

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

**О. С. Неуймін
О. Ю. Мирончук**

ТЕХНОЛОГІЇ ОБ'ЄКТНО- ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ ДЛЯ РАДІОІНЖЕНЕРІВ ДОМАШНЯ КОНТРОЛЬНА РОБОТА

*Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського
як навчальний посібник для здобувачів ступеня бакалавра за освітніми програмами
«Радіотехнічні комп'ютеризовані системи», «Інформаційна та комунікаційна
радіоінженерія», «Інтелектуальні технології радіоелектронної техніки»
спеціальності 172 «Телекомунікації та радіотехніка»
172 «Електронні комунікації та радіотехніка»*

Київ
КПІ ім. Ігоря Сікорського
2023

Рецензент *Пільтай Степан Іванович*, канд., техн. наук, доц., доцент
кафедри радіоінженерії КПІ ім. Ігоря Сікорського

Відповідальний редактор *Жук Сергій Якович*, д-р техн. наук, професор

*Гриф надано Методичною радою КПІ ім. Ігоря Сікорського (протокол № 5 від 23.02.2023 р.)
за поданням Вченої ради радіотехнічного факультету (протокол № 01/2023 від 30.01.2023 р.)*

Електронне мережне навчальне видання

*Неуймін Олександр Станіславович, канд. техн. наук
Мирончук Олександр Юрійович, PhD*

ТЕХНОЛОГІЇ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ ДЛЯ РАДІОІНЖЕНЕРІВ ДОМАШНЯ КОНТРОЛЬНА РОБОТА

Технології об'єктно-орієнтованого програмування для радіоінженерів.
Домашня контрольна робота [Електронний ресурс] : навч. посіб. для студ.
спеціальності 172 «Електронні комунікації та радіотехніка» / О.С. Неуймін,
О.Ю. Мирончук; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл:
1 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2023. – 17 с.

Навчальний посібник містить завдання і рекомендації для виконання
домашньої контрольної роботи з дисципліни «Технології об'єктно-орієнтованого
програмування для радіоінженерів».

© О. С. Неуймін, О. Ю. Мирончук, 2023
© КПІ ім. Ігоря Сікорського, 2023

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ	4
ВСТУП	5
ДОМАШНЯ КОНТРОЛЬНА РОБОТА.....	6
ВАРІАНТИ ЗАВДАНЬ	13
РЕКОМЕНДАЦІЇ ДО ОФОРМЛЕННЯ РОБОТИ.....	15
СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ	16
ДОДАТОК А.....	17

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

ООП	об'єктно-орієнтоване програмування
ПЗ	програмне забезпечення
ПК	персональний комп'ютер
БД	база даних
GUI	graphical user interface
IDE	integrated development environment
UML	unified modeling language

ВСТУП

Навчальний посібник призначений для навчання здобувачів ступеня бакалавра за освітніми програмами «Радіотехнічні комп'ютеризовані системи», «Інформаційна та комунікаційна радіоінженерія», «Інтелектуальні технології радіоелектронної техніки» спеціальності 172 «Електронні комунікації та радіотехніка».

Посібник містить теоретичні відомості, завдання і рекомендації до оформлення та виконання домашньої контрольної роботи з дисципліни «Технології об'єктно-орієнтованого програмування для радіоінженерів».

Домашня контрольна робота виконується на персональному комп'ютері із використанням середовища для розробки програмного забезпечення. Виконання домашньої контрольної роботи спрямоване на підвищення професійних умінь і навичок студентів та їх оцінки у сфері розробки програмного забезпечення.

ДОМАШНЯ КОНТРОЛЬНА РОБОТА

Тема: Розробка програмного забезпечення з використання об'єктно-орієнтованого програмування на мові C++.

Мета роботи: закріпити отримані знання і здобути практичний досвід розробки програмного забезпечення з використанням технології об'єктно-орієнтованого програмування (ООП) шляхом написання програмного коду згідно поставленого завдання.

Завдання:

Створити програмне забезпечення (ПЗ) з використанням ООП на мові C++ відповідно до варіанту задачі з можливістю логування важливих моментів роботи програми. Програма має містити **мінімум два класи** написаних власне студентом для реалізації поставленої задачі. Для опису взаємозв'язків між класами студент має створити Unified Modeling Language (UML) діаграму. Кінцевий результат проєкту необхідно завантажити на власний репозиторій в GitHub, при цьому розробку ПЗ слід вести в GitHub з самого почку створення проєкту.

Для створення програмного забезпечення рекомендується використання **середовища розробки (IDE)** Qt Creator [1, 2], однак студент може використовувати будь-яке інше середовище розробки, наприклад, Microsoft Visual Studio, Clion тощо. IDE Qt Creator дозволяє прискорити та спростити створення застосунків з графічним інтерфейсом на основі бібліотеки Qt (рис.1).

IDE Включає в себе:

- редактор коду C++;
- інтегрований дизайнер елементів GUI;
- елементи управління проєктом і збіркою;

- візуальна програма відлагодження коду;
- інструменти швидкої навігації по коду.

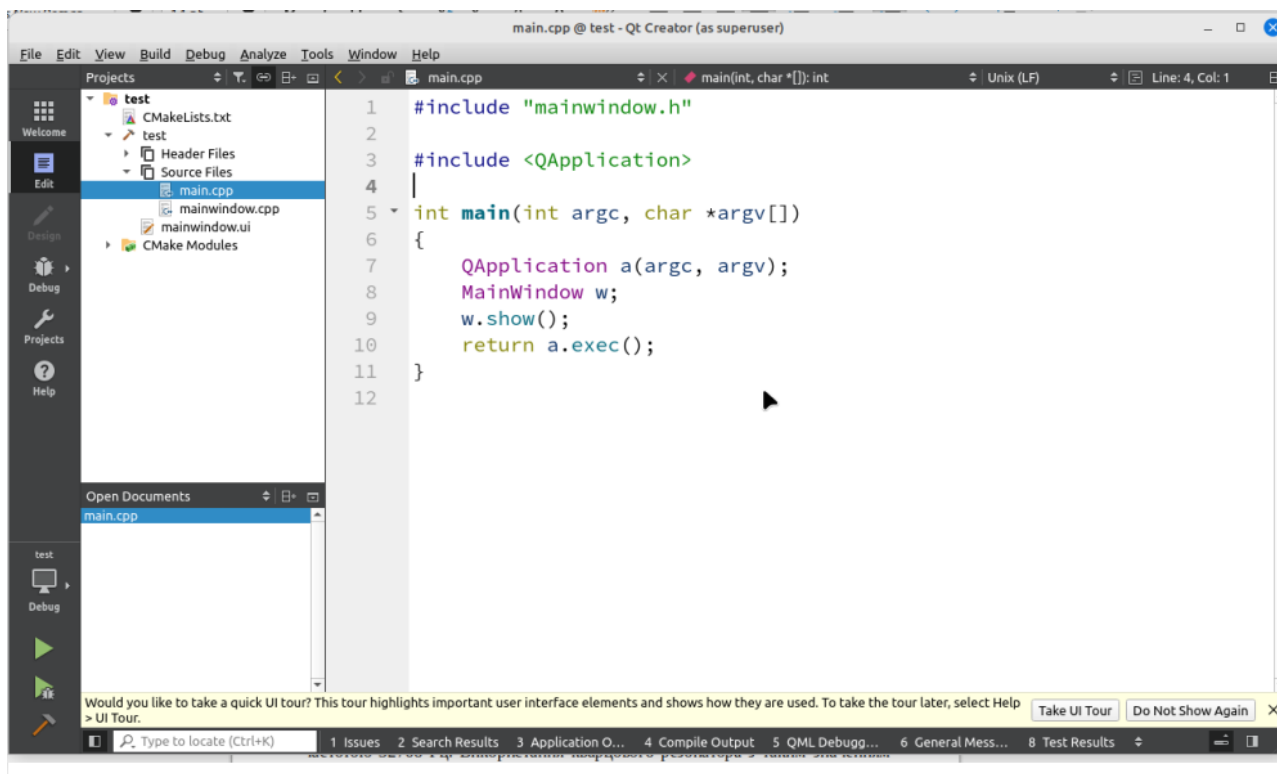


Рисунок 1. Приклад інтерфейсу середовища розробки Qt Creator

UML діаграма - це уніфікована мова моделювання, що використовується у парадигмі ООП для визначення, візуалізації, проєктування й документування програмних систем. UML є невід'ємною частиною уніфікованого процесу розробки ПЗ, і дозволяє розробникам з легкістю спілкуватися між собою про ПЗ.

За допомогою графічної нотації UML (рис.2) можна візуалізувати систему, об'єднати всі компоненти в єдину структуру, уточнювати і покращувати ПЗ у процесі розробки.

Існують наступні відношення між класами:

- Асоціація - взаємодія двох класів – абстрактний зв'язок на ранньому етапі аналізу/проєктування.
- Нестроге агрегування (включення) - один клас містить об'єкти іншого класу (has a). Приклад, спортзал містить снаряди або школа містить учнів.

Реалізується через посилання чи вказівник на деякий клас. Поінформованість - "Зі смертю господаря, раб не вмирає, і навпаки".

- Композиція - суворі агрегація - всередині об'єкта класу є об'єкт іншого класу. Наприклад, трикутник складається з 3-х точок, або Університет - Факультет. Можна реалізувати через посилання, але час життя об'єктів має бути однаковим.
- Наслідування (узагальнення) показує, що похідний клас наслідує атрибути та операції базового класу. Цей взаємозв'язок найлегше описується фразою «А — це Б» (ссавці — це тварини). Наслідування також відоме як "is a" взаємозв'язок (або відношення "це"). Наслідування встановлює сильніші зв'язки між класами, ніж агрегування (приведення між об'єктами, доступ до protected членів). Якщо наслідування можна легко замінити на агрегування, це потрібно зробити.
- Залежність - один клас (залежний) використовує інший клас (незалежний). Реалізується як метод класу, який отримує об'єкт за посиланням.

Такий набір фігур і стрілок значно спрощує вирішення складних завдань в програмуванні, допомагає знайти оптимальні рішення, розробити технічну документацію. UML діаграми легко читаються і необхідні для документування будь-якої нетривіальної системи.

Для побудови UML діаграми можна використовувати різні програмні продукти, наприклад, Microsoft Office або Visio, Google Drawings, Diagrams.net тощо (рис.3).

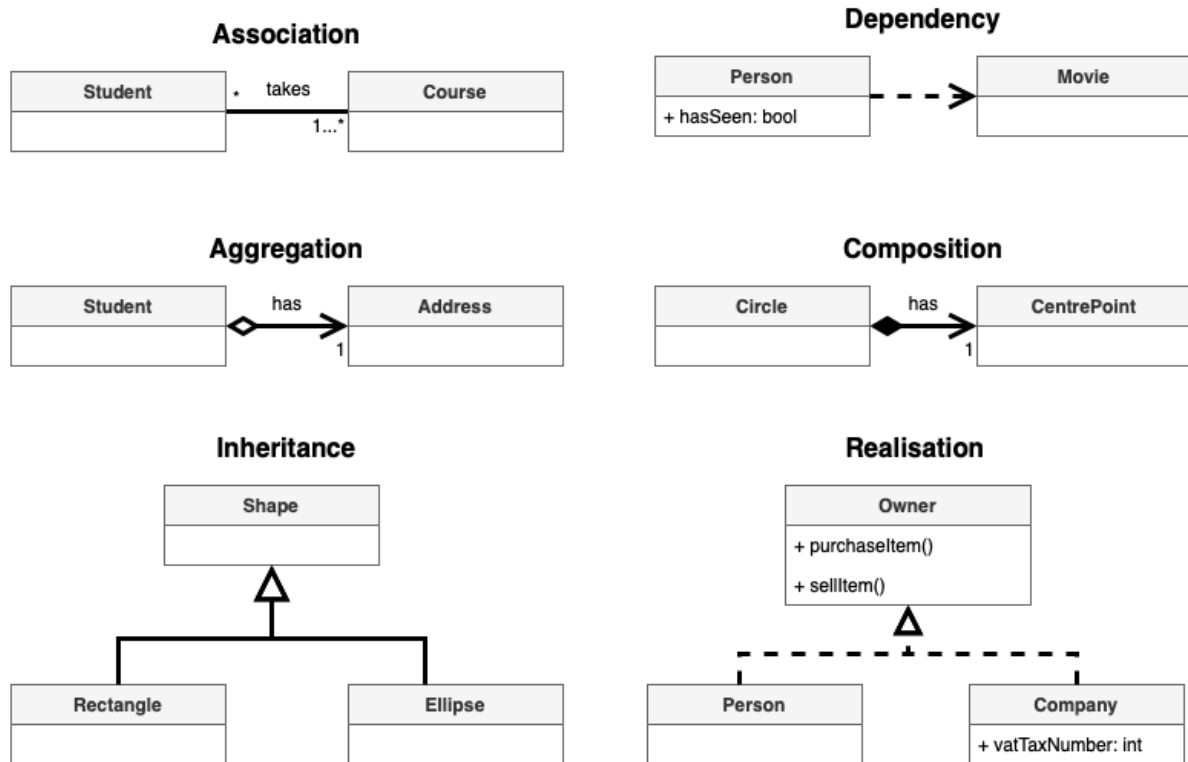


Рисунок 2. Графічна нотація UML

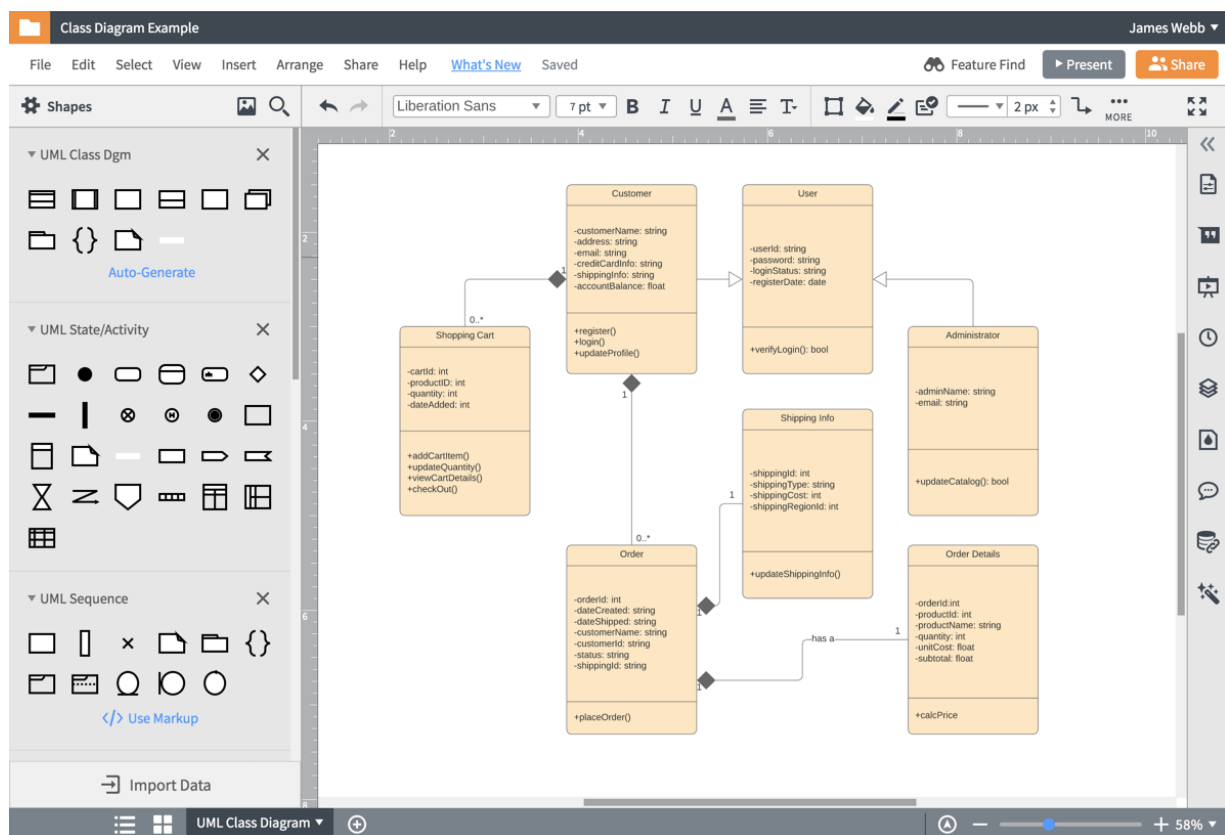


Рисунок 3. Приклад UML діаграми в Lucidchart

GitHub є одним із сервісів на основі Git. Git – це найпопулярніша та безкоштовна система для керування та контролю версіями, в якій зберігається код та історія його змін. Для легшого розуміння, що таке GitHub, можна уявити що це соціальна мережа для розробників, де вони переглядають код один одного, допомагають в розробці, залишають коментарі тощо.

GitHub дозволяє:

- зберігати код;
- використовувати інструменти для спільної роботи;
- оцінювати роботи інших розробників;
- створювати приватні та публічні репозиторії.

Для вирішення задачі пов'язаної з GitHub студенту необхідно зареєструвати акаунт та створити репозиторій для розміщення свого проєкту на сайті <https://github.com/> (рис. 4) [3, 4].

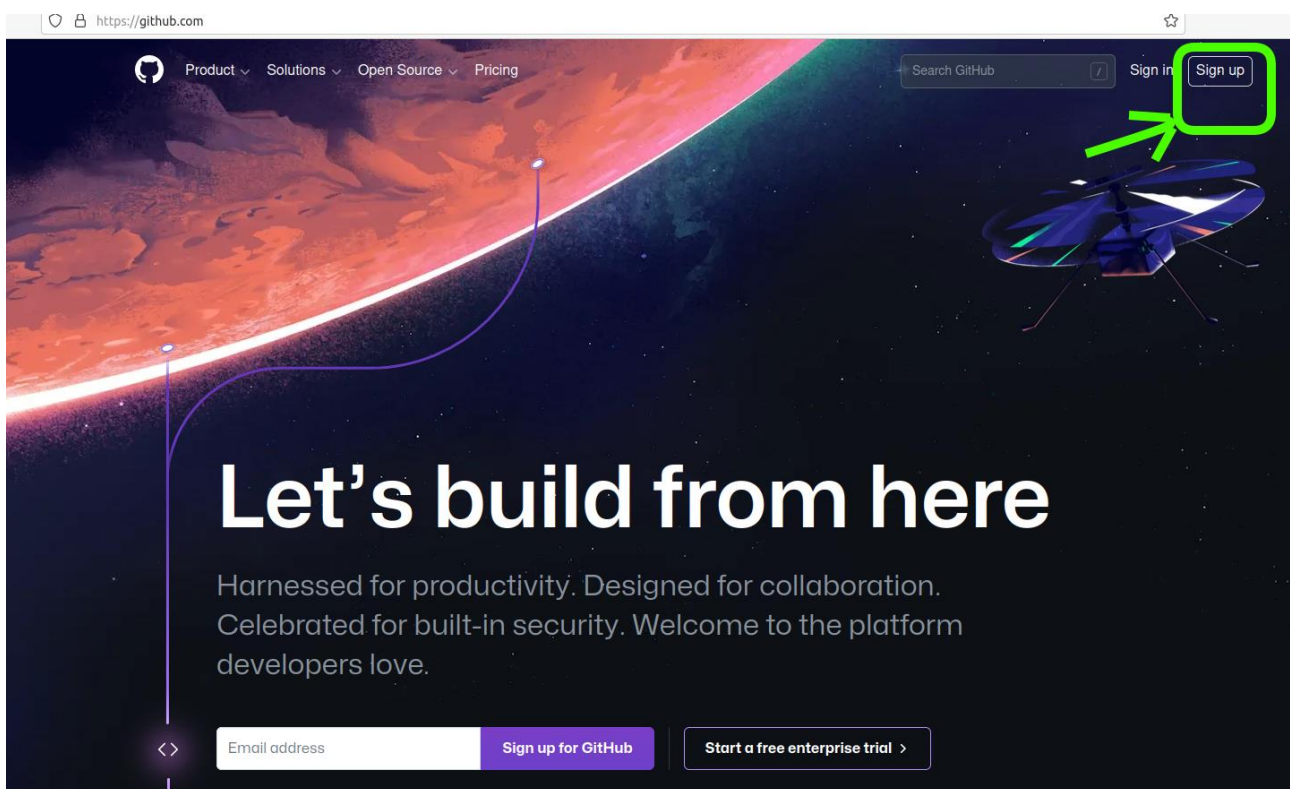


Рисунок 4. Сторінка GitHub

Додавання бібліотеки до проєкту

В роботі, окрім бібліотек Qt, необхідно буде додавати інші бібліотеки до свого проєкту. Цей процес залежить від типу та розташування бібліотеки. Студент може додати системну бібліотеку, свою власну бібліотеку або бібліотеку третьої сторони (3rd party library) (рис. 5). Бібліотека може бути розташована або в дереві збірки поточного проєкту, або в іншому дереві збірки.

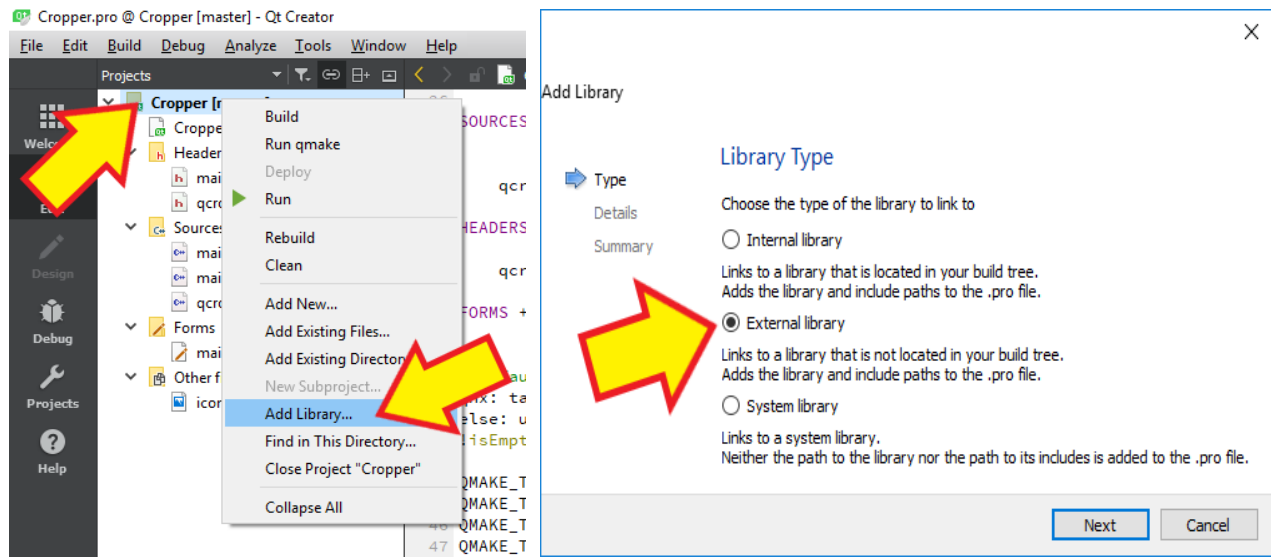


Рисунок 5. Додавання бібліотеки в проєкт на основі qmake

Для власних бібліотек і бібліотек третіх сторін потрібно вказати шляхи. Qt Creator намагається вгадати шлях включення для зовнішньої бібліотеки, але потрібно перевірити його та змінити, якщо необхідно. Qt Creator автоматично додає шлях включення для внутрішньої бібліотеки. Для бібліотеки виберіть цільову платформу та вкажіть статично чи динамічно пов'язана бібліотека (рис. 6). Для статично зв'язаної внутрішньої бібліотеки Qt Creator додає залежності (`target_link_libraries` при використанні CMake або `PRE_TARGETDEPS` при використанні qmake) у файл проєкту.

Під час використання CMake потрібно додати команду `target_link_libraries` до файлу `CMakeLists.txt`:

```
target_link_libraries(myapp PRIVATE mylib)
```

← Add Library

Type
Details
Summary

External Library

Specify the library to link to and the includes path

Library file: Browse...

Include path: Browse...

Platform: ☐ Linux ☐ Mac ☒ Windows

Linkage: Dynamic ☒ Dynamic ☐ Static

Mac: Library ☒ Library ☐ Framework

Windows: ☐ Library inside "debug" or "release" subfolder ☒ Add "d" suffix for debug version ☐ Remove "d" suffix for release version

Next Cancel

Рисунок 6. Вказування шляху до бібліотеки

ВАРІАНТИ ЗАВДАНЬ

Створити класи, специфікації яких наведені нижче. Визначити конструктори і методи `set..()`, `get..()`, `toString()`. Задати критерій вибору даних і вивести ці дані на консоль або вікно GUI. У кожному класі, що володіє інформацією, має бути оголошено кілька конструкторів. Необхідно перевантажити оператор ‘«’ для виводу даних на консоль або вікно GUI. Має бути можливість записати дані всіх об’єктів у файл з шифруванням (студент самостійно обирає спосіб шифрування) [5] або базу даних (БД).

Додатково має бути написаний клас для логування важливих моментів роботи програми з виводом у файл.

Створення графічного інтерфейсу не є обов’язковим, але показує кращі уміння та навички студента.

№	Завдання
1	Student: id, Прізвище, Ім'я, Дата народження, телефон, Факультет, Курс, Група. Створити масив об'єктів. Вивести: а) список студентів заданого факультету; б) списки студентів для кожного факультету і курсу; с) список студентів, які народилися після заданого року; d) список студентів навчальної групи.
2	Customer: id, Прізвище, Ім'я, Номер кредитної картки (4 цифри), Номер банківського рахунку (5 цифр). Створити масив об'єктів. Вивести: а) список покупців в алфавітному порядку; б) список покупців, у яких номер кредитної картки знаходиться в заданому інтервалі.
3	Book: id, Назва, Автор, Видавництво, Рік видання, Кількість сторінок, Ціна, Тип папітурки. Створити масив об'єктів. Вивести: а) список книг заданого автора; б) список книг, випущених заданим видавництвом; с) список книг, випущених після заданого року.
4	Car: id, Марка, Модель, Рік випуску, Колір, Ціна, Реєстраційний номер. Створити масив об'єктів. Вивести: а) список автомобілів заданої марки; б) список автомобілів заданої моделі, які експлуатуються більше n років; с) список автомобілів заданого року випуску, ціна яких більше зазначеної.

5	Airline: Пункт призначення, Номер рейсу, Тип літака, Час вильоту, Дні тижня. Створити масив об'єктів. Вивести: а) список рейсів для заданого пункту призначення; б) список рейсів для заданого дня тижня; с) список рейсів для заданого дня тижня, час вильоту для яких більше заданого.
6	Визначити клас Чотирикутник на площині. Визначити площу і периметр чотирикутника. Створити масив/список об'єктів і підрахувати кількість чотирикутників різного типу (квадрат, прямокутник, ромб, довільний). Визначити для кожної групи найбільший і найменший за площею (периметру) об'єкт.
7	Визначити клас Вектор . Реалізувати методи для обчислення модуля вектора, скалярного добутку, додавання, віднімання, множення на константу. Оголосити масив об'єктів. Написати метод, який для заданої пари векторів буде визначати, чи є вони колінеарними або ортогональними.
8	Створіть клас комплексних чисел , описавши в ньому всі необхідні властивості, підбравши їм зрозумілі імена і правильні типи даних. Опишіть в класі конструктор, що дозволяє при створенні нового об'єкта явно задати всі його властивості. Якщо це необхідно, то перевірте допустимість їх значень в конструкторі. Створіть в класі метод, що обчислює модуль суми двох комплексних чисел. З використанням побудованого класу створіть два комплексних числа: $3i + 1$ і $2i - 1$. Обчисліть за допомогою методу і виведіть на екран модуль їх суми.
9	Мобільний зв'язок. Визначити ієрархію тарифів мобільного компанії. Створити список тарифів компанії. Підрахувати загальну чисельність клієнтів. Провести сортування тарифів на основі розміру абонентської плати. Знайти тариф в компанії, що відповідає заданому діапазону параметрів.
10	Авіакомпанія. Визначити ієрархію літаків. Створити авіакомпанію. Порахувати загальну місткість і вантажопідйомність. Провести сортування літаків компанії подалі польоту. Знайти літак в компанії, що відповідає заданому діапазону параметрів споживання пального.

РЕКОМЕНДАЦІЇ ДО ОФОРМЛЕННЯ РОБОТИ

Результатом виконання домашньої контрольної роботи є звіт і програмний код вирішеної задачі, написаний в середовищі розробки програмного забезпечення.

Оцінка за виконання домашньої контрольної роботи виставляється по результатах її захисту. Під час захисту оцінюється якість та оформлення звіту, правильність роботи ПЗ, структура коду та UML діаграма ПЗ, знання написаної програми, теоретичні знання ООП. Правильність роботи програми демонструється на ПК. За наявності значних помилок робота не приймається і повертається на доопрацювання. Якщо оформлення роботи має недоліки, то така робота не приймається, і повертається студенту на доопрацювання.

Звіт про виконання домашньої контрольної роботи повинен мати наступну структуру:

- титульний лист;
- зміст;
- умова задачі;
- опис розробленого ПЗ;
- UML діаграма;
- програмний код;
- посилання на репозиторій GitHub;
- висновки;
- список використаної літератури.

Зразок титульної сторінки звіту про виконання домашньої контрольної роботи наводиться в додатках. Програмний код повинен містити коментарі, які описують основні складові програми.

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Qt6 QML Book. A book about Qt6 : [Веб-сайт]. – Електронні дані. – Режим доступу: <https://www.qt.io/product/qt6/qml-book>, вільний – Назва з екрана.
2. Application Development with Qt Creator: [Електронний ресурс] / Ray Rischpater //Birmingham - Mumbai – Електронні дані. – Режим доступу: <https://www.nanorobotdesign.com/tmp/c++/Qt.pdf> , вільний – Назва з екрана.
3. Створюємо репозиторій та заливаємо на GitHub [Електронний ресурс]. – Електронні дані. – Режим доступу: <https://pplug-community.gitbook.io/pplug-c-qt-roadmap-book/naivazhlivishii-rozdil-u-cii-knizhci/bazova-robota-z-git-ta-stvorenniya-vlasnogo-repozitoriyu/stvoryuyemo-repozitorii-ta-zalivayemo-na-github> , вільний – Назва з екрана.
4. Що таке GitHub і як з ним працювати. [Електронний ресурс]. – Електронні дані. – Режим доступу: <https://training.qatestlab.com/blog/technical-articles/what-is-github-and-how-to-work/> вільний – Назва з екрана.
5. Crypto++® Library [Електронний ресурс]. – Електронні дані. – Режим доступу: <https://www.cryptopp.com/> вільний – Назва з екрана.
6. Adding Libraries to Projects [Електронний ресурс]. – Електронні дані. – Режим доступу: <https://doc.qt.io/qtcreator/creator-project-qmake-libraries.html> вільний – Назва з екрана.
7. Adding External Libraries to Qt Projects [Електронний ресурс]. – Електронні дані. – Режим доступу: <https://amin-ahmadi.com/2018/10/27/adding-external-libraries-to-qt-projects/> вільний – Назва з екрана.

ДОДАТОК А
ЗРАЗОК ТИТУЛЬНОГО ЛИСТА

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

ДОМАШНЯ КОНТРОЛЬНА РОБОТА

з дисципліни «Технології об'єктно-орієнтованого програмування для
радіоінженерів»

Виконав (ла):
студент (ка) РТФ
групи _____
ПІБ

Київ – 20__