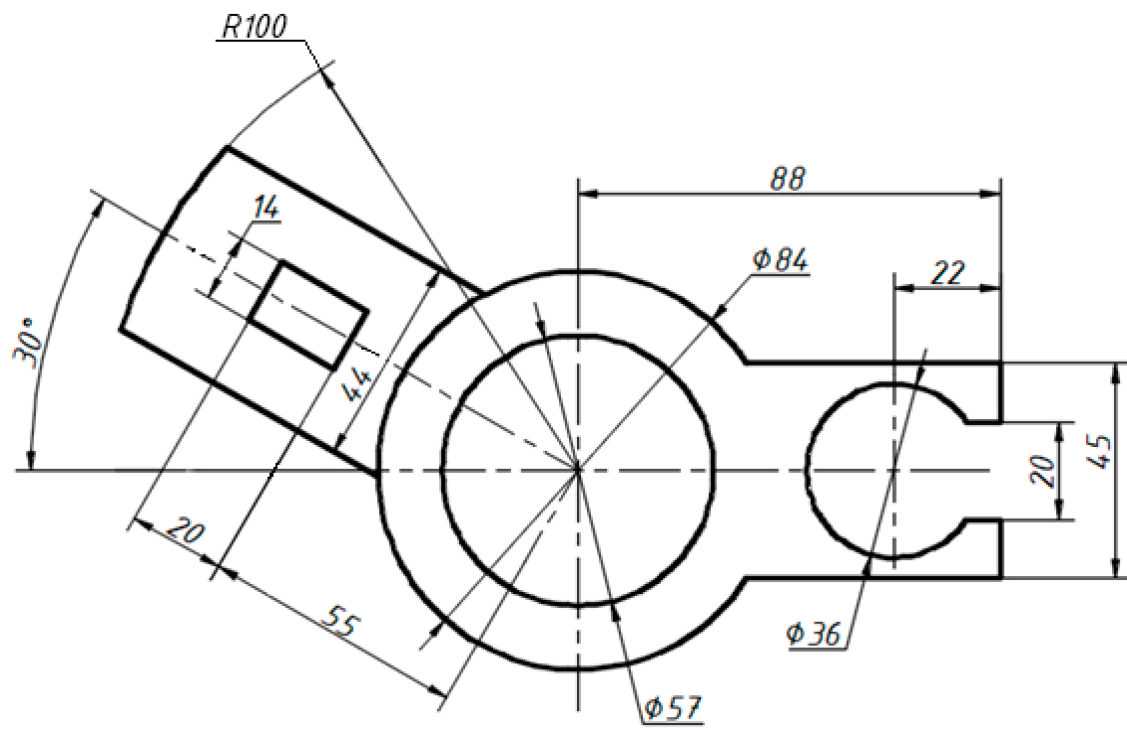
Variant 13



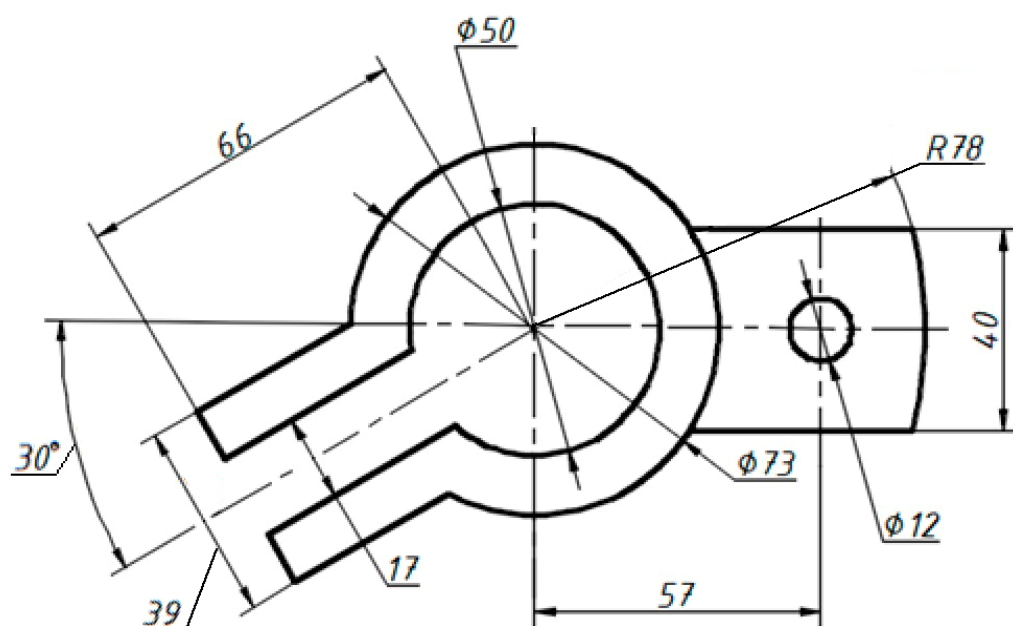
Variant 14



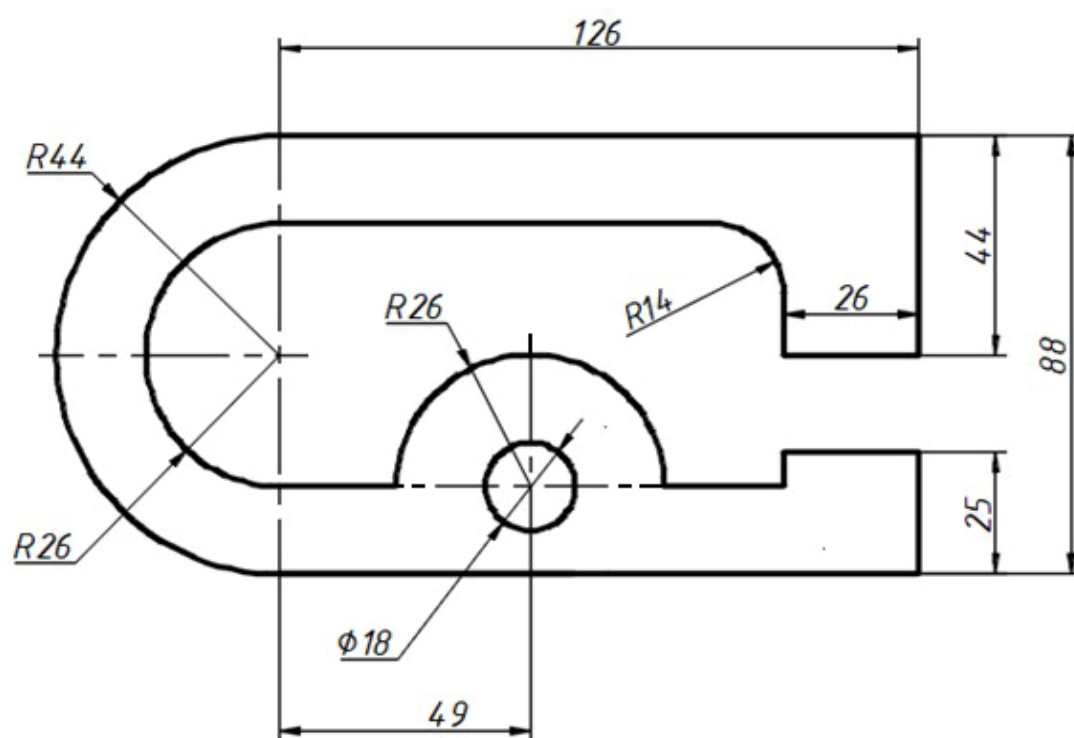
Variant 15



Variant 16



Variant 17

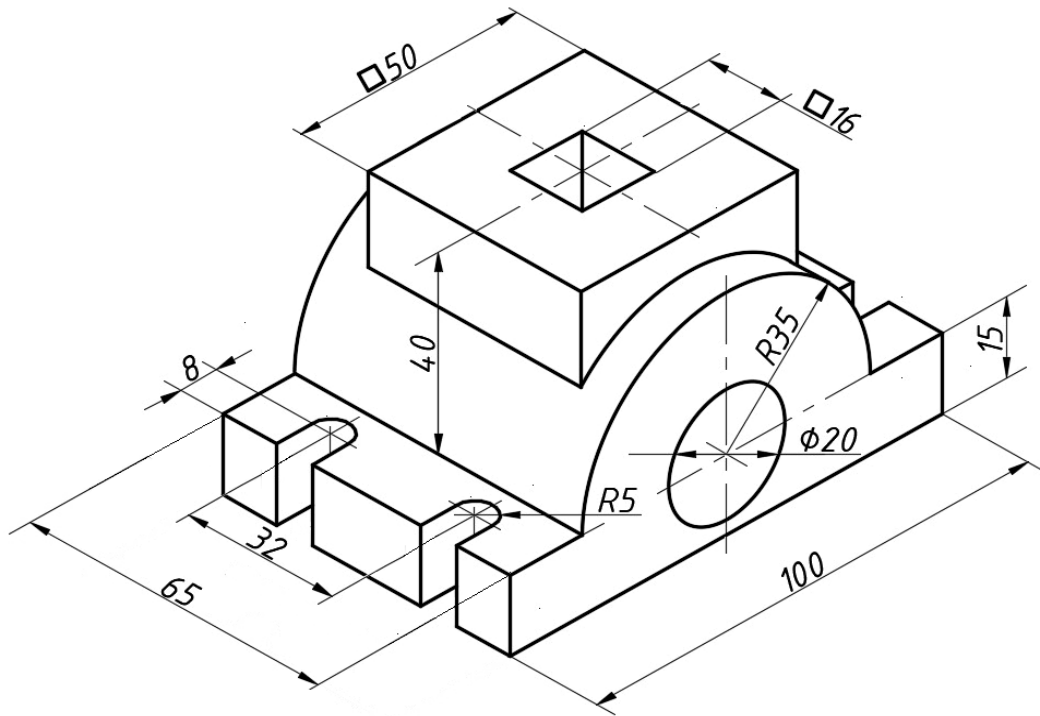


Variant 18

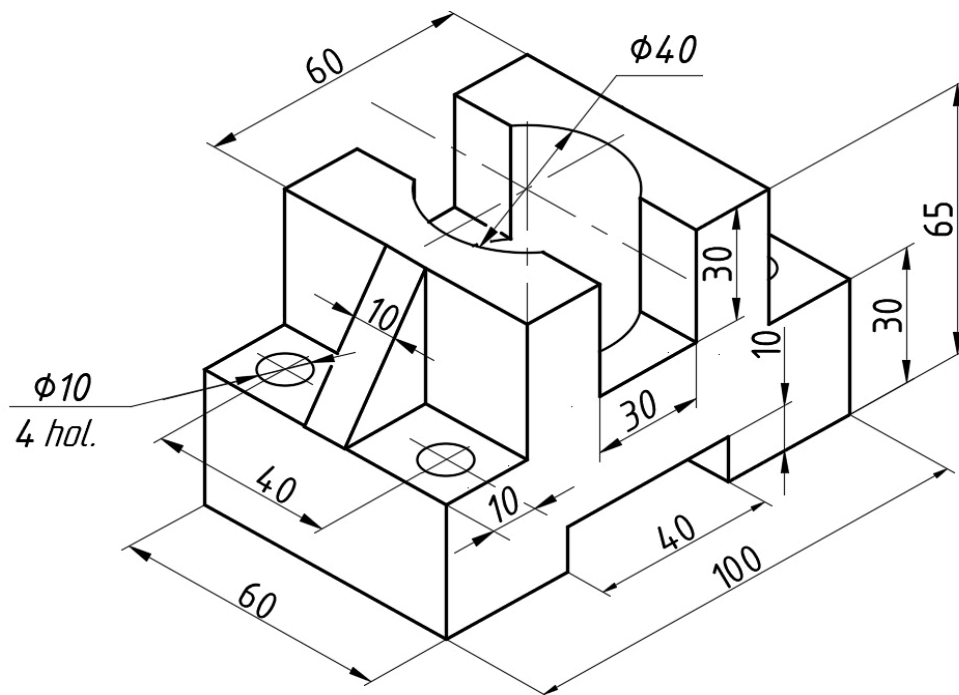


B.2. Projection Drawing

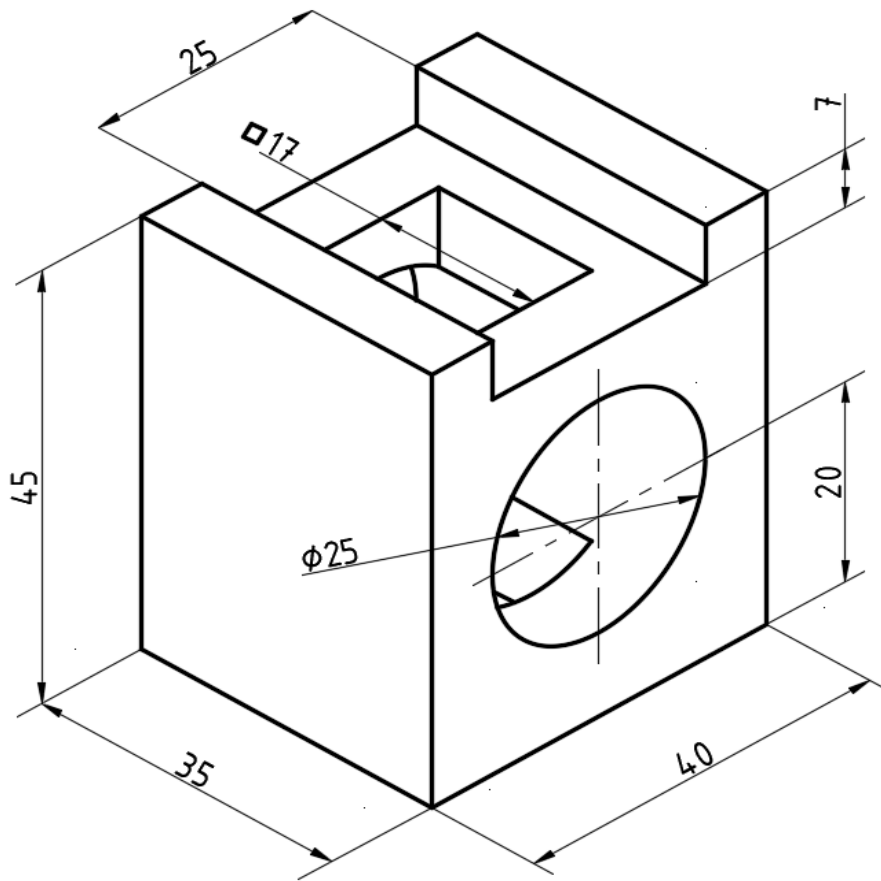
Based on the given illustration, construct all required views and sectional representations of the part. Throughout the drawing process, make use of layers as well as polar and object snap tracking, and complete the task by applying dimensions.



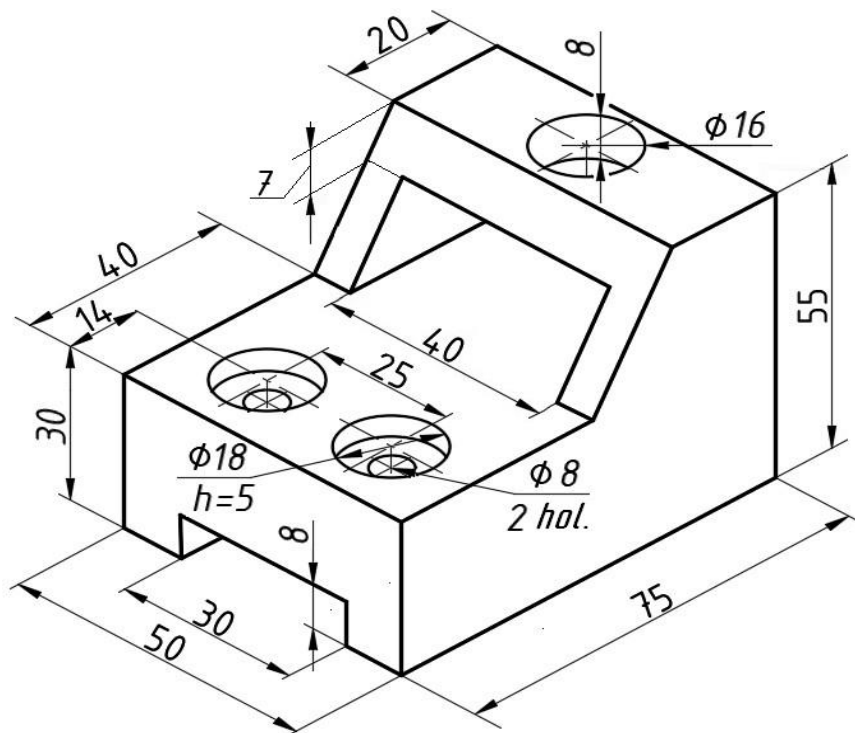
Variant 1



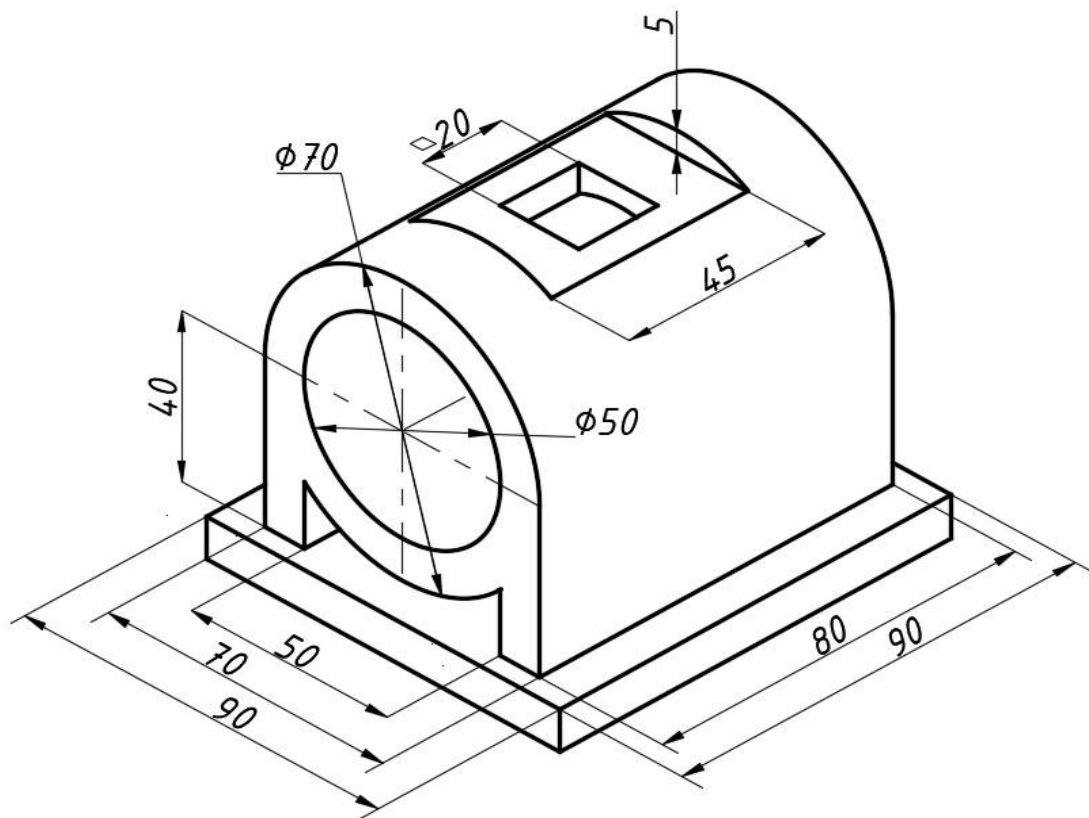
Variant 2



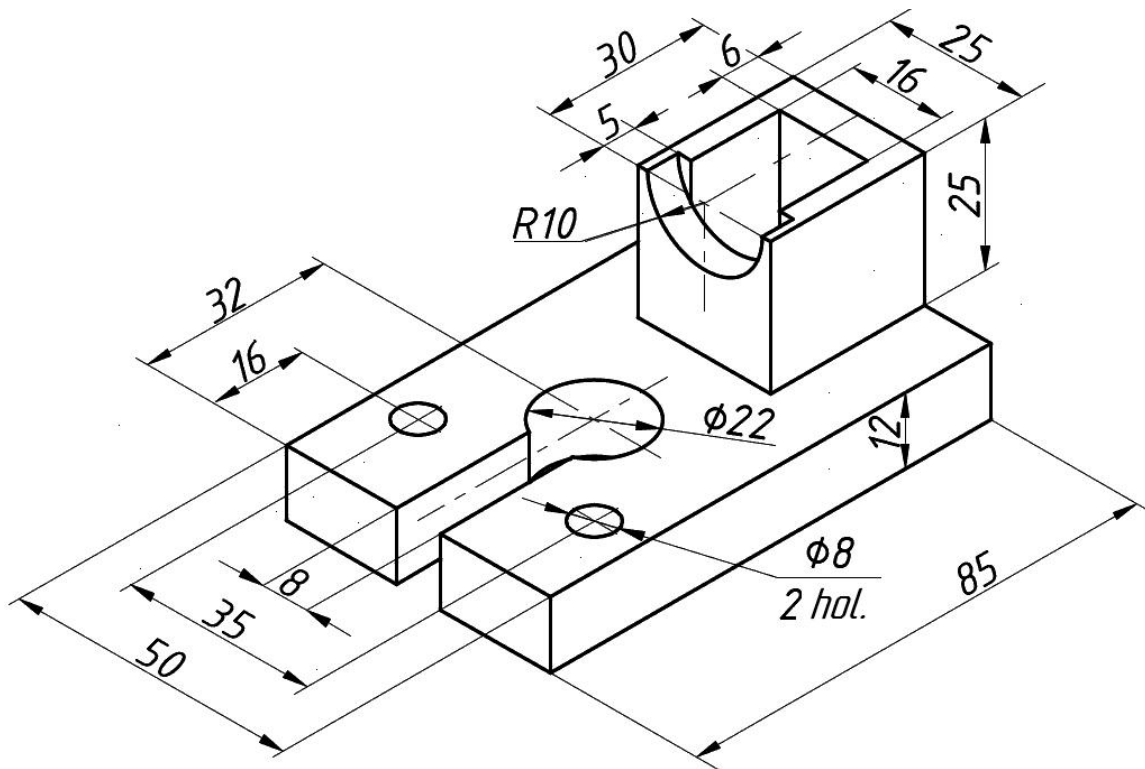
Variant 3



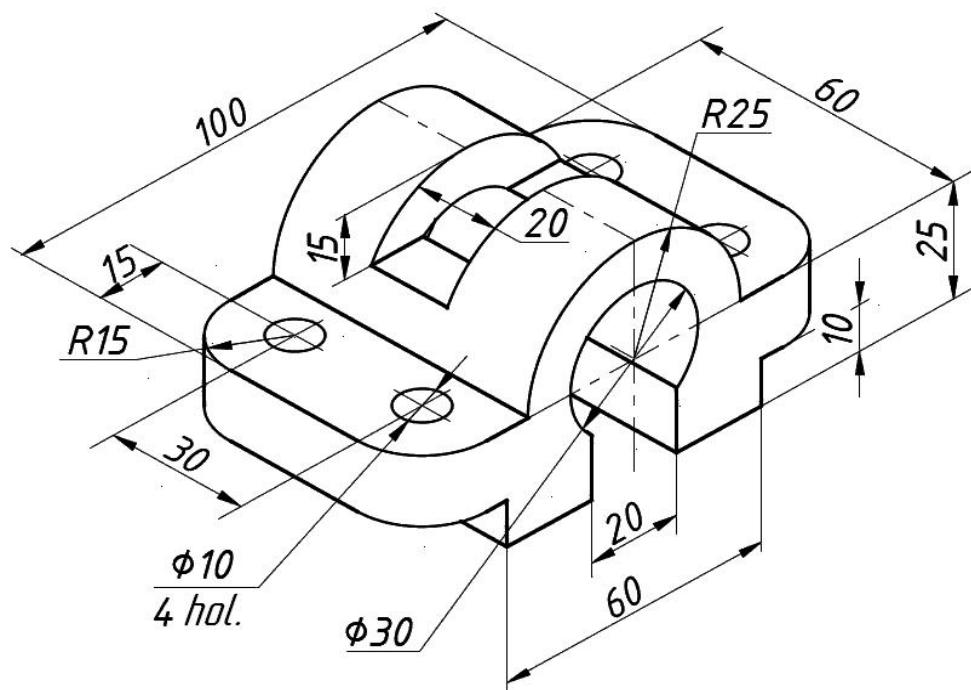
Variant 4



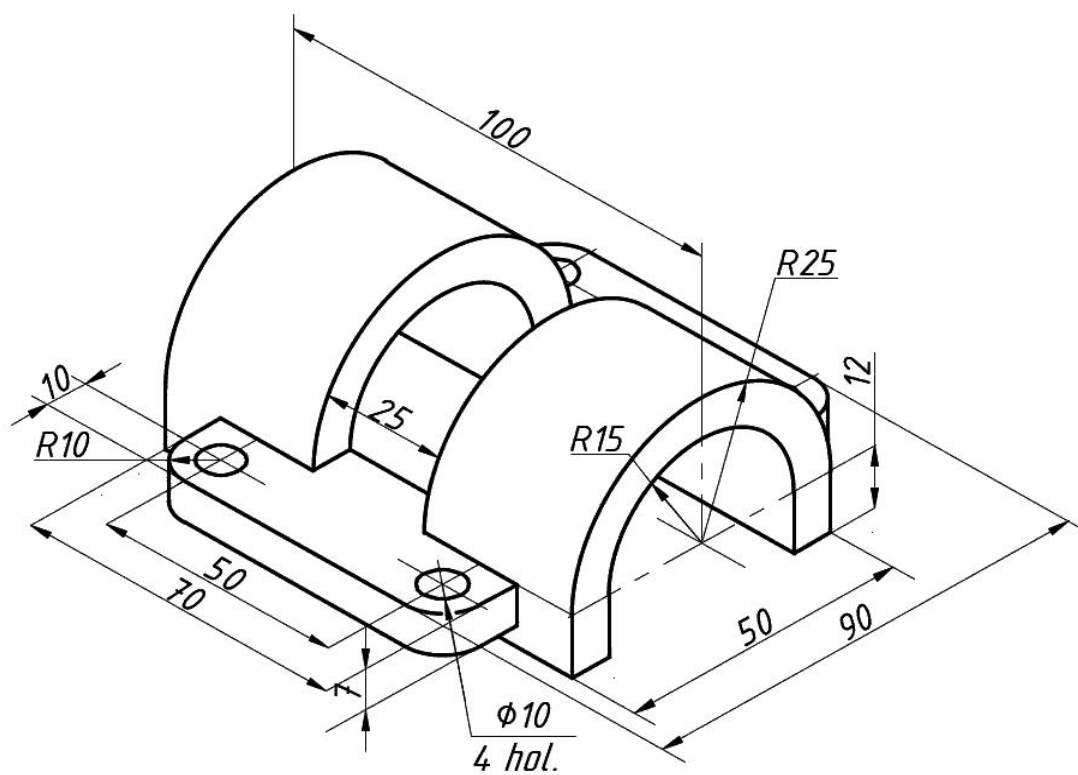
Variant 7



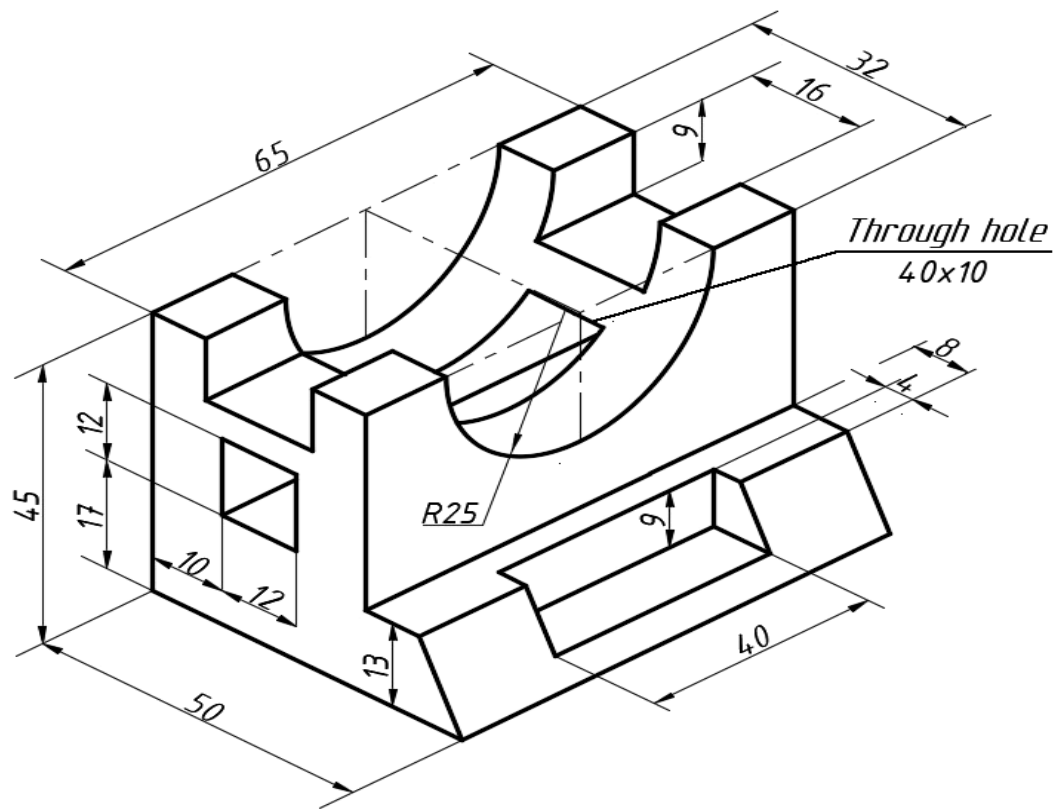
Variant 8



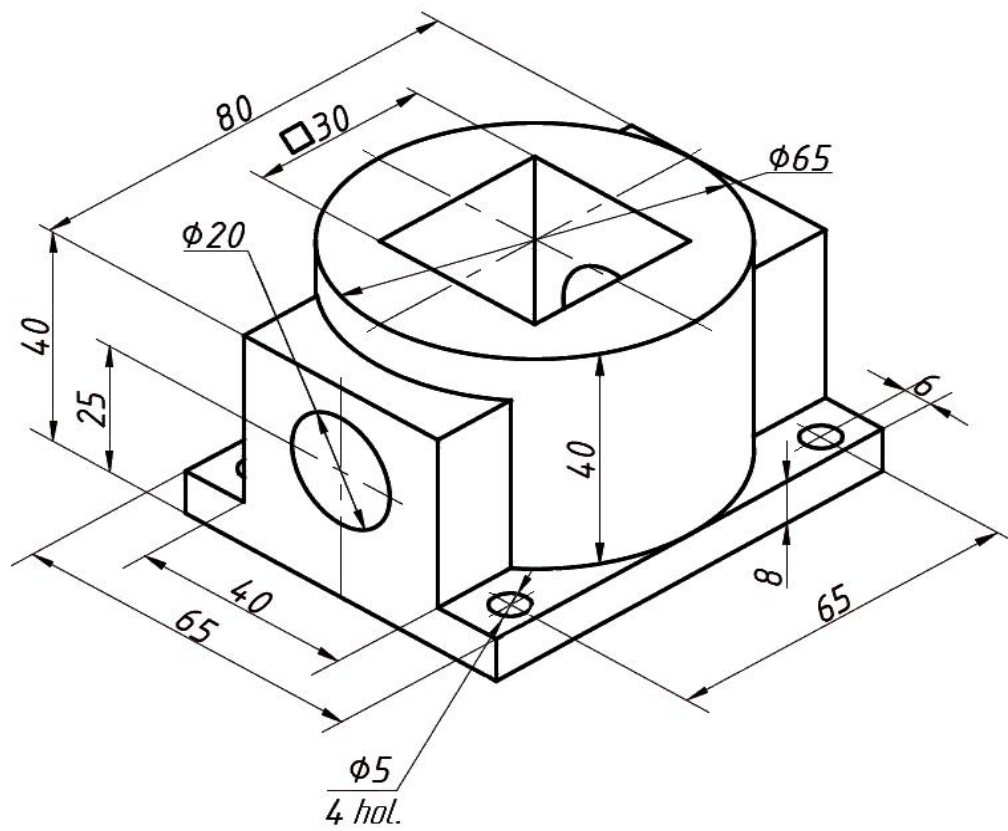
Variant 9



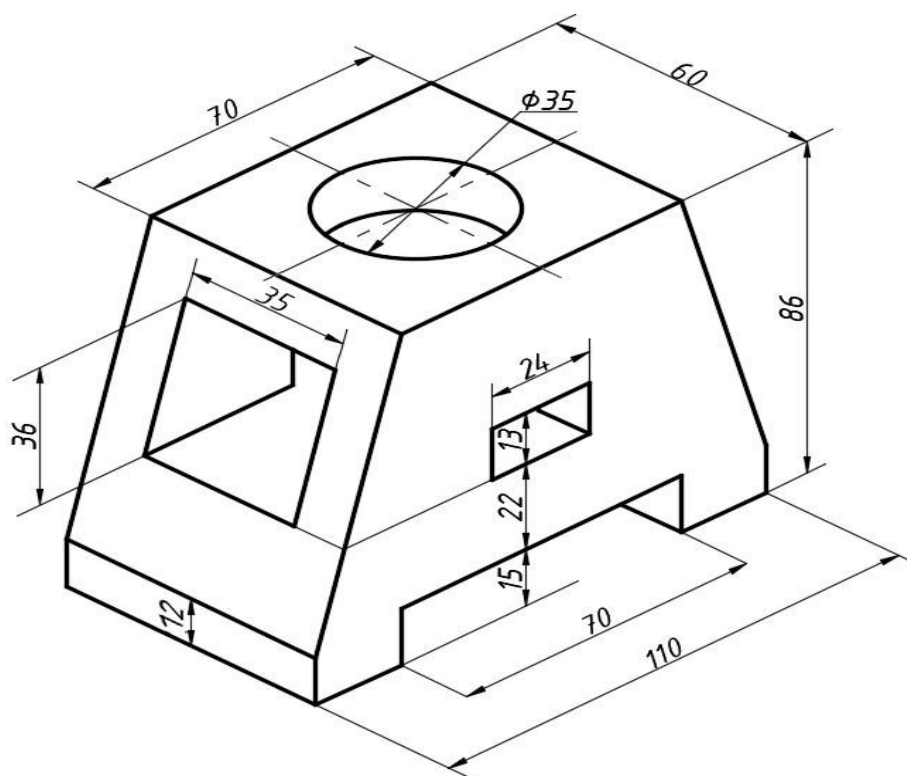
Variant 10



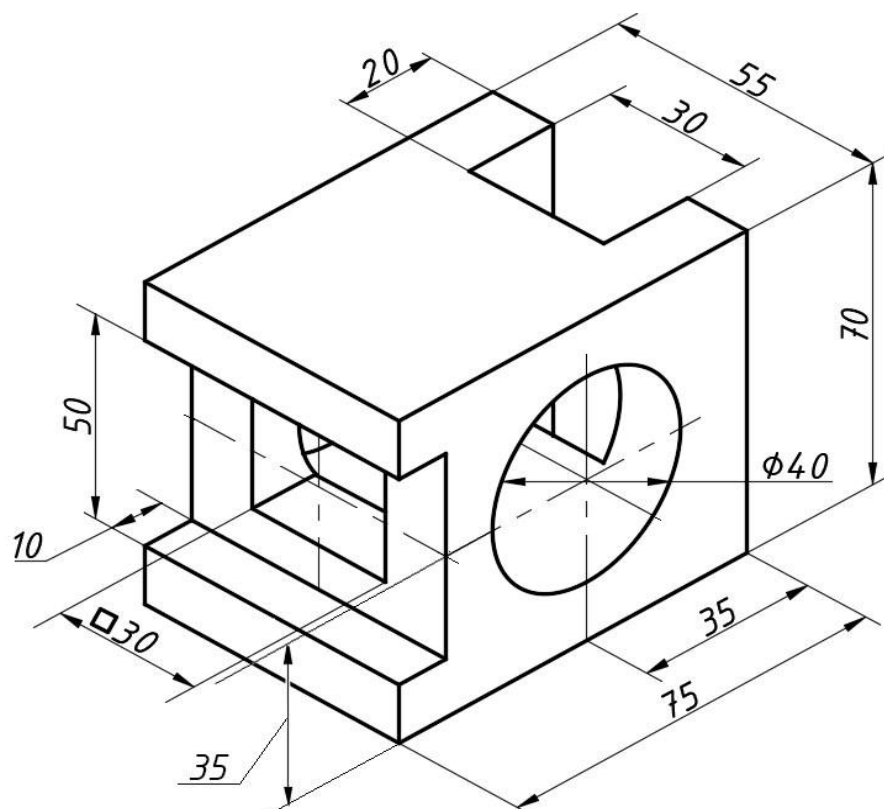
Variant 11



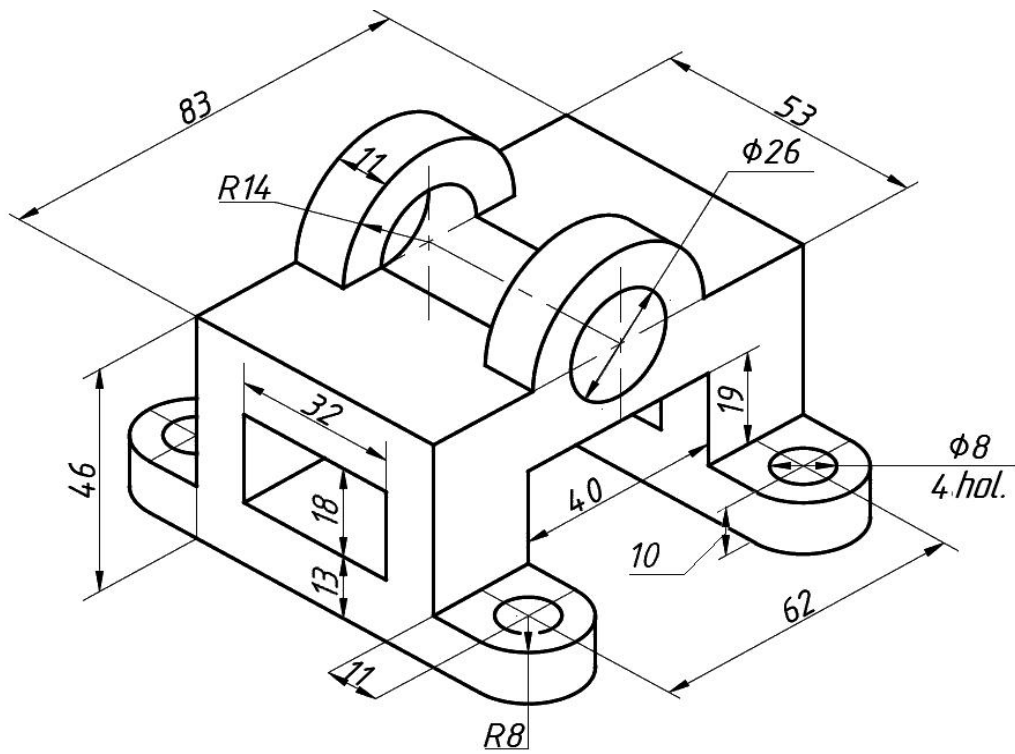
Variant 12



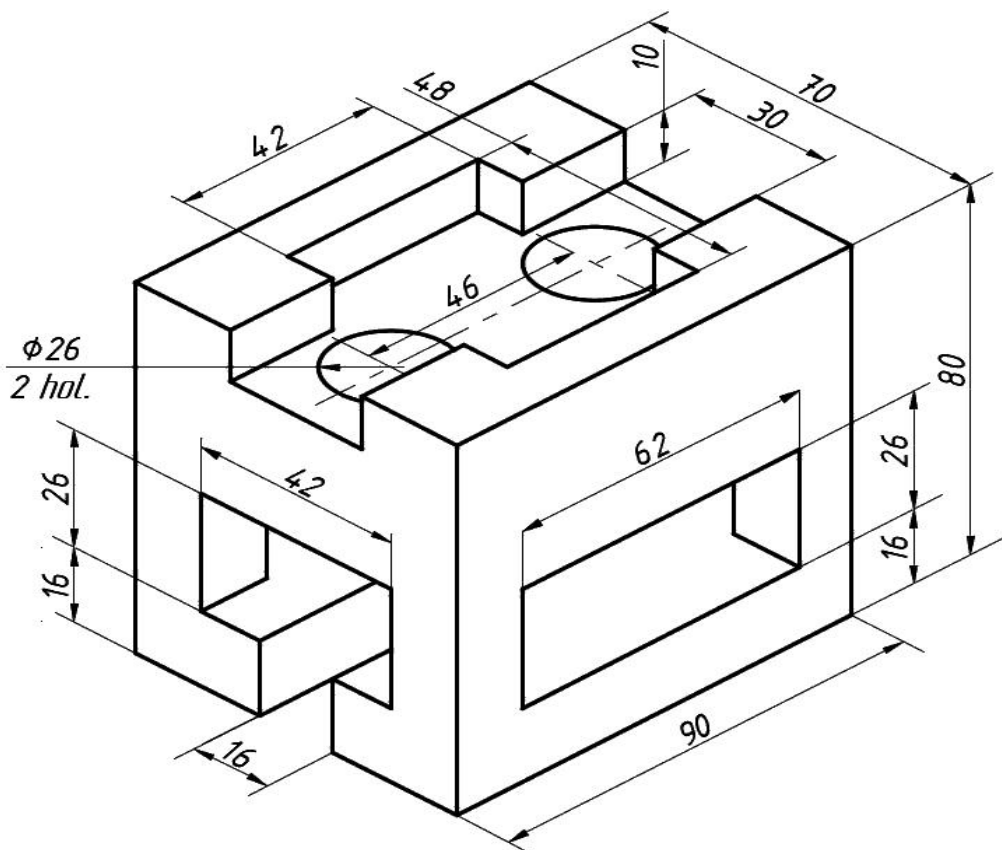
Variant 13



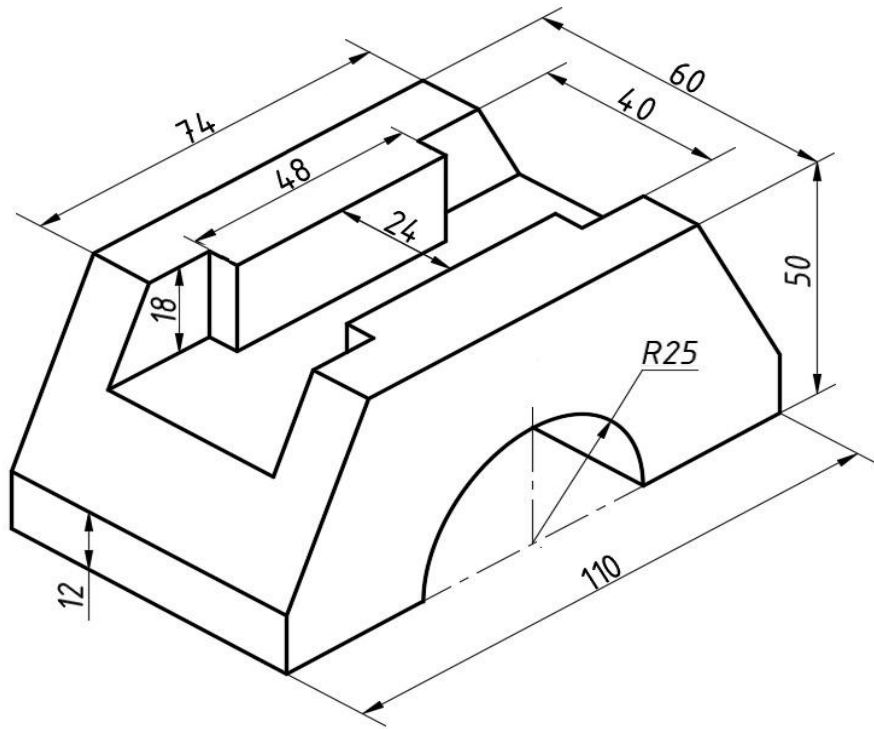
Variant 14



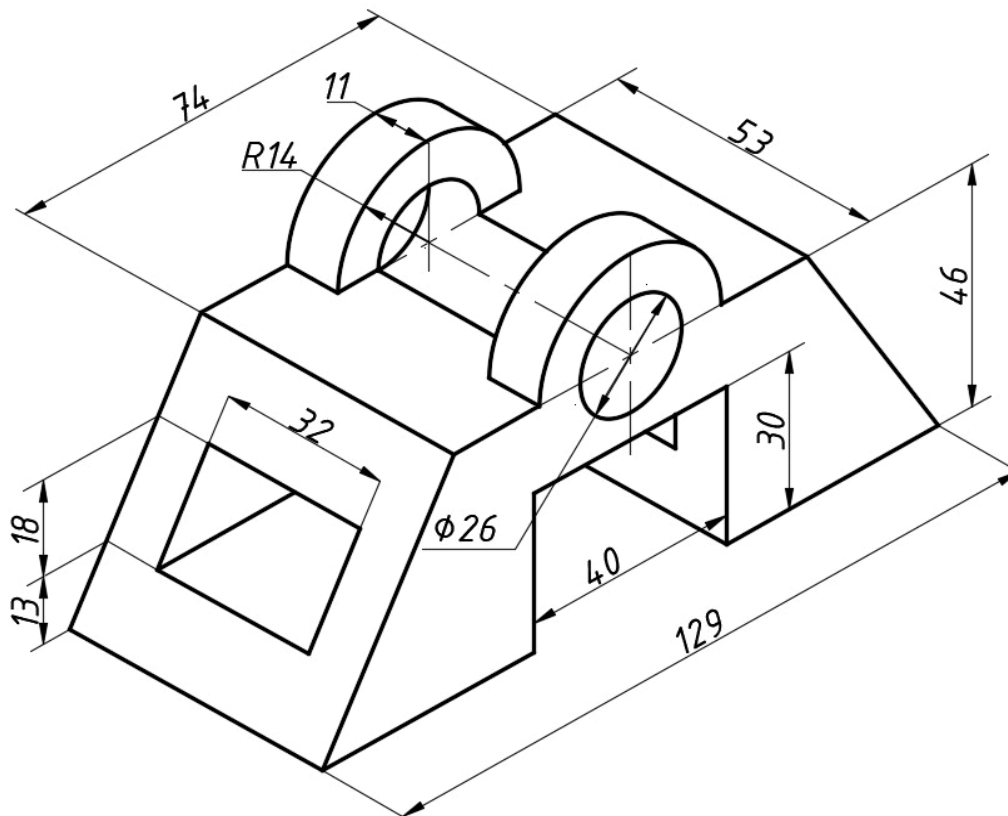
Variant 15



Variant 16



Variant 17



Variant 18



B.3. Three-Dimensional Constructions

Construct the three-dimensional figures from their given projections using solid primitives (parallelepiped, cylinder, cone, wedge, sphere) according to the assigned variant. Create the required views (front, top, left, and an axonometric view), add dimensions, and complete the drawing according to the standards.

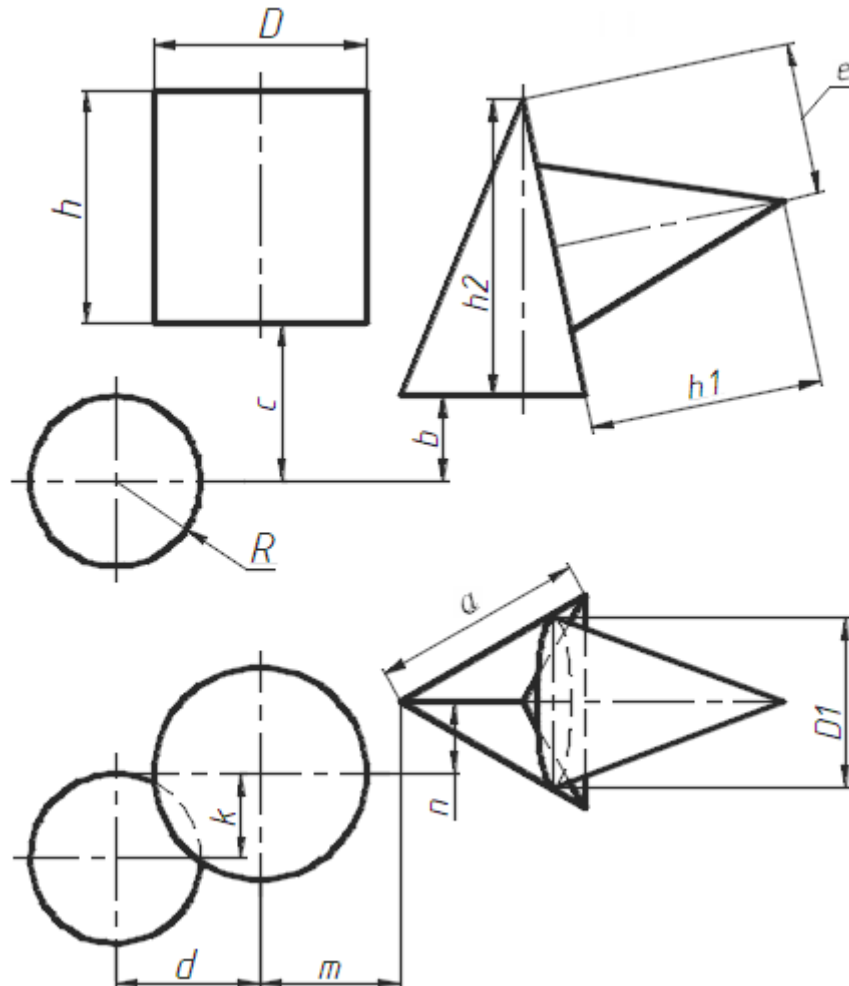


Fig. B.1. Variant 1 of primitive arrangement

Dimensions in Fig. B.1 (corresponding to the sizes of the primitives)														
Var.	D	h	$D1$	$h1$	a	$h2$	R	b	c	d	k	m	n	e
1	70	80	40	55	50	85	30	15	30	35	25	45	10	45
5	65	80	45	50	55	60	25	5	30	5	5	60	-10	30
9	70	65	20	55	40	30	25	10	0	-30	-60	35	-35	10
13	75	70	30	15	55	80	20	0	-15	60	30	-45	70	0
17	55	70	55	70	45	50	30	45	0	60	0	30	30	30
21	65	65	35	50	55	65	30	5	25	10	5	60	10	30

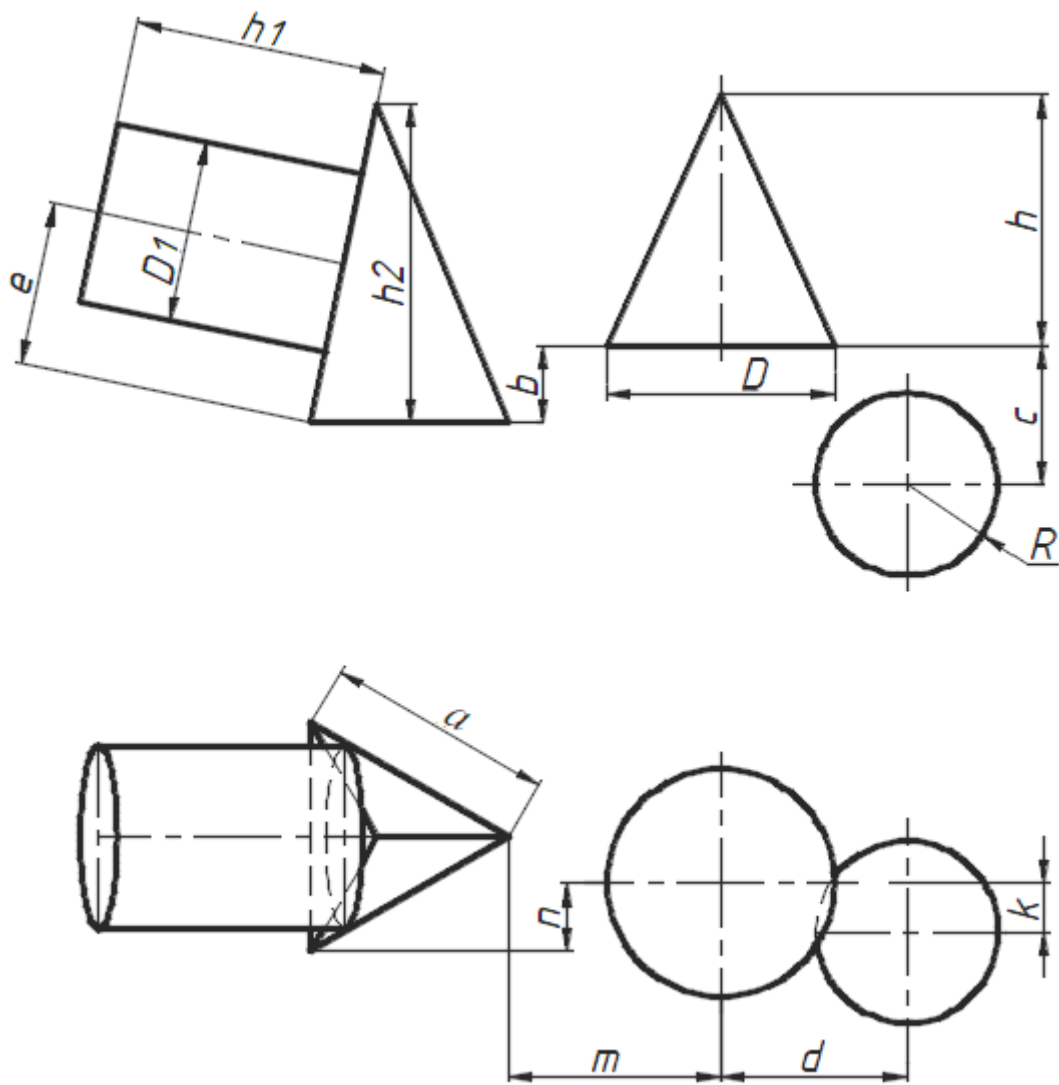


Fig. B.2. Variant 2 of primitive arrangement

Dimensions in Fig. B.2 (corresponding to the sizes of the primitives)														
Var.	D	h	$D1$	$h1$	a	$h2$	R	b	c	d	k	m	n	e
2	55	70	55	70	45	50	30	45	35	0	0	35	30	30
6	70	80	40	55	50	85	30	15	45	35	35	80	10	45
10	70	65	40	55	40	30	25	10	0	0	60	25	-35	40
14	75	70	30	15	55	80	20	0	15	15	-60	80	0	0
18	65	80	45	50	55	60	25	5	30	5	5	60	-10	30
22	55	65	55	55	40	30	25	10	0	35	60	25	-30	40

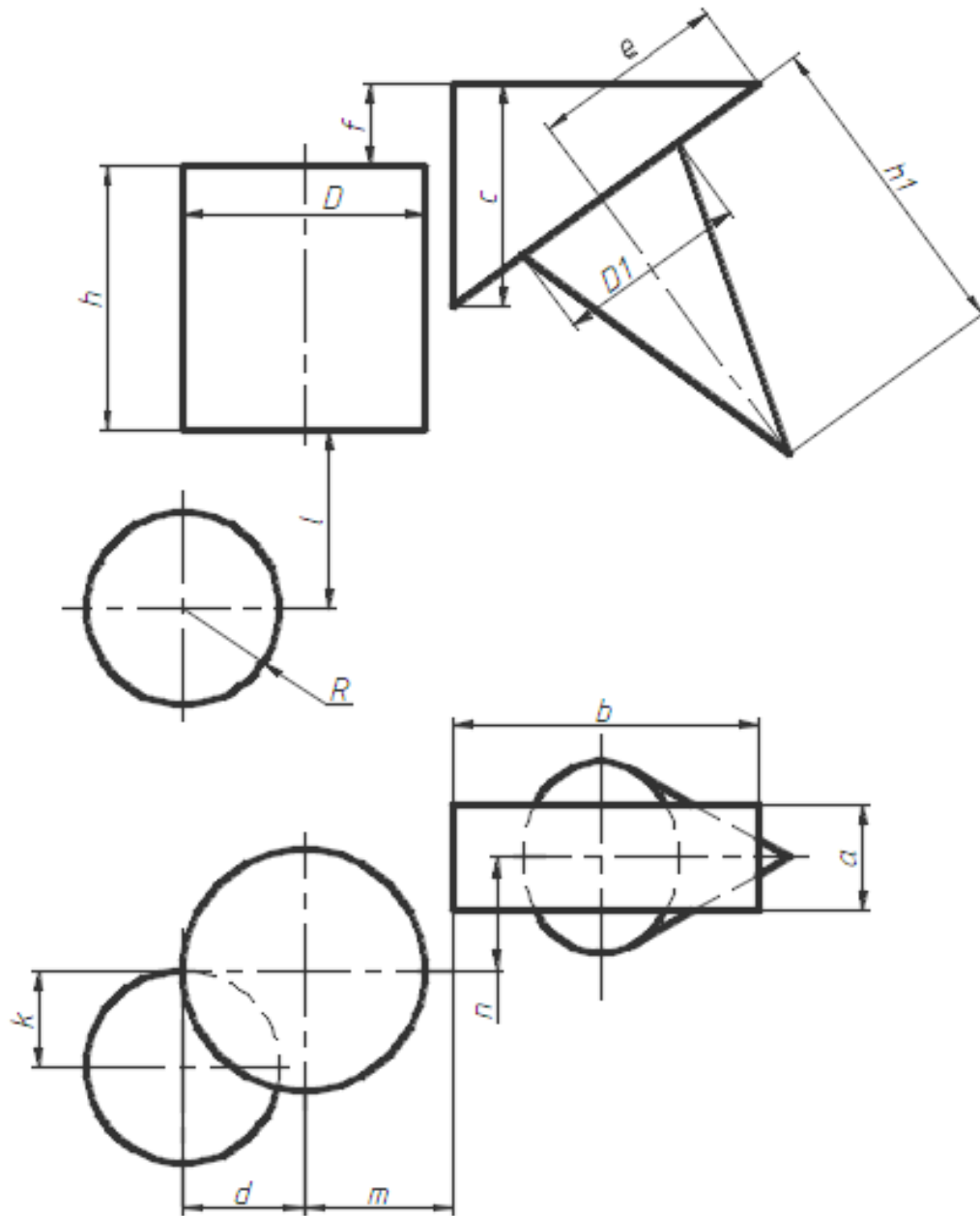


Fig. B.3. Variant 3 of primitive arrangement

Dimensions in Fig. B.3 (corresponding to the sizes of the primitives)															
Var.	D	h	$D1$	$h1$	R	a	b	c	d	m	n	k	l	f	e
3	55	70	45	70	25	30	50	45	30	30	0	25	30	15	30
7	70	80	40	55	30	85	30	15	70	35	-10	65	0	-10	15
11	65	65	40	45	20	30	45	10	-20	40	20	30	25	0	5
15	75	70	30	15	25	40	20	50	45	45	30	35	0	-15	30
19	65	80	45	50	35	60	25	55	30	5	80	30	40	0	30
23	60	75	40	45	20	30	45	10	20	40	20	30	25	15	5

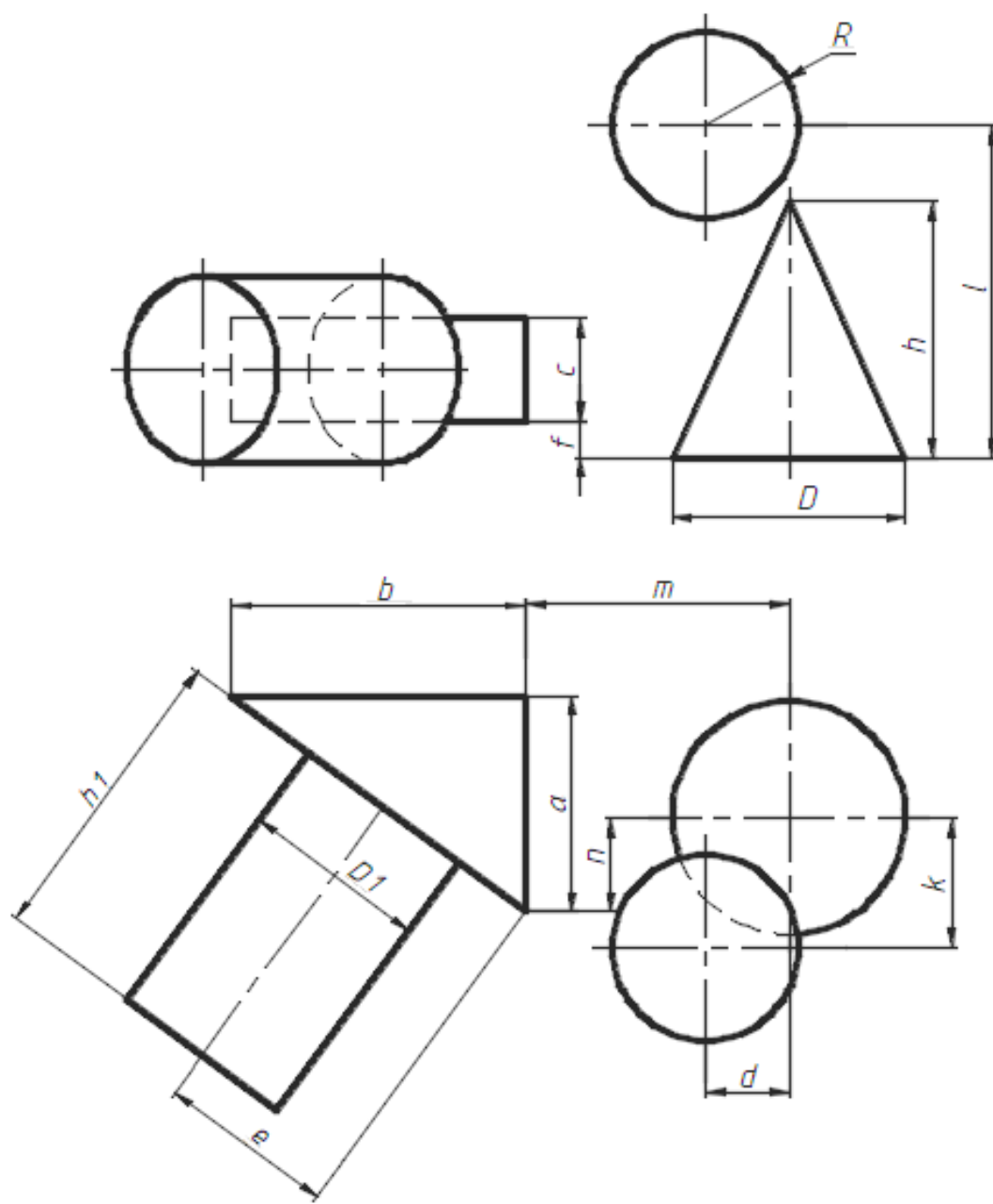
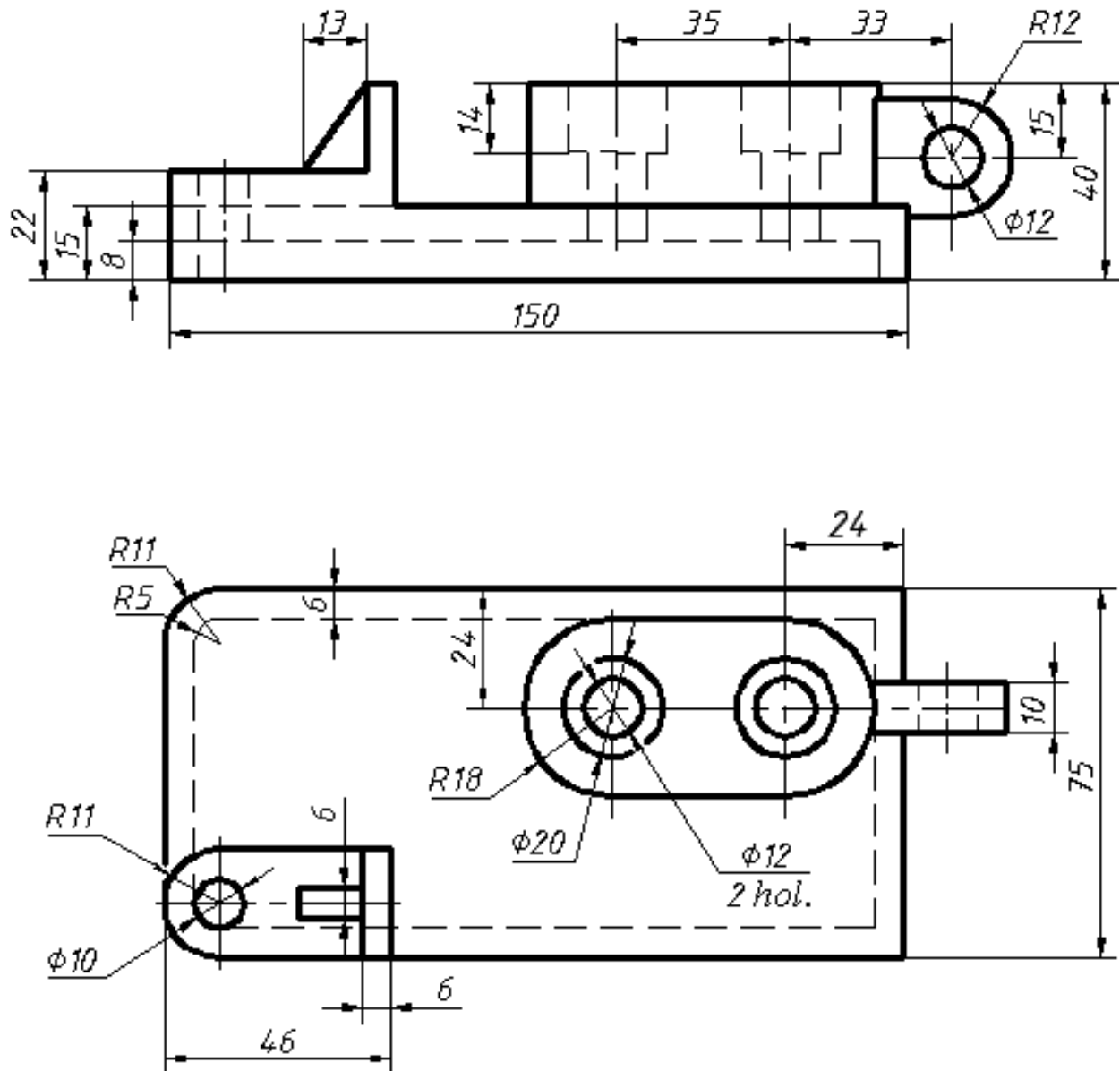


Fig. B.4. Variant 4 of primitive arrangement

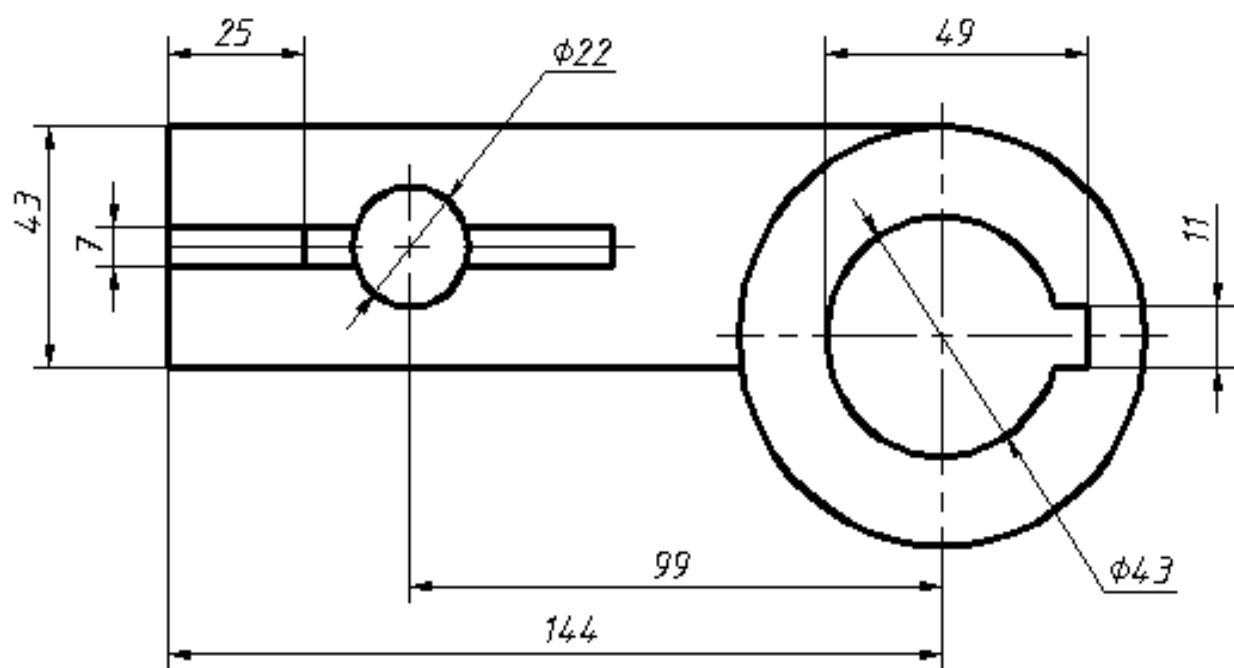
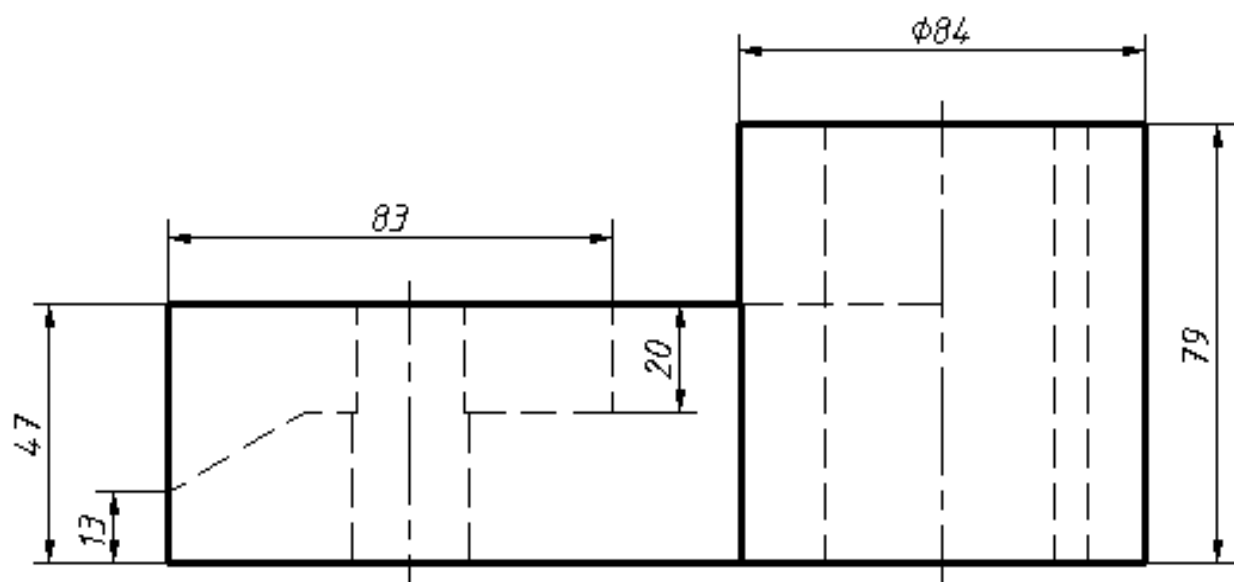
Dimensions in Fig. B.4 (corresponding to the sizes of the primitives)															
Var.	D	h	$D1$	$h1$	R	a	B	c	d	m	n	k	l	f	e
4	65	65	40	45	20	30	45	10	-20	40	20	30	75	0	5
8	55	70	45	70	25	30	50	45	40	30	0	45	55	-15	30
12	75	70	30	15	25	40	20	50	45	65	30	65	0	15	30
16	65	80	45	50	35	60	25	55	30	45	-20	35	70	0	30
20	70	80	40	55	30	85	30	35	0	35	-10	65	0	-40	15
24	60	75	30	25	25	40	20	50	45	65	30	-65	0	15	30

B.4. Creating a Solid Model from Its Projections

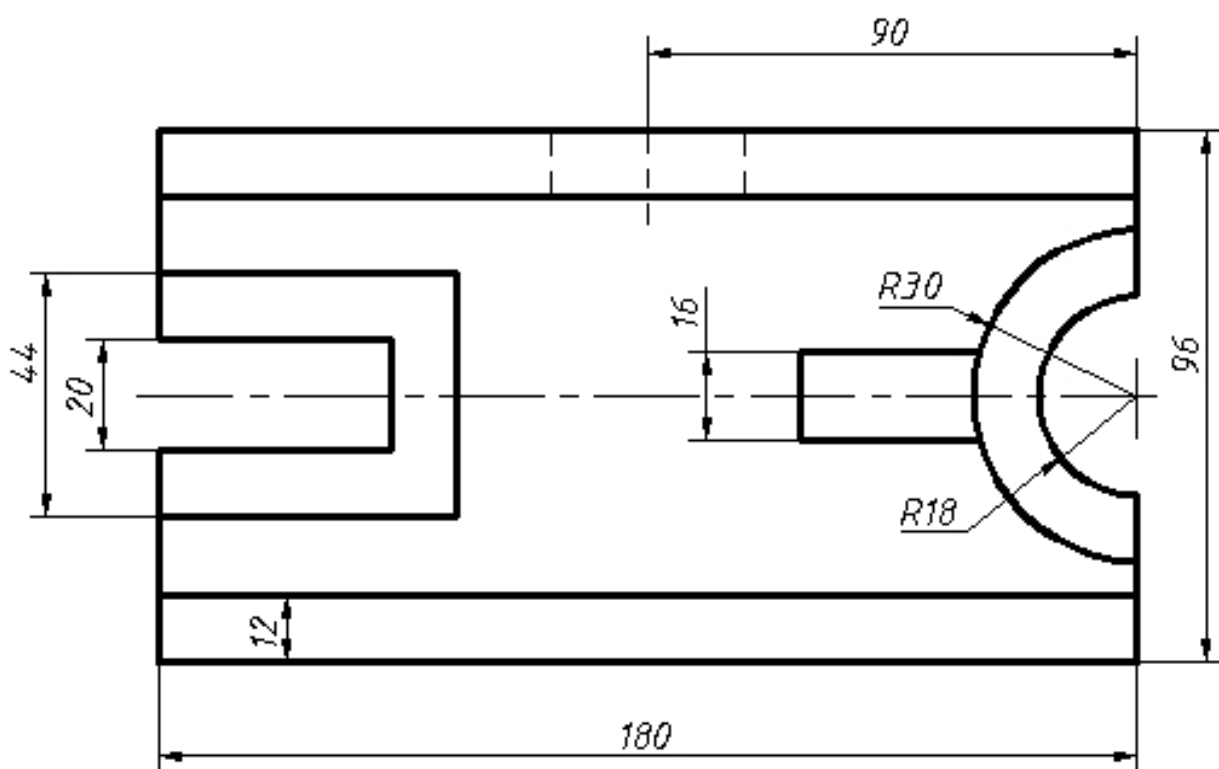
Construct a three-dimensional solid model of the part based on its two projections.

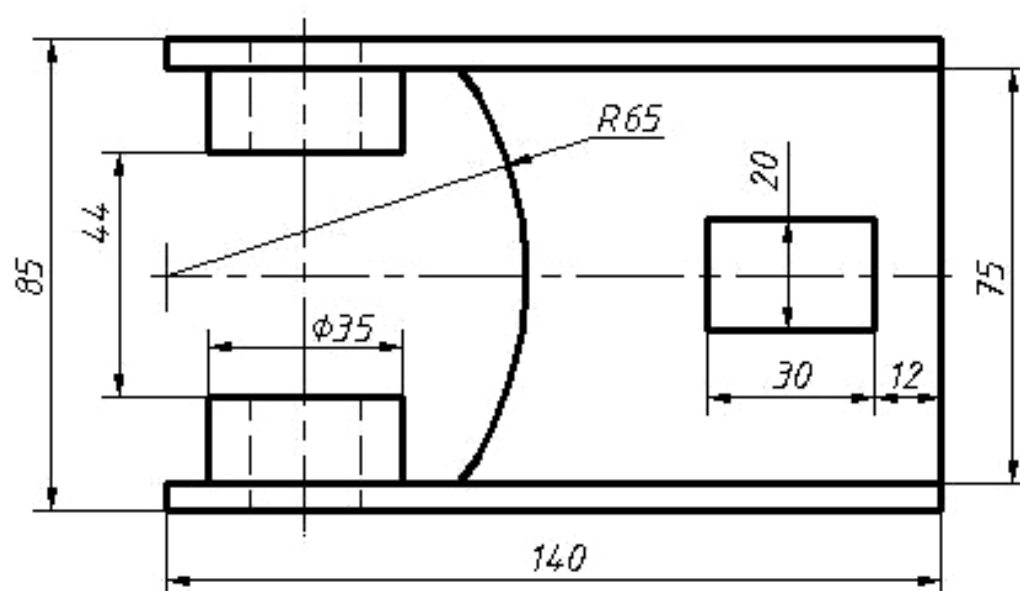
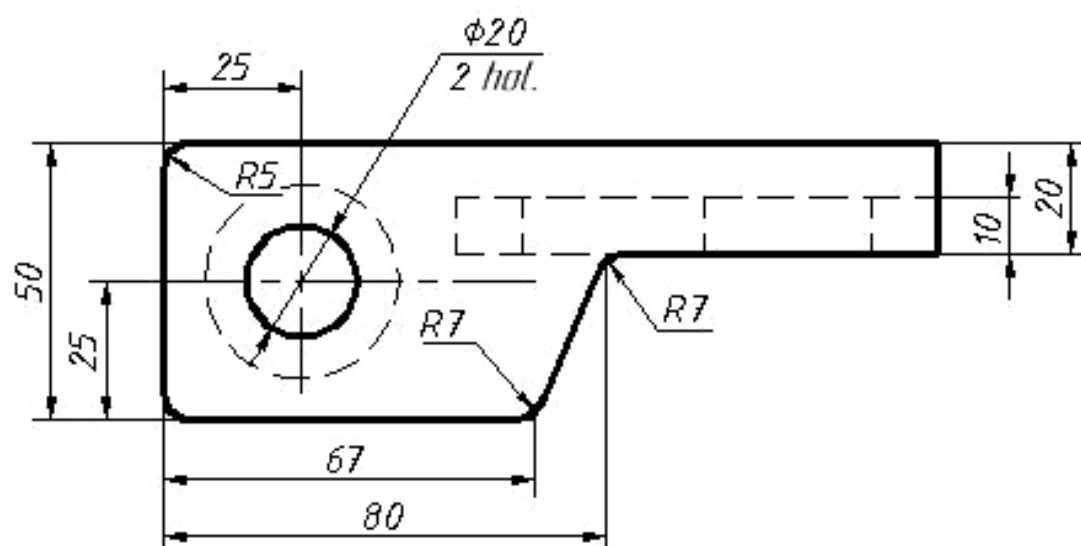


Variant 1

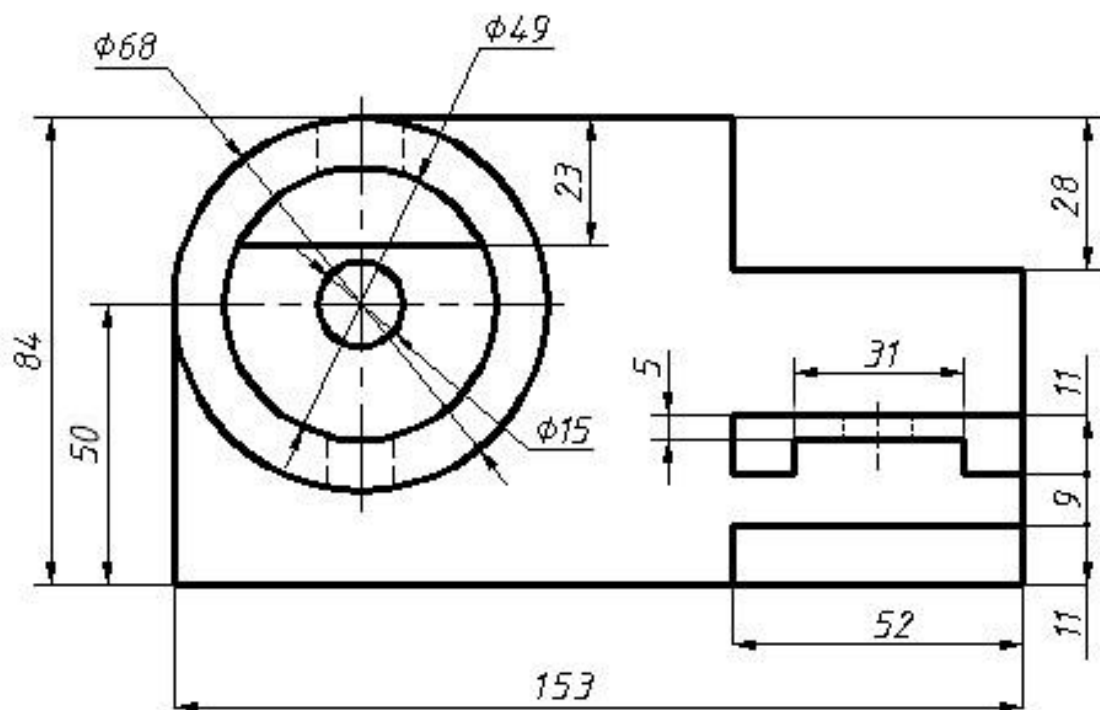
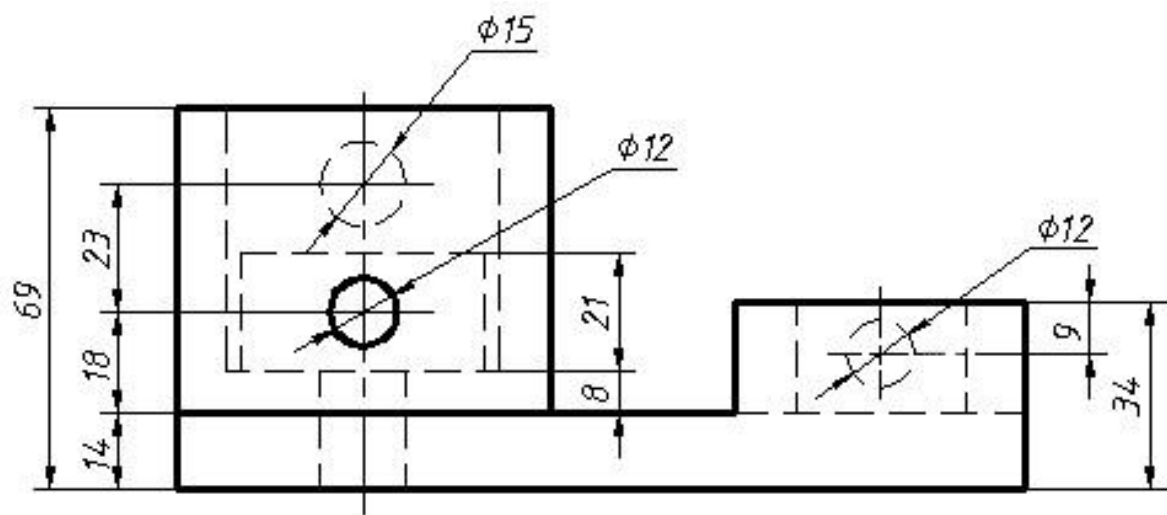


Variant 2

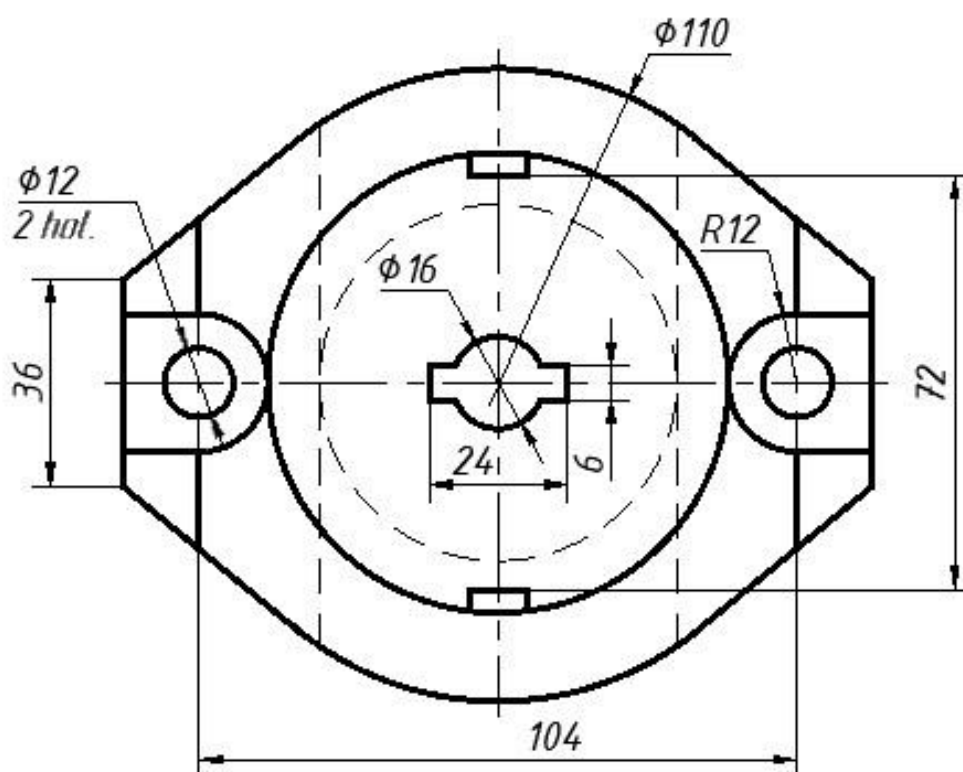
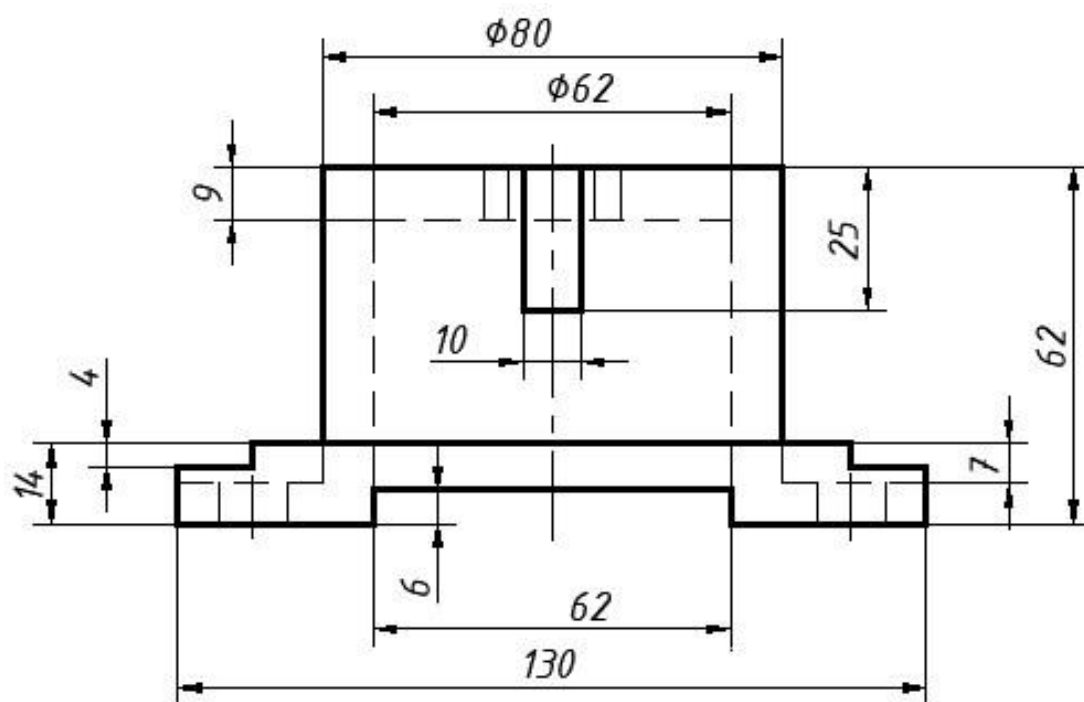




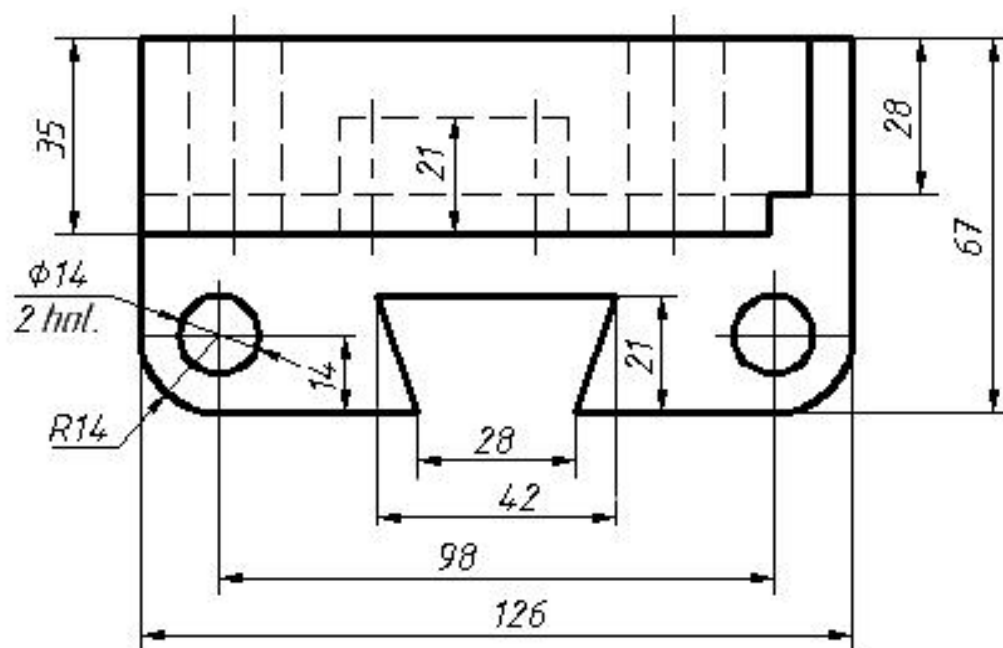
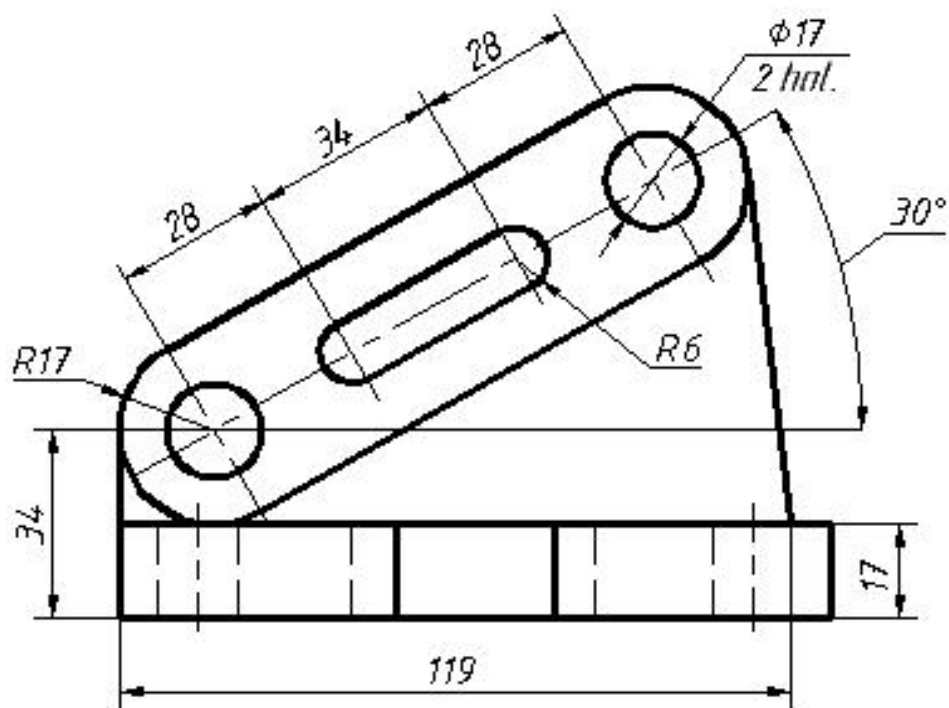
Variant 4



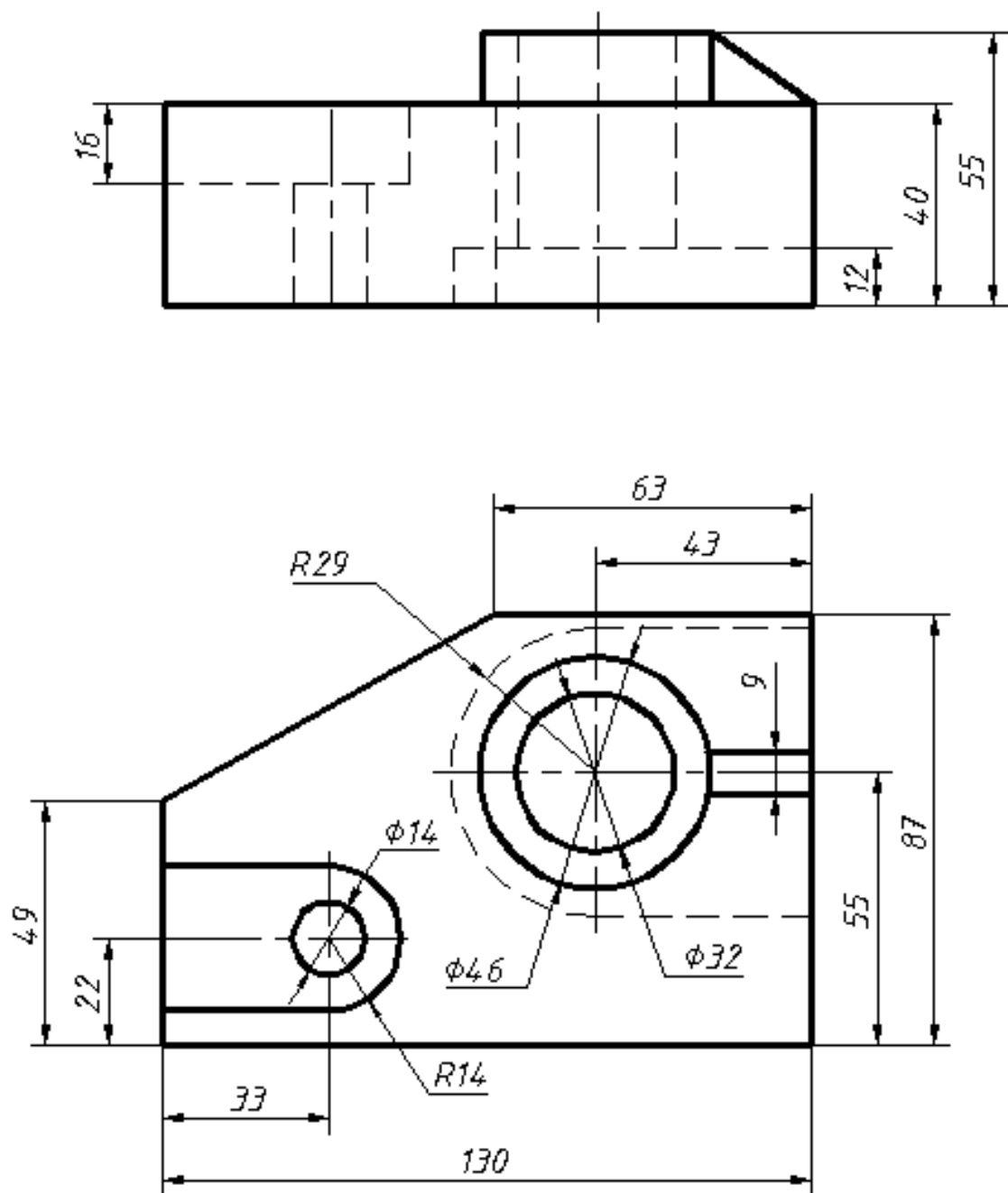
Variant 5



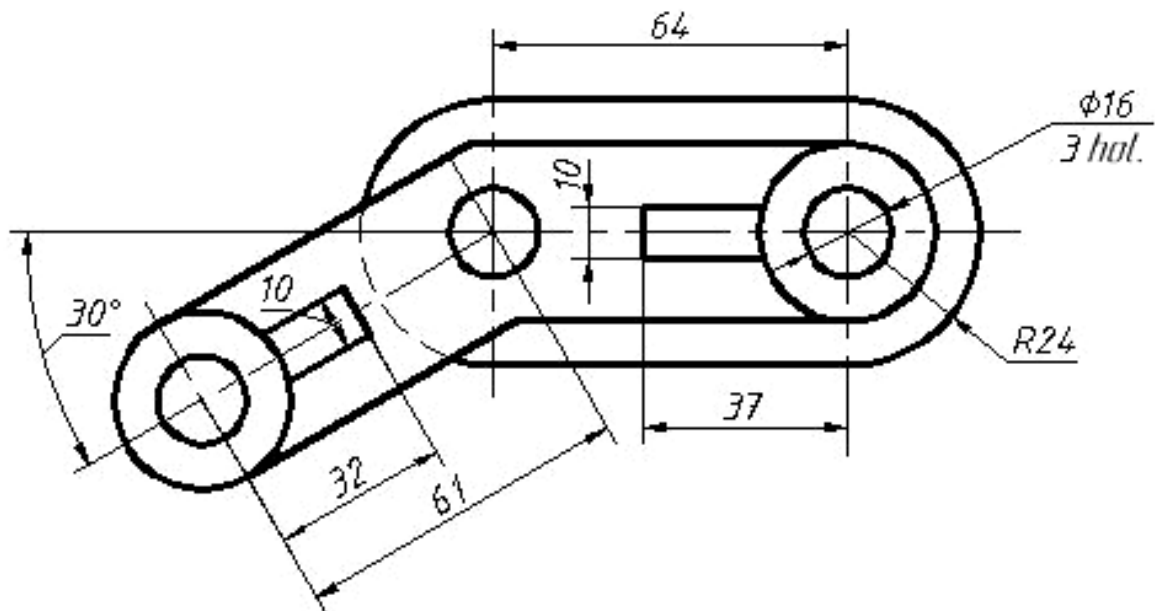
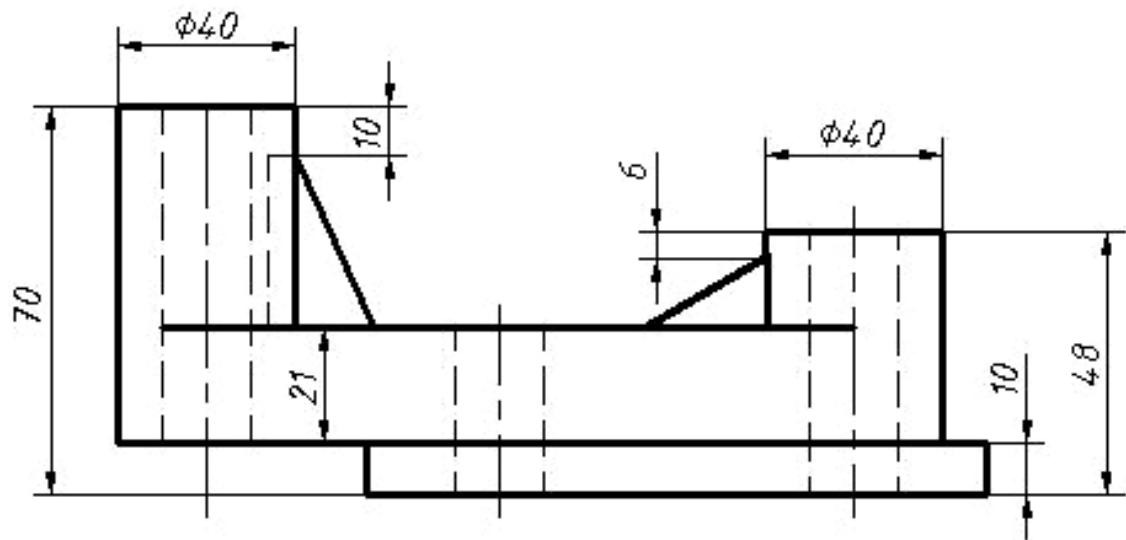
Variant 6



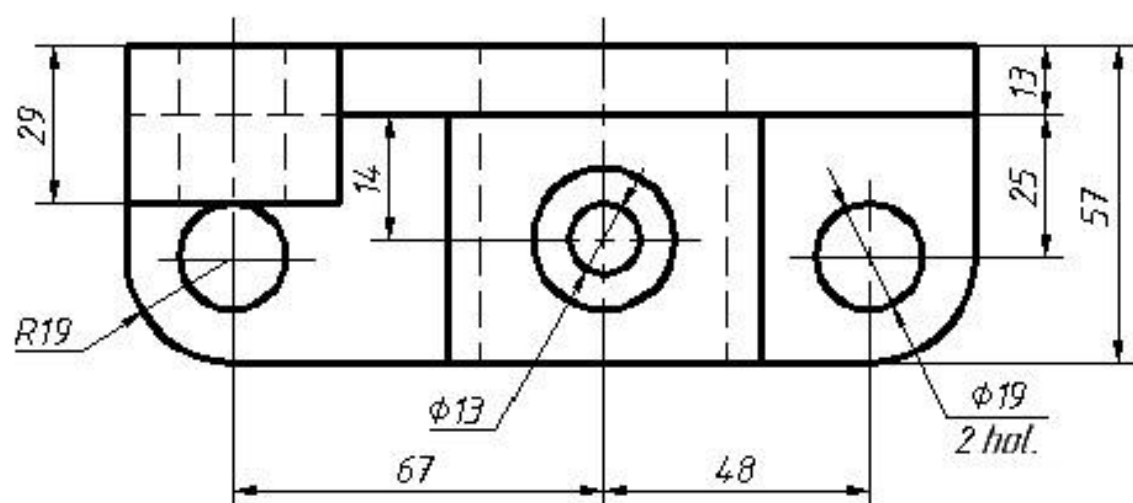
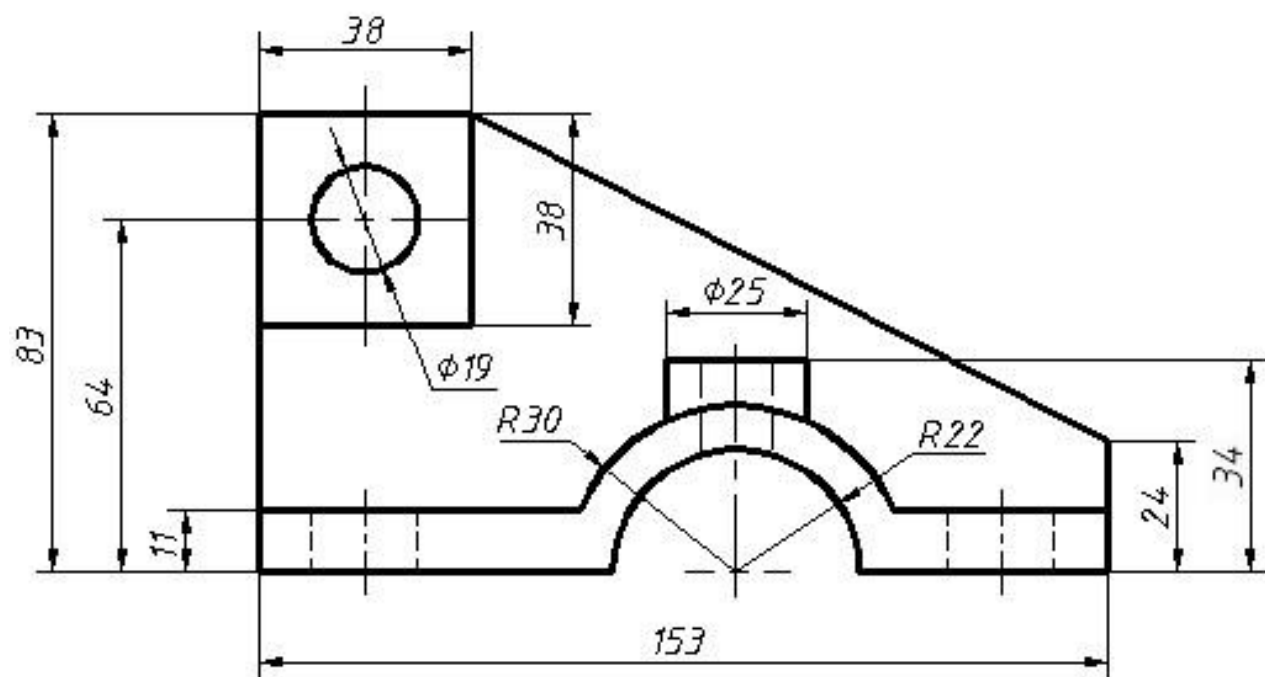
Variant 7



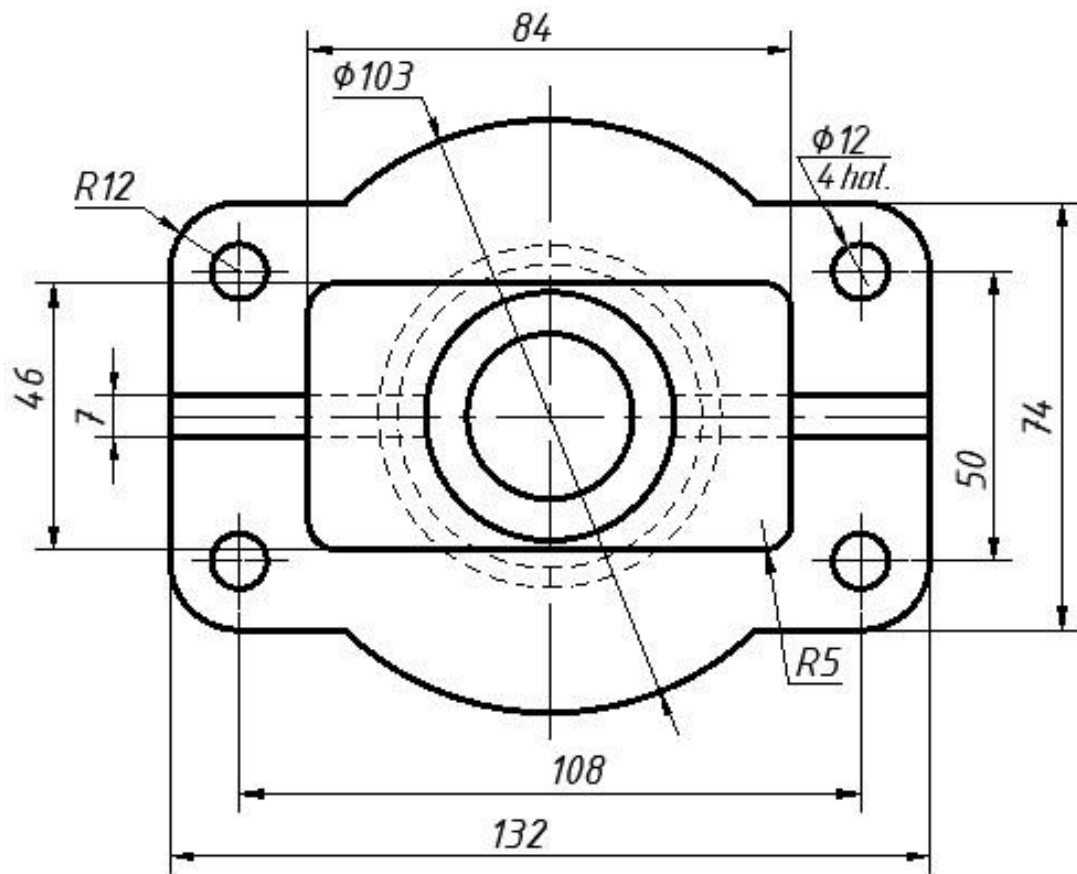
Variant 8

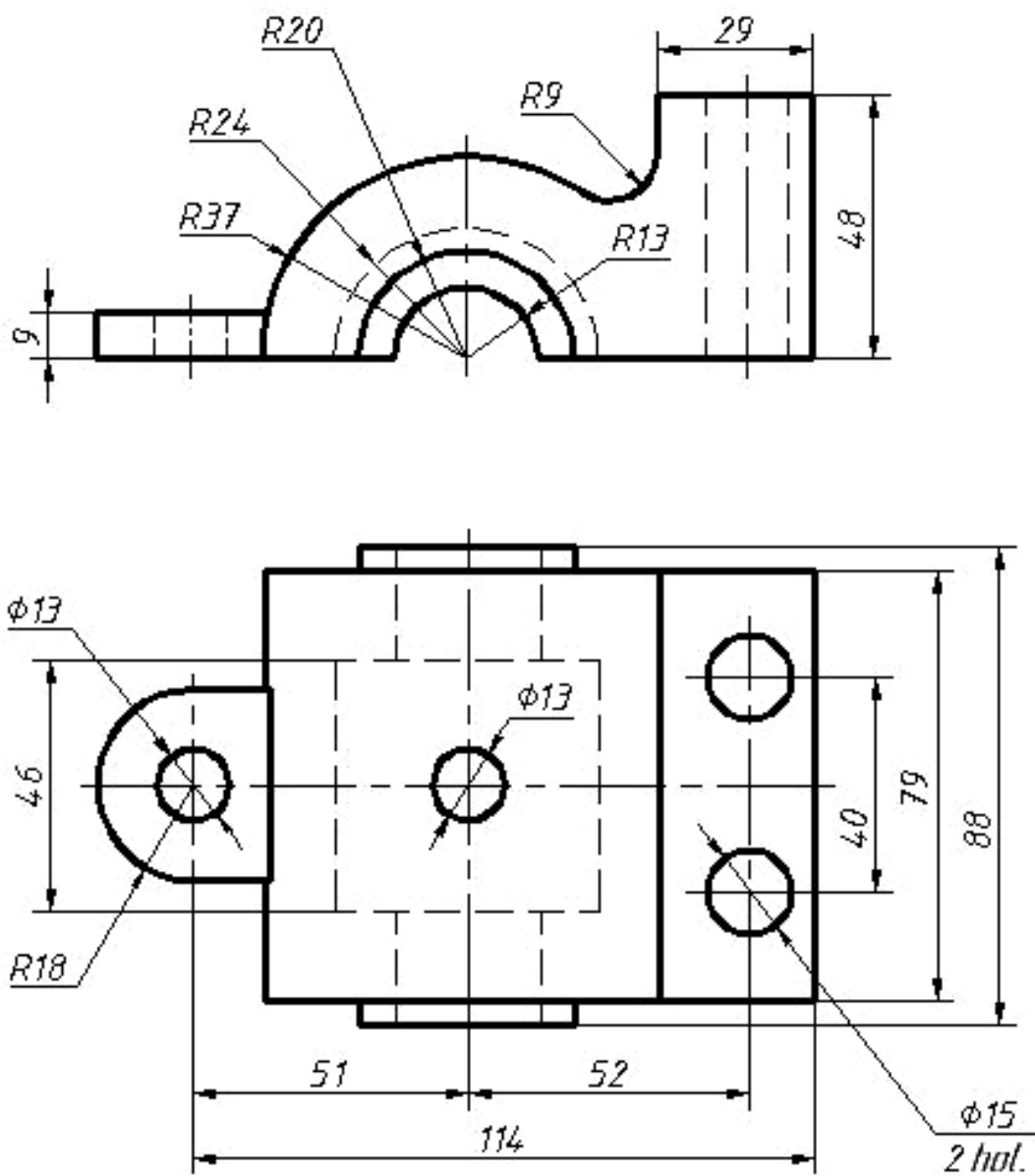


Variant 9

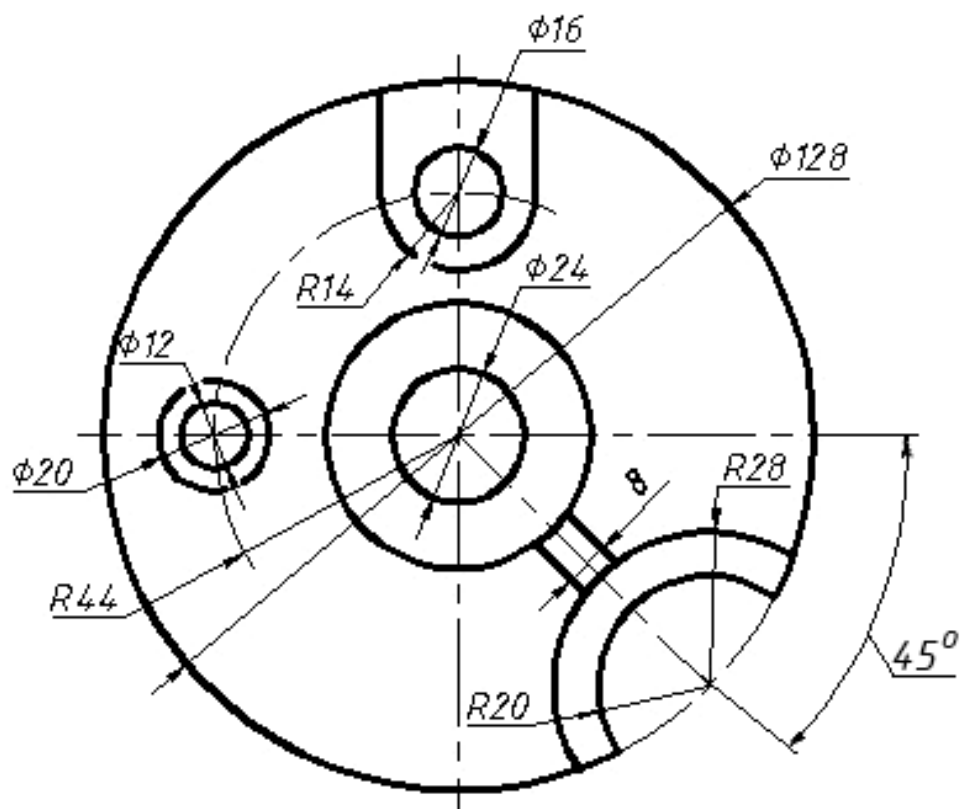
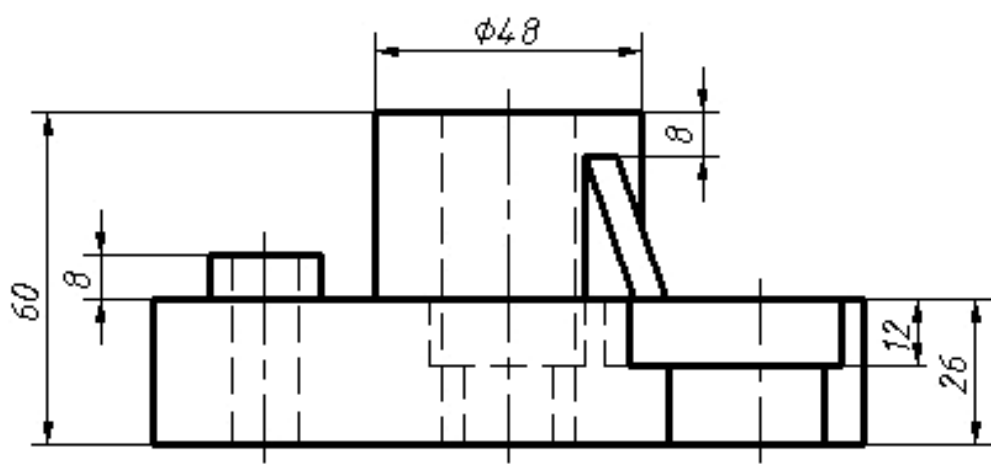


Variant 10

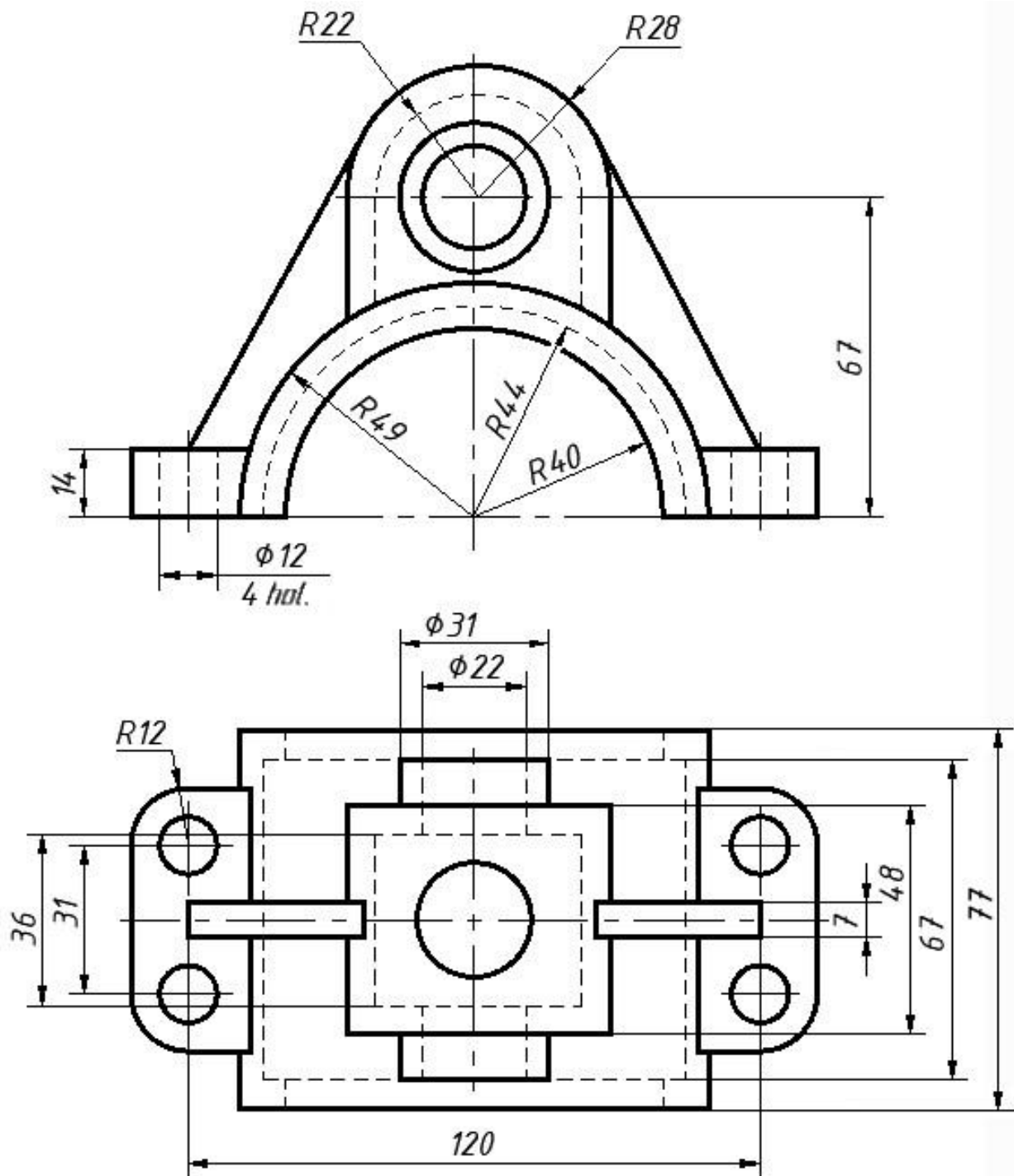




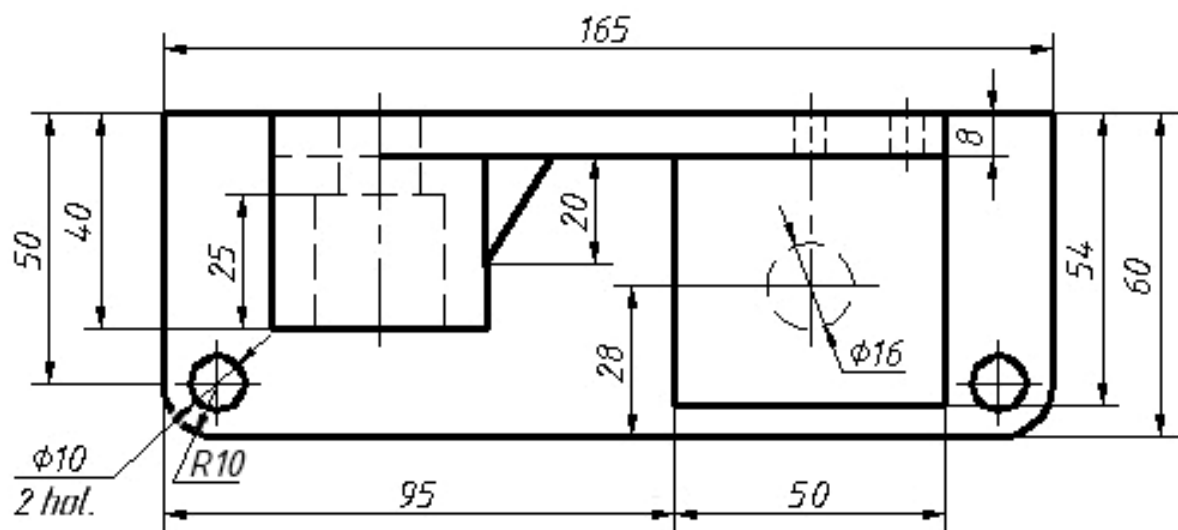
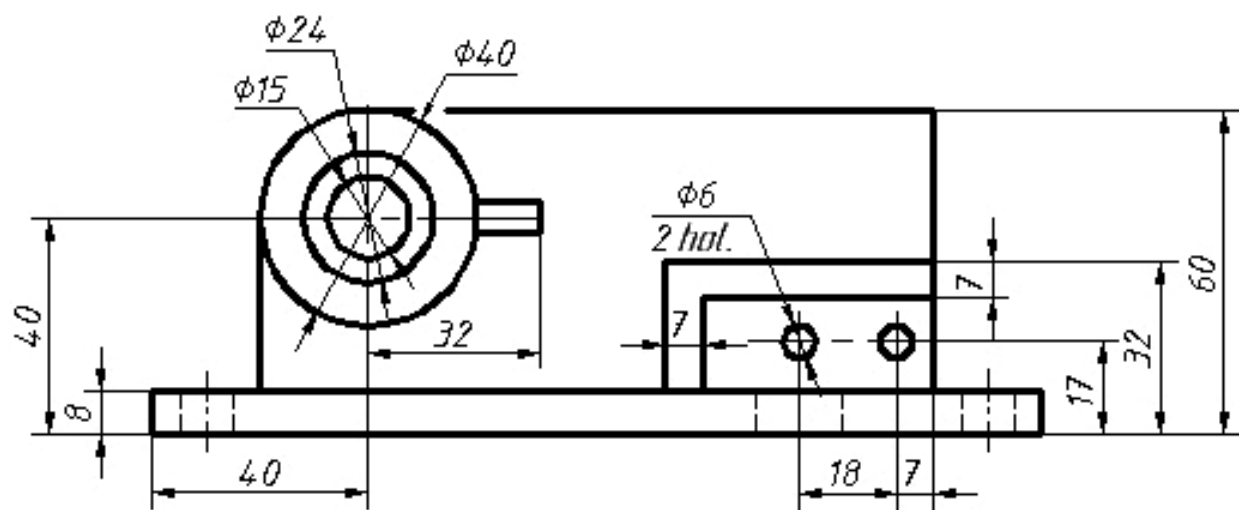
Variant 12



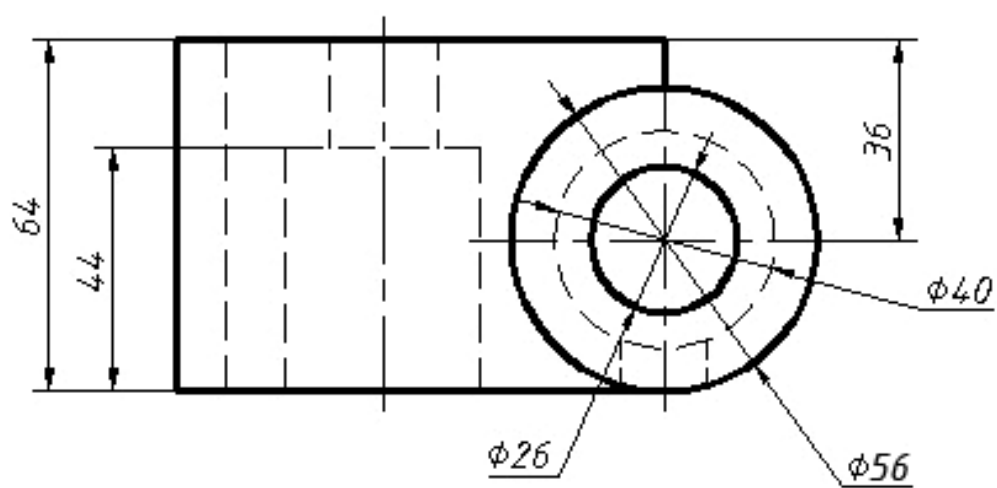
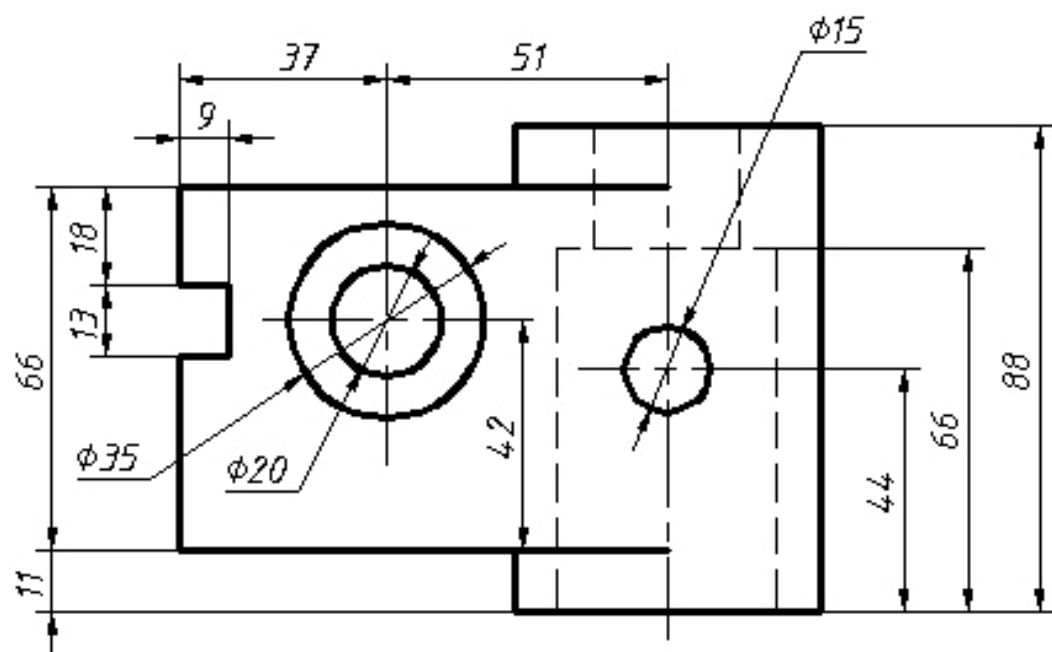
Variant 13



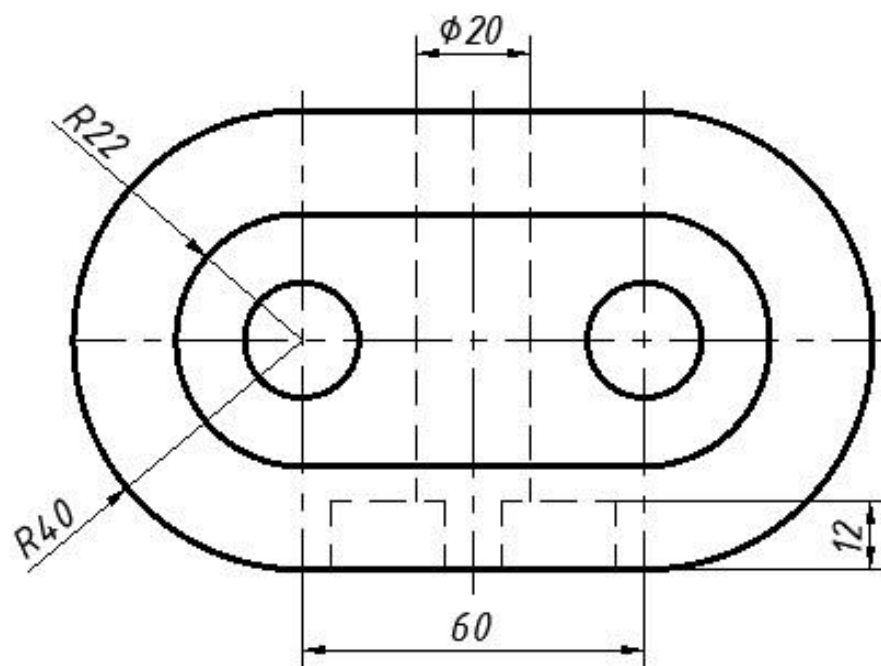
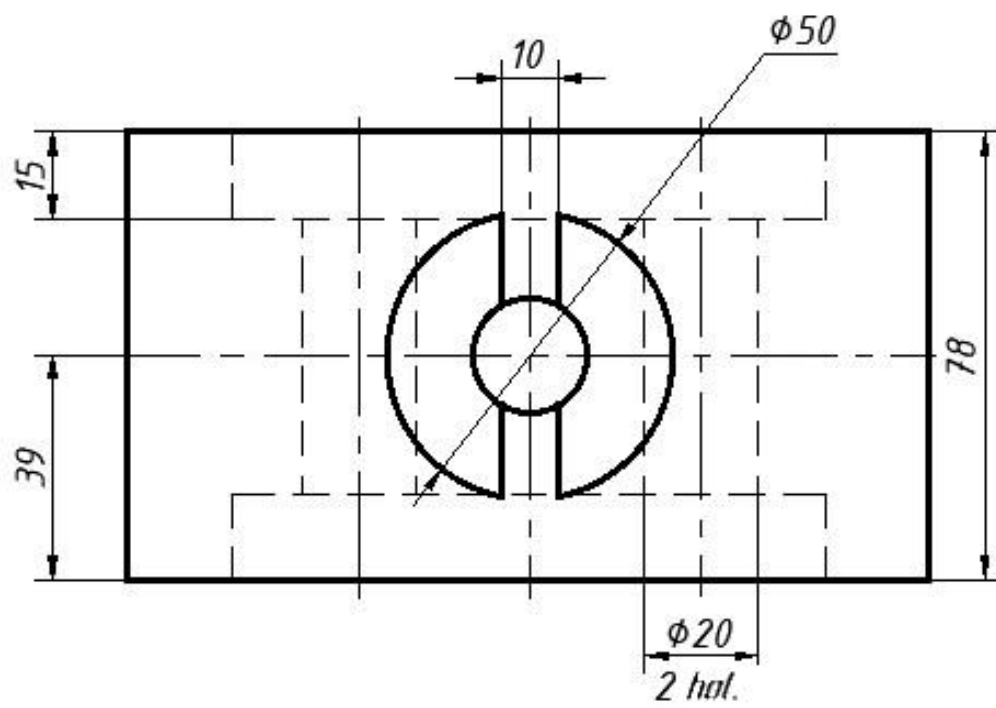
Variant 14



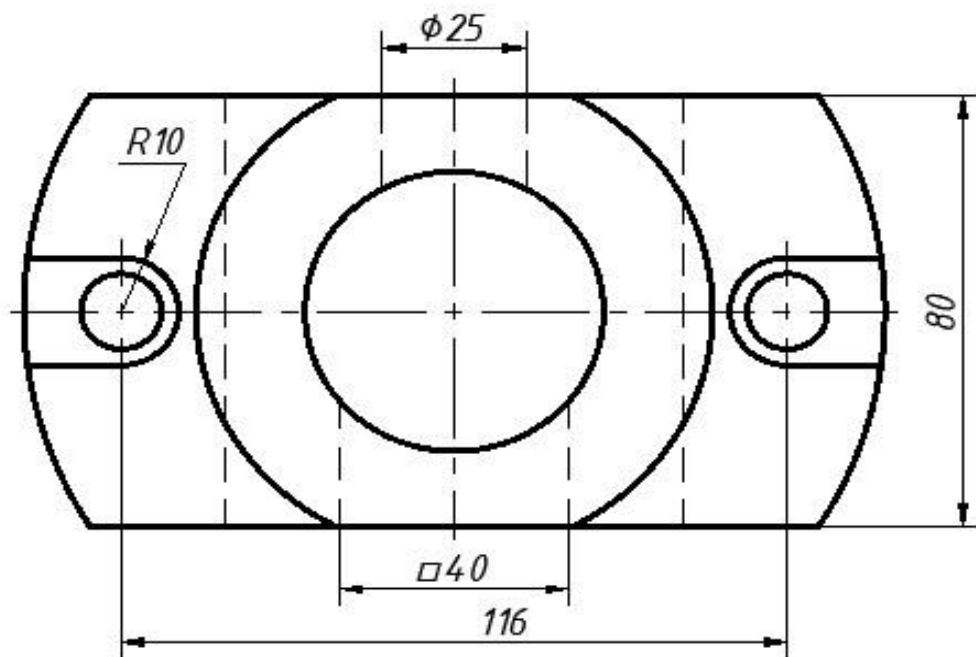
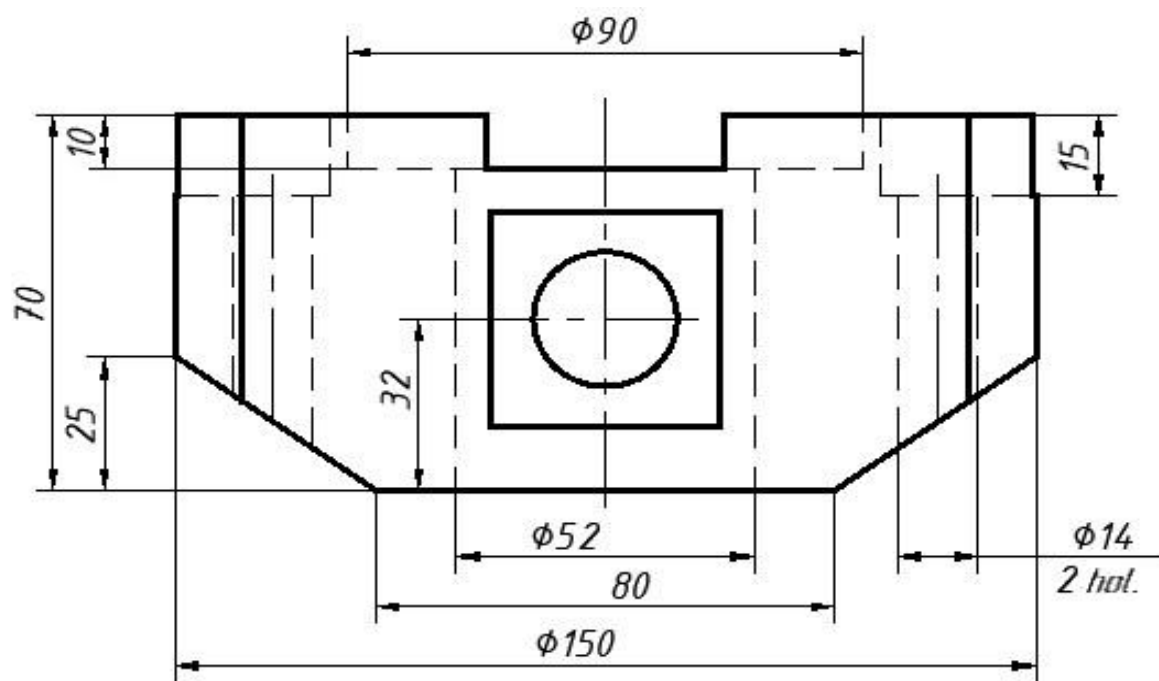
Variant 15



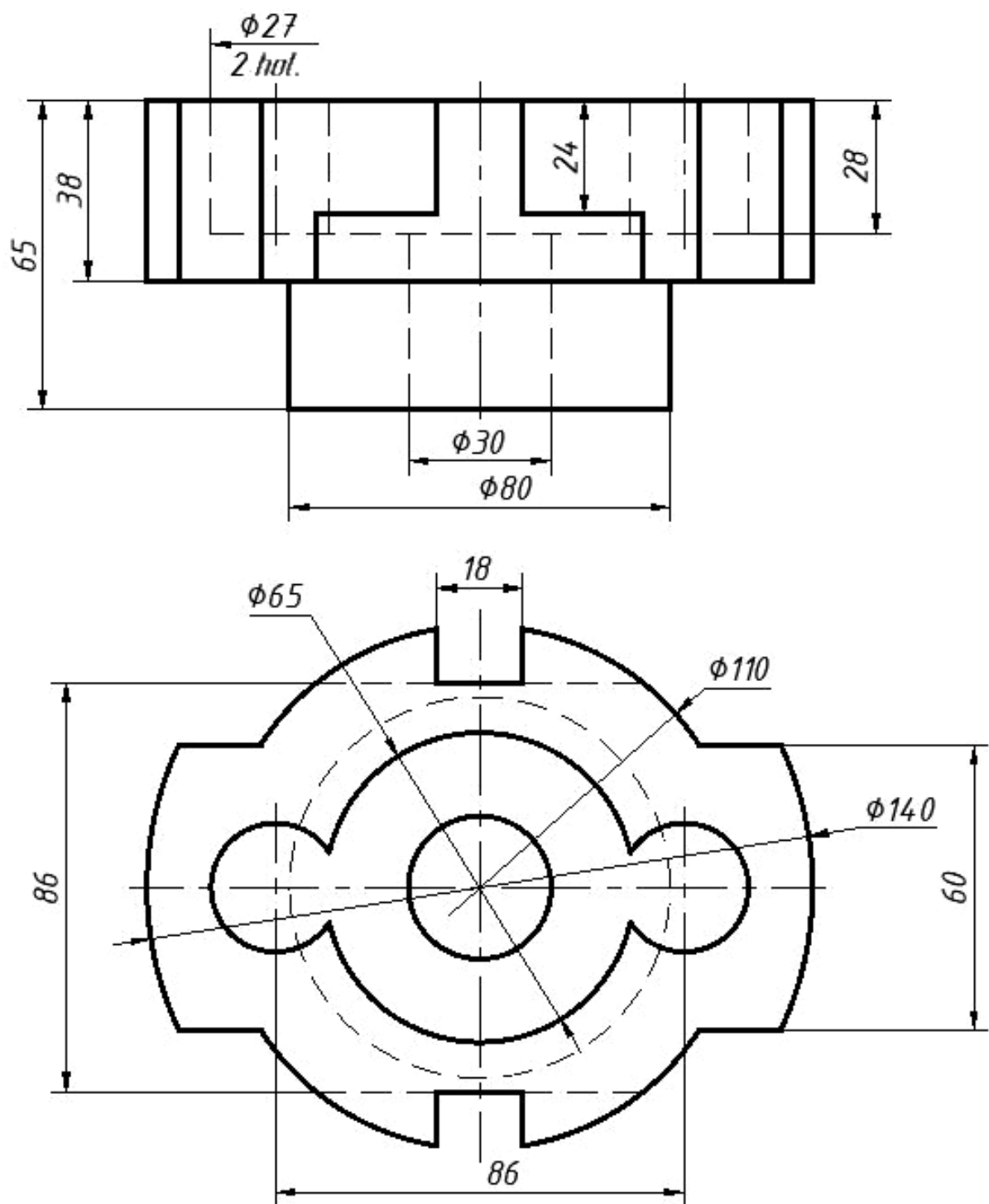
Variant 16



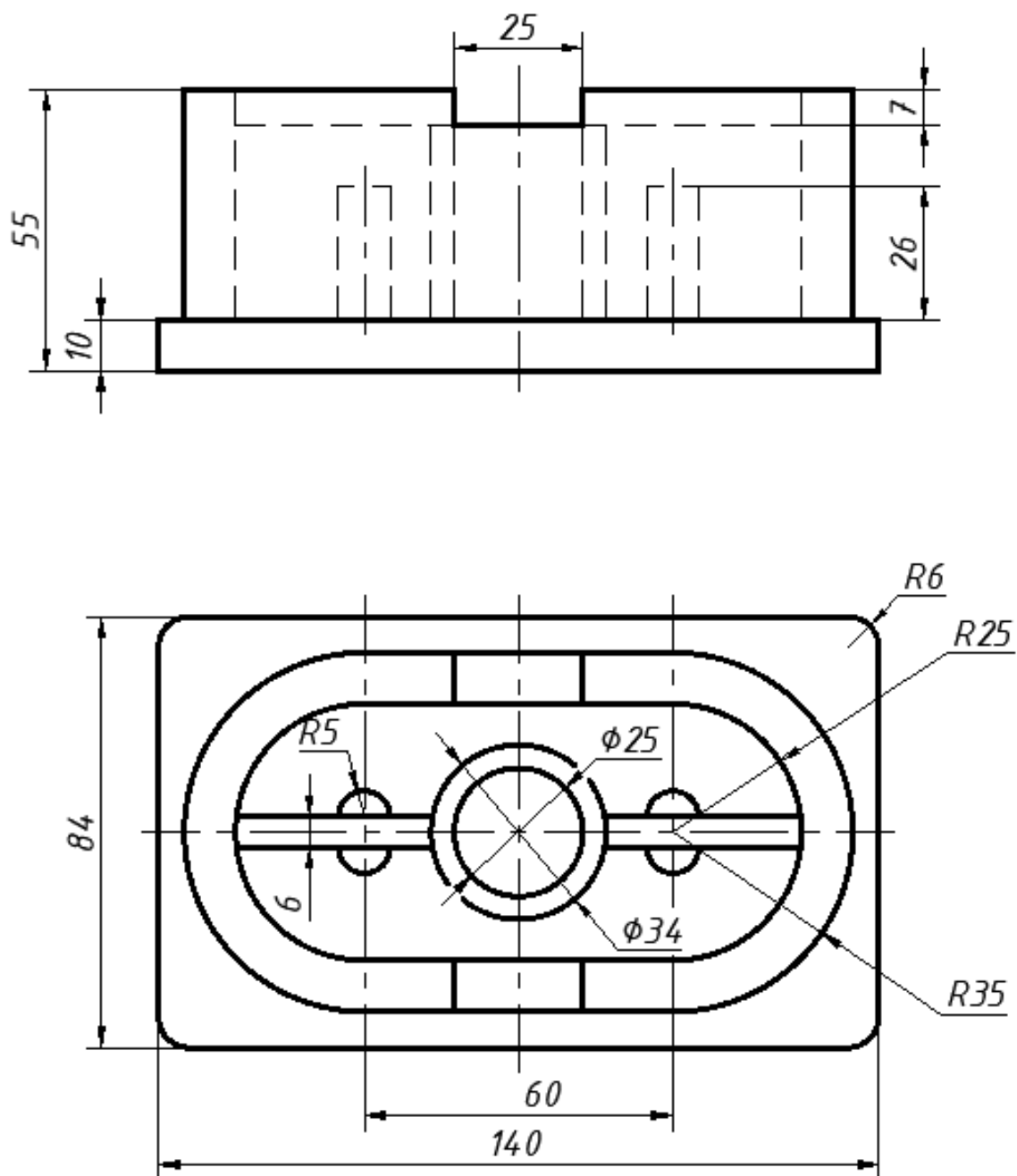
Variant 17



Variant 18



Variant 19



Variant 20

Навчальне видання

Геннадій Вірченко
Ганна Шепель
Ольга Лебедєва
Петро Яблонський
Тетяна Надкернична

**Комп'ютерна графіка в AutoCAD:
автоматизація засобами
Visual Basic for Application**

Підручник

(Англійською мовою)

В авторській редакції

Технічне редагування *Марії Скрінік*

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Свідоцтво про державну реєстрацію: серія ДК № 5354 від 25.05.2017 р.
просп. Берестейський, 37, м. Київ, 03056
<https://kpi.en>

Реєстр. № П 25/26-025. Формат 60×84^{1/8}. Гарнітура Times.
1 файл (13,2 Мб). Обл.-вид. арк. 23,35. Поз. 26-1-1-004.

Видавництво «Політехніка» КПІ ім. Ігоря Сікорського
вул. Політехнічна, 14, корп. 15
м. Київ, 03056, Україна
тел. (044) 204-81-78