Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту

A blue and white logo

Description automatically generated

**Звіт**

**Звіт**

**про виконання лабораторних та практичних робіт блоку № 1**

На тему: «Розробка, програмування та код. Середовища для розробки.»

***з дисципліни:*** «Мови та парадигми програмування»

до:

Практичних Робіт № 1

**Виконав:**

Студент групи ШІ-14

Гембара Юрій Зіновійович

# **Тема роботи:**

Вступ до Розробки: Налаштування та Використання Середовища

# **Мета роботи:**

Налаштувати та ознайомитись з початковим середовищем розробки на c++

# **Теоретичні відомості:**

Теоретичні відомості з переліком важливих тем:

* Тема 1: Ознайомлення з Package Managers OS та командами
* Тема 2: Ознайомлення з Console Commands  в Linux подібному терміналі
* Тема 3: Встановлення та сконфігурування Visual Studio Code
* Тема 4: Встановлення Розширення для C++ на систему та Visual Studio Code
* Тема 5: Ознайомлення з Дебагером та Лінтером для C++
* Тема 6: Встановлення та ознайомлення з Git та командами
* Тема 7: Зареєстрування та ознайомлення з GitHub
* Тема 8: Ознайомлення з GitHub пул реквестами та Код ревю
* Тема 9: Зареєстрування та ознайомлення з Trello
* Тема 10: Зареєструвння та ознайомлення з Algotester
* Тема 11: Ознайомлення з FlowCharts та Draw.io
* Тема 12: Ознайомлення з Word та створення Звітів на Практичні та Лабораторні
* Тема 13: Запуск програмного коду C++ в  робочому середовищі
* Тема 14: Виконання теоретичний план по ознайомленню з інструментами

Індивідуальний план опрацювання теорії:

* Завдання №1: Ознайомлення з Package Managers OS та командами.
  + Джерела Інформації
    - Стаття: https://www.msys2.org/docs/package-management/
  + Що опрацьовано:
    - Система керування пакетами — набір системного програмного забезпечення, що дозволяє керувати процесом установки програмного забезпечення. Я використовував такий Package Manager як MSYS2. Він використовує порт pacman з Лінукса для керування пакетами та їх створення. У ньому я сконфігурував VS code використовуючи декілька команд, напириклад:

pacman -S --needed base-devel mingw-w64-ucrt-x86\_64-toolchain

* + Статус: Ознайомлений
  + Початок опрацювання теми: 01.10.23
  + Звершення опрацювання теми: 01.10.23
* Завдання №2: Ознайомлення з Console Commands  в Linux подібному терміналі
  + Джерела Інформації:
    - Статті:

https://en.wikipedia.org/wiki/Linux\_console

https://kinsta.com/blog/linux-commands/

* + Що опрацьовано:
    - Ознайомився з можливими командами у Терміналі Лінукса.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 01.10.23
  + Звершення опрацювання теми: 01.10.23
* Завдання №3: Встановлення та сконфігурування Visual Studio Code.
  + Джерела Інформації:
    - Статті:

https://code.visualstudio.com/docs/setup/windows

https://uk.wikipedia.org/wiki/Visual\_Studio\_Code

* + Що опрацьовано:
    - Встановив та сконфігурував середовище VS Code, додав необхідні пакети для розробки на C++. Запустив для тесту «Hello World!».

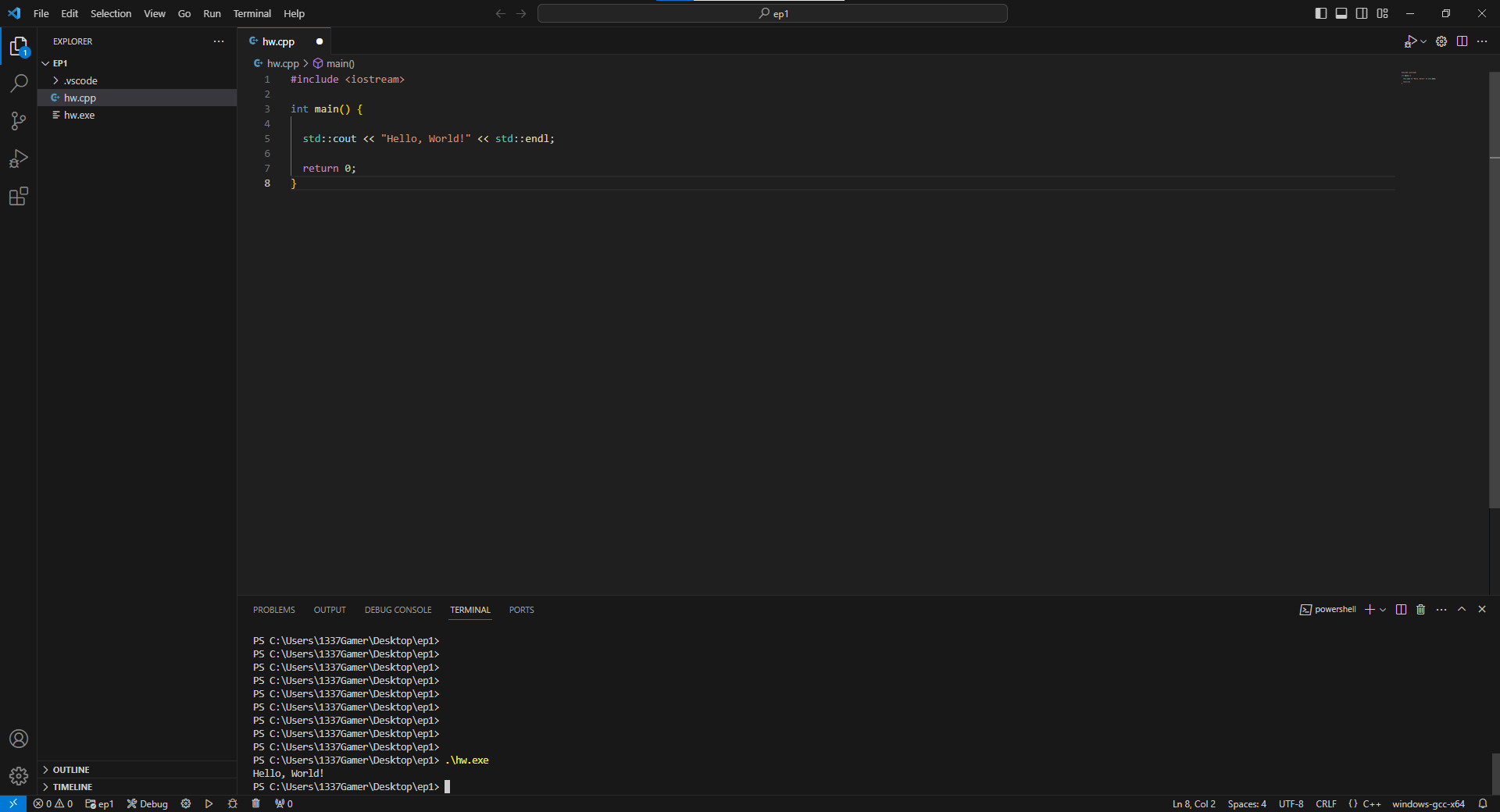


Рисунок 1 Hello World! у компіляторі

* + - Visual Studio Code, який також зазвичай називають VS Code — це редактор початкового коду, створений Microsoft для Windows, Linux і macOS. Функції включають підтримку налагодження, підсвічування синтаксису, інтелектуальне завершення коду, фрагменти, рефакторинг коду та вбудований Git. Користувачі можуть змінювати тему, комбінації клавіш, параметри та встановлювати розширення, які додають функціональність.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 03.10.23
  + Звершення опрацювання теми: 03.10.23
* Завдання №4 Встановлення Розширення для C++ на систему та Visual Studio Code.
  + Джерела Інформації:
  + Не знадобились
  + Що опрацьовано:

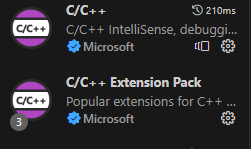


Рисунок 2 Встановлені розширення у компіляторі

* + - Встановив необхідний пакет у Visual Studio Code.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 03.10.23
  + Звершення опрацювання теми: 03.10.23
* Завдання №5 Ознайомлення з Дебагером та Лінтером для C++.
  + Джерела Інформації:
    - Стаття: https://cs.franklin.edu/~shaffstj/oldcs2/debugging.htm
  + Що опрацьовано:

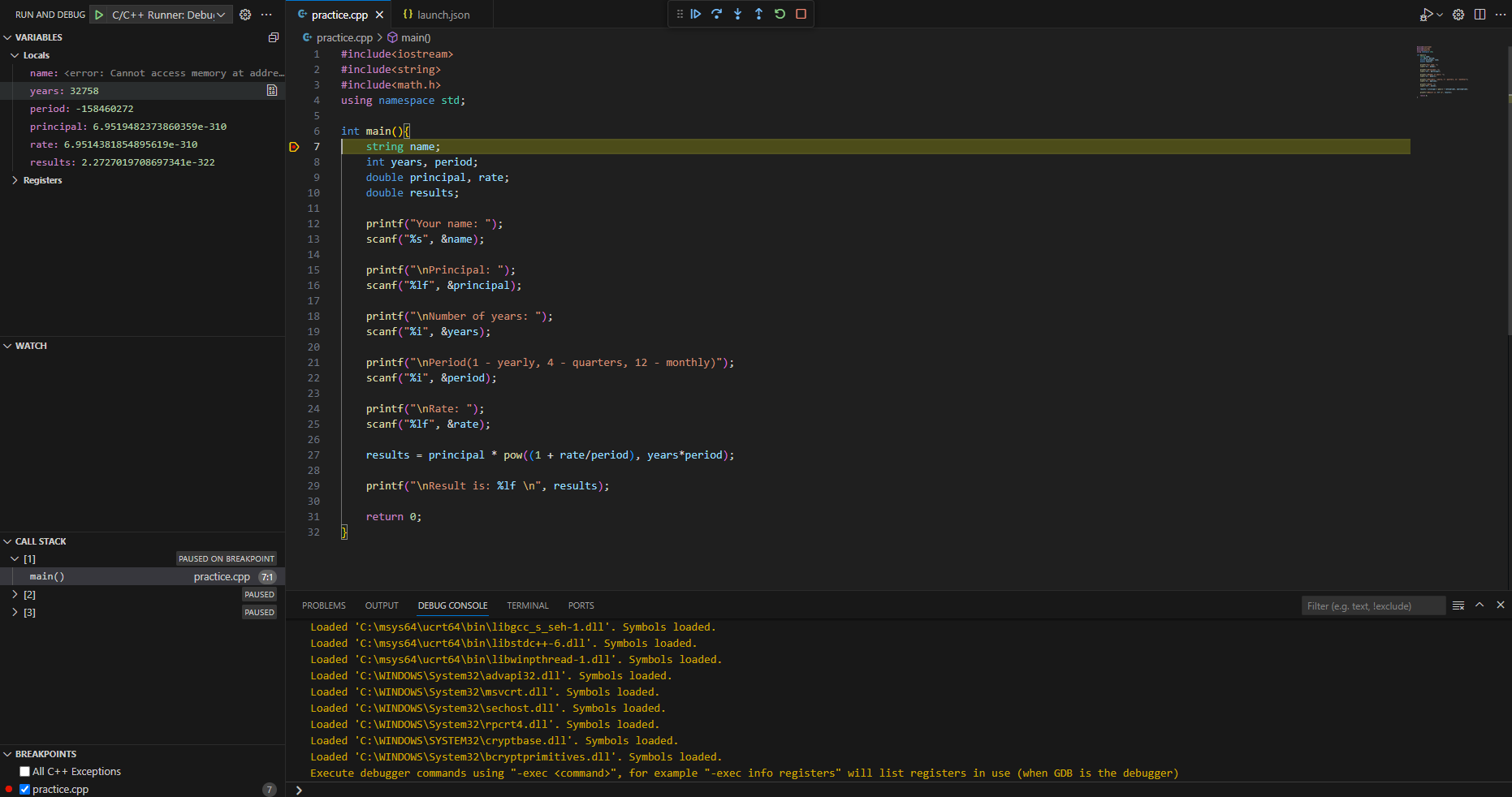


Рисунок 3 Робота дебагера

* + - Відлагоджувач (зневаджувач, англ. *debugger*, також зустр. англіцизм: деба́гер, українською означає відлагоджувати (програма що допомагає виявити помилки в програмному коді, в даному випадку) — комп'ютерна програма, яка використовується для тестування і виправлення вад інших програм. Як варіант, код для розгляду може бути запущено на емуляторі інструкцій, що дозволяє більший контроль зупинки процесу при заданих умовах, але, як правило, тоді виконання коду відбувається набагато повільніше, ніж якщо це робиться напряму на процесорі.
    - Лінтер допомагає зробити код чистішим і зрозумілішим для інших програмістів, які його читатимуть. Крім того, він допомагає шукати баги та стилістичні помилки. Пояснюємо, навіщо розробнику потрібний лінтер і як він допомагає економити час при спільній роботі над кодом.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 03.10.23
  + Звершення опрацювання теми: 03.10.23
* Завдання №6 Встановлення та ознайомлення з Git та командами
  + Джерела Інформації:
    - Відео: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=video&cd=&cad=rja&uact=8&ved=2ahUKEwjkmZm1qqCCAxVi\_rsIHSE7D00QtwJ6BAgWEAI&url=https%3A%2F%2Fwww.simplilearn.com%2Ftutorials%2Fgit-tutorial%2Fgit-installation-on-windows&usg=AOvVaw1FcyfXzyTvVAy7xJU8hb3F&opi=89978449
  + Що опрацьовано:
    - Встановлено Git, проведено ознайомлення з списком можливих команд для GitBash
    - Git — розподілена система керування версіями файлів та спільної роботи. Проєкт створив Лінус Торвальдс для керування розробкою ядра Linux, а сьогодні підтримується Джуніо Хамано (англ. *Junio C. Hamano*). Git є однією з найефективніших, надійних і високопродуктивних систем керування версіями, що надає гнучкі засоби нелінійної розробки, що базуються на відгалуженні і злитті гілок. Для забезпечення цілісності історії та стійкості до змін заднім числом використовуються криптографічні методи, також можлива прив'язка цифрових підписів розробників до тегів і комітів.

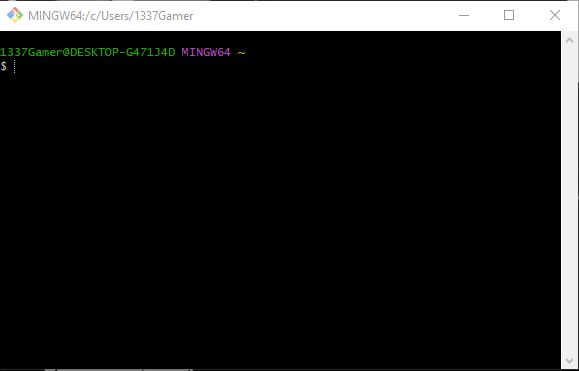


Рисунок 4 Запущена консоль MINGW64

* + - Статус: Ознайомлений
  + Початок опрацювання теми: 04.10.23
  + Звершення опрацювання теми: 04.10.23
* Завдання №7 Зареєстрування та ознайомлення з GitHub.
  + Джерела Інформації:
  + Не знадобились
  + Що опрацьовано:
    - Зареєстровано акаунт GitHub, проведено ознайомлення з можливістю цього сервісу
    - GitHub — один з найбільших вебсервісів для спільної розробки програмного забезпечення. Існують платні та безплатні тарифні плани користування. Базується на системі керування версіями Git і розроблений на Ruby on Rails і Erlang компанією GitHub, Inc (спочатку вона називалася Logical Awesome). Сервіс безплатний для проєктів із відкритим вихідним кодом і надає користувачам усі свої можливості (включаючи SSL), а для окремих проєктів пропонуються різні платні тарифні плани.

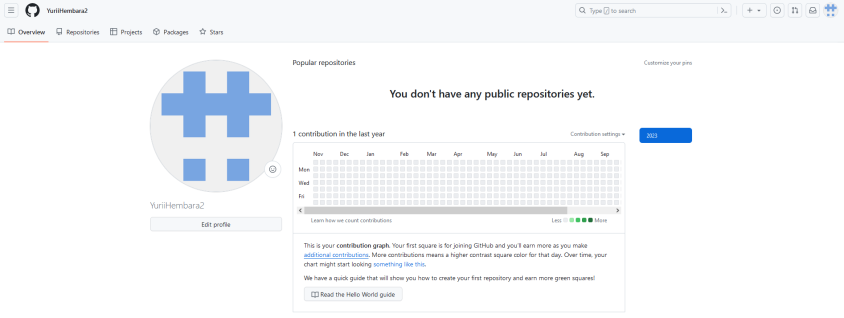


Рисунок 5 Профіль на GitHub

* + Статус: Ознайомлений
  + Початок опрацювання теми: 04.10.23
  + Звершення опрацювання теми: 04.10.23
* Завдання №8 Ознайомлення з GitHub пул реквестами та Код ревю .
  + Джерела Інформації:
    - Відео: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=video&cd=&cad=rja&uact=8&ved=2ahUKEwi104XTqqCCAxX\_\_7sIHd\_XDQoQtwJ6BAgJEAI&url=https%3A%2F%2Fwww.youtube.com%2Fwatch%3Fv%3DjRLGobWwA3Y&usg=AOvVaw28m2Ph6t-D\_AA5EPKf3Og4&opi=89978449
  + Що опрацьовано:
  + Pull request'и дозволяють вам розповісти іншим про ті зміни, які ви розмістили у своєму GitHub-репозиторії. Як тільки pull request відправлено, зацікавлені сторони розглядають ваші зміни, обговорюють можливі правки або навіть додають додаткові коміти, якщо потрібно.
  + Код-рев'ю - це процес перевірки коду, який дозволяє: виявити помилки, пропуски, вразливості та стилістичні недоліки (з точки зору проекту або прийнятих у команді правил). Приклад діалогу між автором та ревьюєром.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 07.10.23
  + Звершення опрацювання теми: 07.10.23
* Завдання №9 Зареєстрування та ознайомлення з Trello.
  + Джерела Інформації:
  + Не знадобилися
  + Що опрацьовано:
    - Зареєстровано акаунт Trello та проведено ознайомлення з його можливостями.
    - Trello – хмарна програма для керування проектами невеликих груп, розроблена Fog Creek Software. Trello використовує парадигму для управління проектами, відому як канбан, метод, який спочатку був популяризований Toyota в 1980-х для управління ланцюжками постачання.

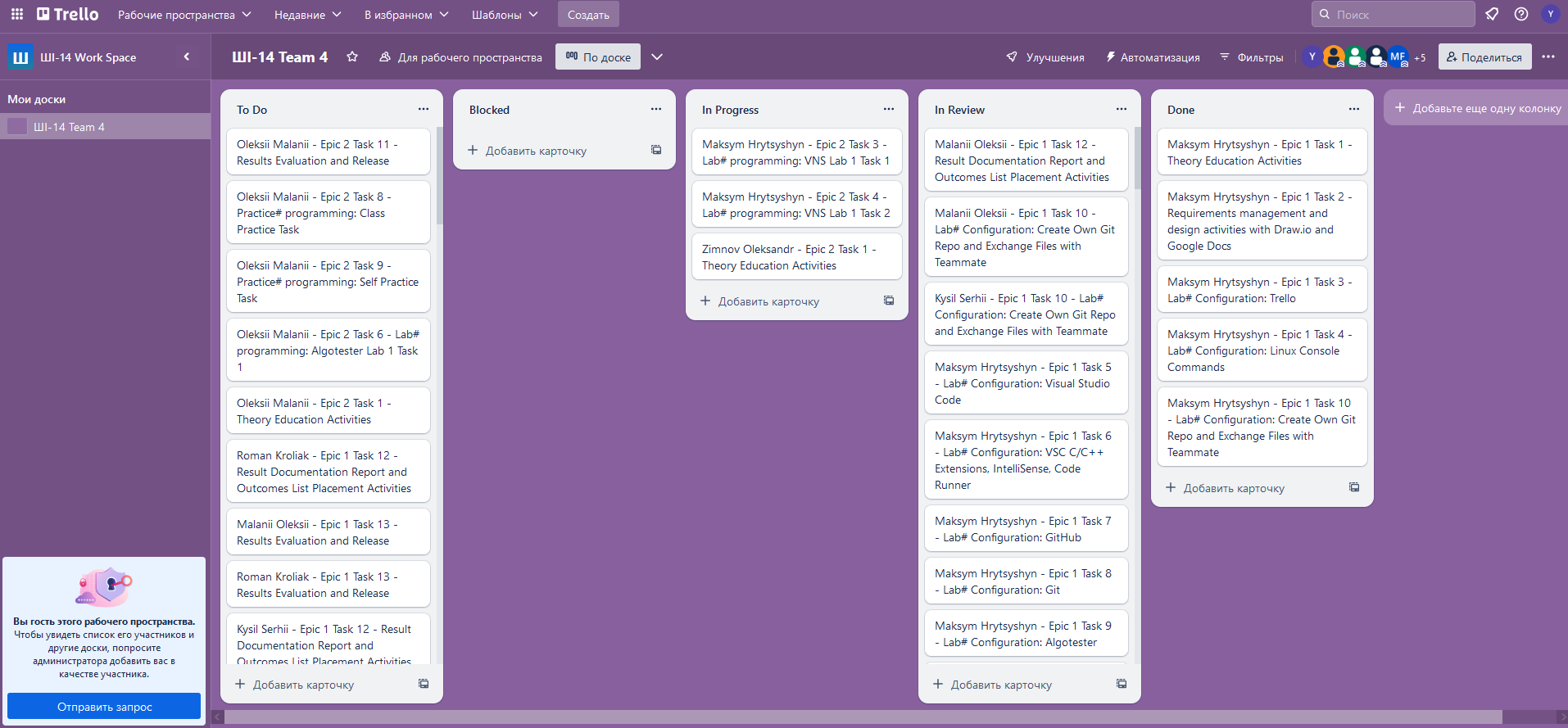


Рисунок 6 Дошка у Trello

* + Статус: Ознайомлений
  + Початок опрацювання теми: 08.10.23
  + Звершення опрацювання теми: 08.10.23
* Завдання №10 Зареєструвння та ознайомлення з Algotester.
  + Джерела Інформації:
  + Не знадобилися
  + Що опрацьовано:
    - Зареєстровано акаунт Algotester та проведено ознайомлення з його можливостями.
    - LNU Algotesters— перша в Україні школа мистецтва розв'язування алгоритмічно складних задач, переможець півфіналу світу у південно-східному європейському регіоні з алгоритмічного програмування серед студентських команд SEERC 2016.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 08.10.23
  + Звершення опрацювання теми: 08.10.23
* Завдання №11 Ознайомлення з FlowCharts та Draw.io.
  + Джерела Інформації:
  + Не знадобилися
  + Що опрацьовано:
  + Блок-схема (англ. *block scheme, flowchart, block diagram, flow diagram*; нім. *Block-schema*) — Представлення алгоритму розв'язування або аналізу задачі за допомогою геометричних елементів (блоків), які позначають операції, потік, дані тощо. Блок вхідних та вихідних даних прийнято позначати паралелограмом, блок обчислень (обробки) даних — прямокутником, блок прийняття рішень — ромбом, еліпсом — початок та кінець алгоритму.
  + Draw.io — це онлайн-програмне забезпечення для настільних комп’ютерів із відкритим кодом. Це програмне забезпечення для створення блок-схем і схем, розроблене відповідно до сучасних обов’язків і чутливості професіоналів. Крім того, ця програма може справити на користувачів гарне враження завдяки своєму інтуїтивно зрозумілому інтерфейсу, який дозволяє їм розміщувати свої дані в більш зручному вигляді. Це тому, що його інтерфейс містить параметри та інструменти, які легко доступні та зрозумілі користувачам будь-якого рівня. Крім того, можна очікувати, що програма для створення блок-схем може бути універсальною програмою. Тому що окрім того, що користувачі можуть використовувати онлайн і офлайн, що робить Draw.io безкоштовним інструментом, він також постачається з різними шаблонами та макетами для будь-яких художніх вимог, які можуть знадобитися користувачеві.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 09.10.23
  + Звершення опрацювання теми: 09.10.23
* Завдання №12 Ознайомлення з Word та створення Звітів на Практичні та Лабораторні.
  + Джерела Інформації:
  + Не знадобилися
  + Що опрацьовано:
    - Ознайомлений зі зразком виконання звіту для практичних/лабораторних робіт.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 11.10.23
  + Звершення опрацювання теми: 11.10.23
* Завдання №13 Запуск програмного коду C++ в  робочому середовищі.
  + Джерела Інформації:
  + Не знадобилися
  + Що опрацьовано:
    - Написано та запущено програму, що рахує депозити.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 12.10.23
  + Звершення опрацювання теми: 12.10.23
* Завдання №14 Виконання теоретичний план по ознайомленню з інструментами.
  + Джерела Інформації:
  + Не знадобилися
  + Що опрацьовано:
    - Виконано ознайомлення з інструментами середовища розробки на C++
  + Статус: Ознайомлений
  + Початок опрацювання теми: 12.10.23
  + Звершення опрацювання теми: 12.10.23

# **Виконання роботи:**

## **1. Опрацювання завдання та вимог до програм та середовища:**

Завдання №1 Програма, що рахує депозити

* Варіант завдання: один на всіх
* Деталі завдання: написати програму, яка б рахувала депозити користувачеві.
* Важливі деталі для врахування в імплементації програми: обов'язкова умова – використання printf та scanf

## **2. Дизайн та планована оцінка часу виконання завдань:**

Програма №1 Програма, що рахує депозити

* Блок-схема:



Рисунок 7 Блок схема до програми

* Планований час на реалізацію: 1 година
* Важливі деталі для врахування в імплементації: обов'язкова умова – використання printf та scanf

Програма №2 Самостійна практична робота (Hello World)

* Блок-схема:

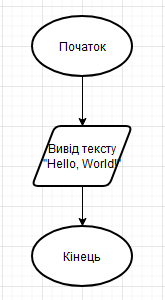


Рисунок 8 Блок схема до програми

* Планований час на реалізацію: 3 хвилини

## **3. Конфігурація середовища до виконання завдань:**

Для виконання завданнь мені потрібен був сконфігурований VS Code.

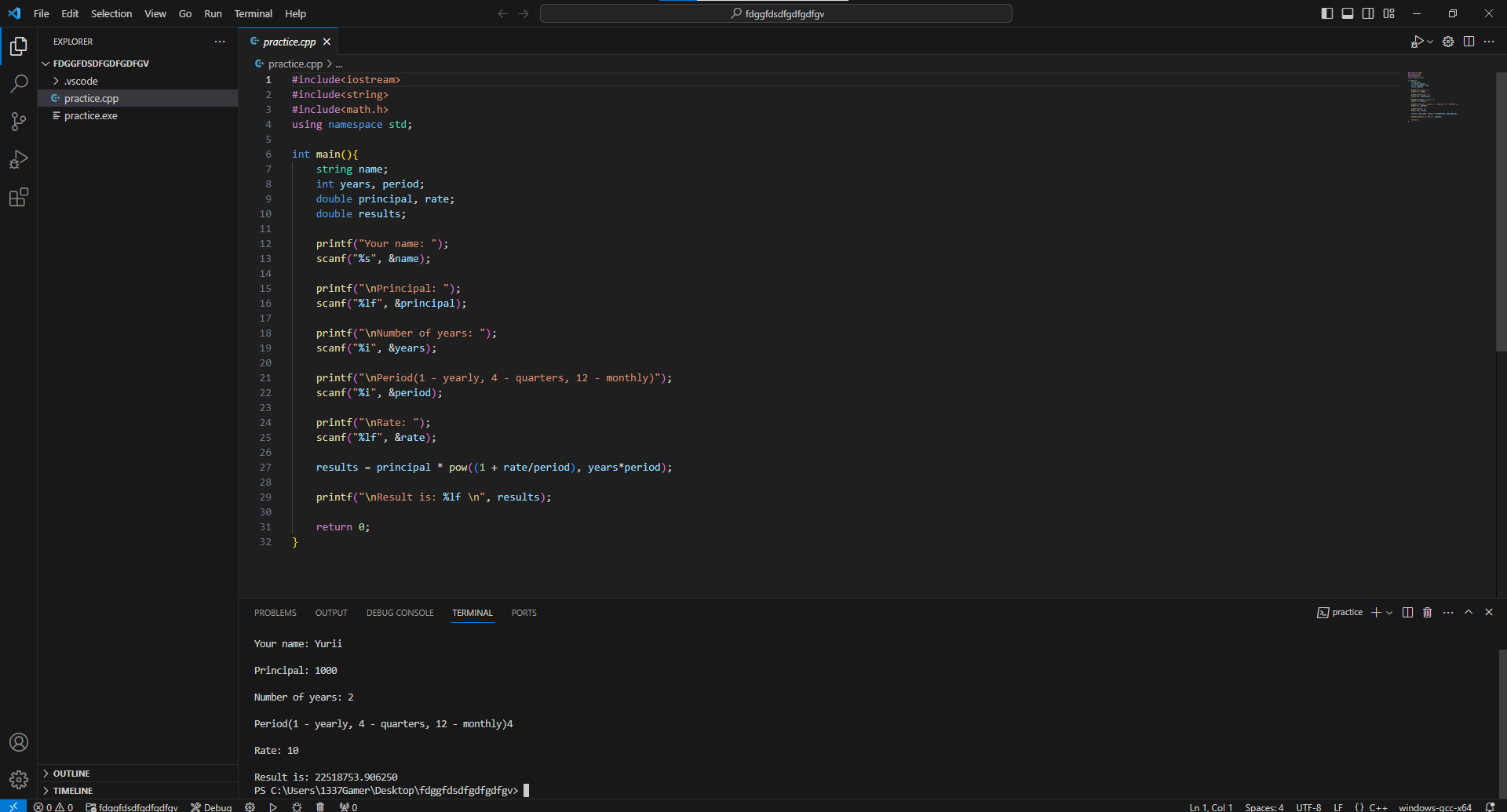


Рисунок 9 Робоча програма у VS Code

## **4. Код програм з посиланням на зовнішні ресурси:**

Завдання №1 Програма рахує депозит користувача використовуючи printf та scanf

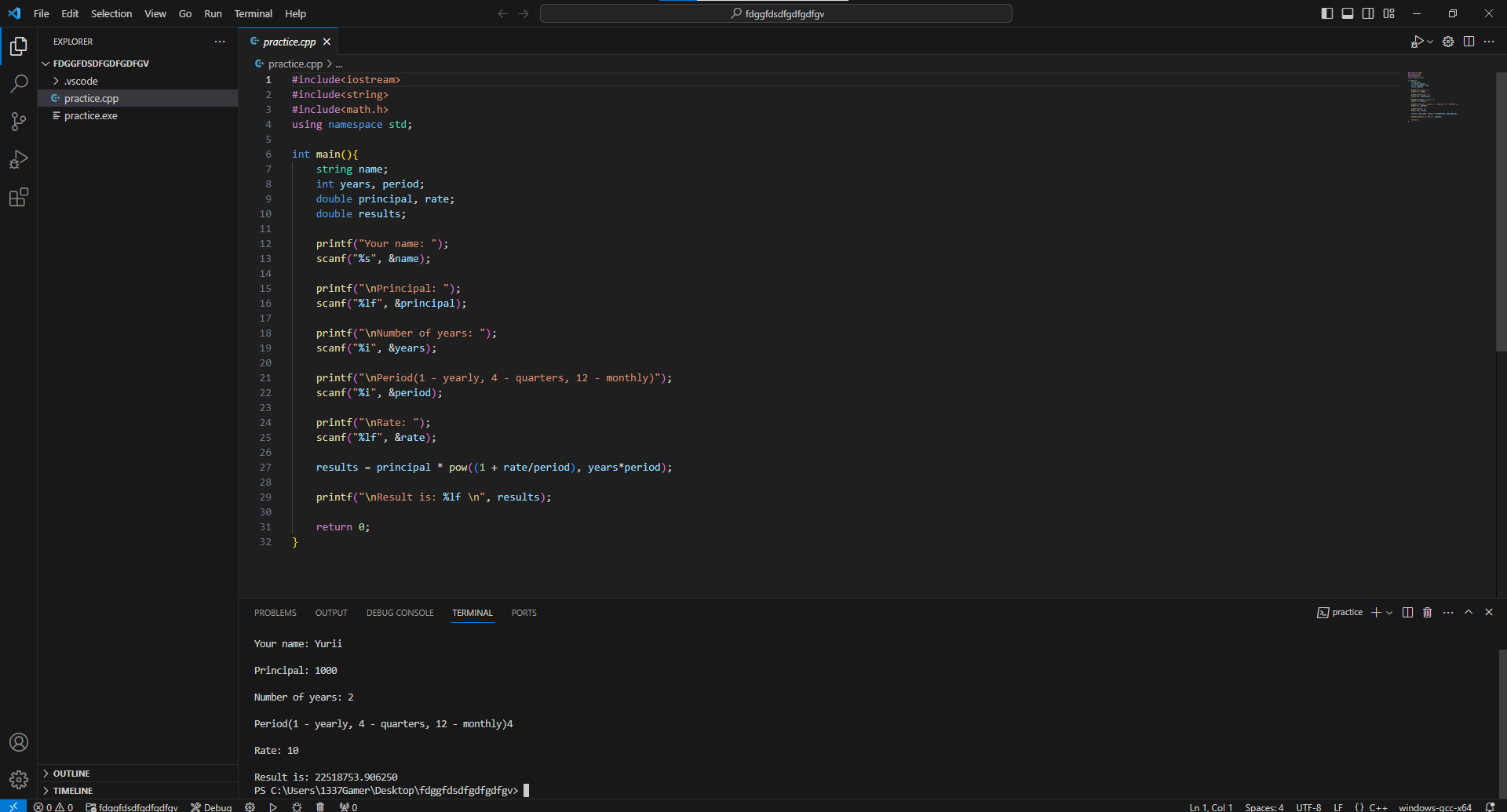
Посилання на файл програми у пул-запиті GitHub: https://github.com/artificial-intelligence-department/ai\_programming\_playground/blob/epic\_1\_pactice\_and\_labs\_yurii\_hembara/ai\_14/yurii\_hembara/Epic1/vns\_lab\_1\_task\_1\_yurii\_hembara.cpp

Рисунок 10 Код програми

Завдання №2 Самостійна практична робота (Hello World)

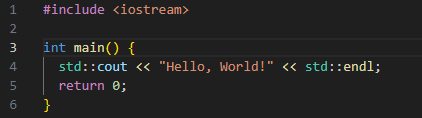


Рисунок 11 Код програми Hello World

Посилання на файл програми у пул-запиті GitHub: https://github.com/artificial-intelligence-department/ai\_programming\_playground/blob/epic\_1\_pactice\_and\_labs\_yurii\_hembara/ai\_14/yurii\_hembara/Epic1/self\_practice\_1\_task\_1\_yurii\_hembara.cpp

## **5. Результати виконання завдань, тестування та фактично затрачений час:**

Завдання №1 Було виконано і протестовано без проблем.

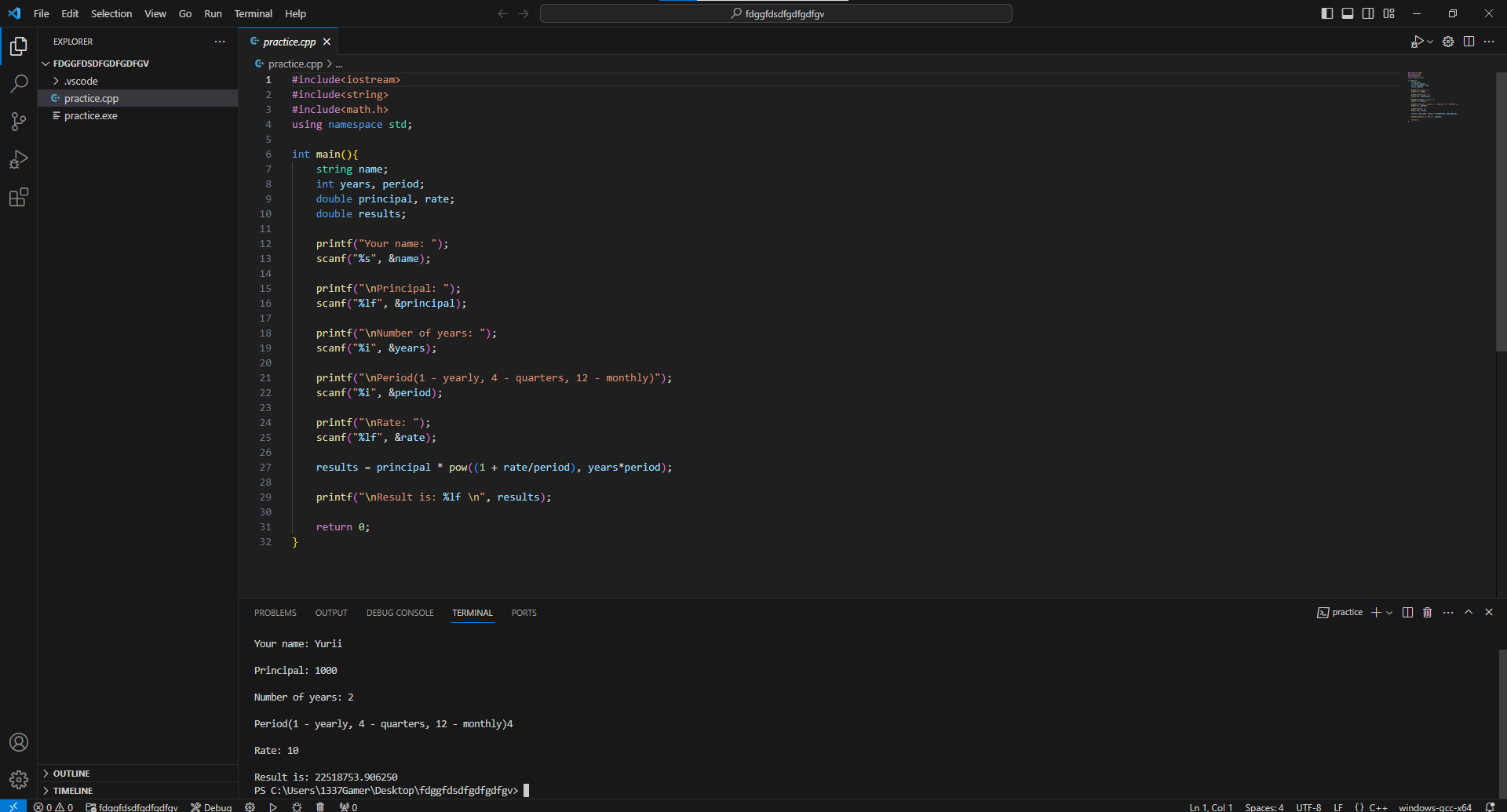


Рисунок 12 Результат роботи програми

Час затрачений на виконання завдання: 1.5 години

Завдання №2 Було виконано і протестовано без проблем.



Рисунок 13 Результат роботи програми

Час затрачений на виконання завдання: 3 хвилини

# **Висновки:**

В ході виконання завдань першого епіка, я ознайомився з середовищем програмування на C++ та взаємодіяв з деякими його елементами.