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-13

Сіренко Юрій Сергійович

# **Тема роботи:**

Налаштування та використання робочого середовища для програмування. Знайомство з VS Code, GitHub, Git Bash, Draw.io, Trello, створення перших програм.

# **Мета роботи:**

Ознайомитись з налаштуванням та використанням середовища. Встановити та налаштувати VS Code, ознайомитись з середовищем GitHub, , встановити та налаштувати Git Bash, ознайомитись з Algotester, Draw.io, Trello. Також ознайомитись з базовими операціями в мові програмування c++: ввід, вивід, обчислення значень; створити першу програму.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Налаштування VS Code
* Тема №2: GitHub, Git Bash
* Тема №3: Draw.io
* Тема №4: Trello
* Тема №5: Algotester
* Тема №6: MSYS2

1. Індивідуальний план опрацювання теорії:

* Тема №1: Налаштування VS Code
  + Джерела Інформації
  + Відео 1 - [Debugging C++ Program in Visual Studio Code (VSCode)](https://youtu.be/2VokW_Jt0oM?si=oNLBV6dY87HlJPGh)
  + Відео 2 - [How to Set up Visual Studio Code for C and C++ Programming](https://www.youtube.com/watch?v=77v-Poud_io&ab_channel=LearningLad)
  + Стаття 1 - [[Step By Step Guide] How to set up visual studio code for C and C++ programming 2021.](https://dev.to/narottam04/step-by-step-guide-how-to-set-up-visual-studio-code-for-c-and-c-programming-2021-1f0i)
  + Стаття 2 - [How to Write And Run C and C++ Code in Visual Studio Code](https://www.freecodecamp.org/news/how-to-write-and-run-c-cpp-code-on-visual-studio-code/)
  + Стаття 3 - [How to setup C/C++ in Visual Studio Code?](https://tooabstractive.com/how-to-tech/how-to-setup-c-and-cpp-in-visual-studio-code/)
  + Що опрацьовано:
    - Закачано та налаштовано VS Code.
    - Додано необхідні розширення для роботи з кодом. Сконфігуровано компілятор g++ та дебагер, детальніше це описано в Тема №6: MSYS2.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 26.09.2023
  + Звершення опрацювання теми: 30.09.2023
* Тема №2: GitHub, Git Bash
  + Джерела Інформації:
    - Відео 1 - [Введення до git](https://www.youtube.com/watch?v=ySKJF3ewfVk&list=PLJ6ZMUSN40FF8pBX4bv1mhVIgoem33Zfv&ab_channel=programmingmentorua)
    - Стаття 1 - [Set up Git](https://docs.github.com/en/get-started/quickstart/set-up-git)
    - Стаття 2 - [Customizing Git - Git Configuration](https://git-scm.com/book/en/v2/Customizing-Git-Git-Configuration)
  + Що опрацьовано:
    - Налаштовано профіль GitHub.
    - Ознайомлено з базовими командами Git.
    - Робота з Репозиторієм: завантаження та вивантаження файлів.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 30.09.2023
  + Звершення опрацювання теми: 06.10.2023
* Тема №3: Draw.io
  + Джерела Інформації:
    - Відео 1 - [Як створювати блок-схеми, діаграми в Draw.io.](https://www.youtube.com/watch?v=BAHx1xoA9kI&ab_channel=%D0%9F%D1%80%D0%BE%D1%81%D1%82%D0%BE%D0%B9%D0%B8%D0%BD%D1%84%D0%BE%D0%B1%D0%B8%D0%B7%D0%BD%D0%B5%D1%81)
    - Стаття 1 - [Design Flowchart In Programming](https://www.programiz.com/article/flowchart-programming)
  + Що опрацьовано:
    - Спосіб та мета створення зручних діаграм.
    - Найпоширеніші фігури в алгоритмах.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 06.10.2023
  + Звершення опрацювання теми: 07.10.2023
* Тема №4 Trello
  + Джерела Інформації:
    - Стаття 1 - [Get started with trello!](https://trello.com/guide/trello-101)
    - Відео 1 - [Ultimate Trello Setup (You'll want to copy this board!)](https://www.youtube.com/watch?v=0Nyz7OLD3Wo&ab_channel=Simpletivity)
  + Що опрацьовано:
    - Мета та процес створення дошки в Trello.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 08.10.2023
  + Звершення опрацювання теми: 08.10.2023
* Тема №5: Algotester.
  + Що опрацьовано:
    - Ознайомлено з цією платформою.
    - Виконано базове завдання в Algotester.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 26.09.2023
  + Звершення опрацювання теми: 25.10.2023
* Тема №6 MSYS2
  + Джерела Інформації:
    - Відео 1 - [How to Install MinGW (GCC/G++) Compiler in Windows 11](https://www.youtube.com/watch?v=fsmVbLMzBoo&ab_channel=ProgrammingKnowledge)
    - Відео 2 - [How to Install C and C++ compiler ( gcc g++ gdb ) on Windows 11 using mingw and msys2](https://www.youtube.com/watch?v=-gxwT-eAfvU&ab_channel=ProgrammingWithNick)
    - Стаття 1 - [Installing c++/g++ on Windows](https://www3.cs.stonybrook.edu/~alee/g++/g++.html)
    - Стаття 2 - [How to Install C and C++ Compilers on Windows](https://www.freecodecamp.org/news/how-to-install-c-and-cpp-compiler-on-windows/)
  + Що опрацьовано:
    - Інстальовано компілятор та дебагер за допомогою Mingw.
    - Введено низку команд по інструкції щоб завантажити g++, gdb та gcc компілятори.
    - Додано шлях до середовища змінних.
    - Перевірено наявність компіляторів.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 26.09.2023
  + Звершення опрацювання теми: 26.10.2023

# **Виконання роботи:**

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

Завдання №1 Робота над теорією

* Деталі завдання:
* Робота над вивченням та опрацюванням матеріалів, доданих вище відповідно до тем. Виділення найважливішої інформації та акцент на незрозумілі речі для одногрупників чи тіммейтів .

Завдання №2 Налаштування середовища програмування (VS Code)

* Деталі завдання
* Для налаштування програмного середовища необхідно познайомитись з базовими розширеннями: Code C/C++ Extensions, Code Runner, C/C++ Runner

Коротко про ці роширення:

C/C++ Extension Pack - надає підтримку синтаксису, автодоповнення, підказок, перевірки помилок

Code Runner - дозволяє швидко запускати C/C++ код з одного кліку або гарячої клавіші

C/C++ Runner – містить як ранер коду так і дебагер. Дебагер дозволяє швидко переглянути всі значення змінних в певному місці в коді

Завдання №3 Конфігурація Git та GitHub

* Деталі завдання:
* Створення облікового запису в GitHub. З’єднання Git’а з GitHub’ом. Налаштування Git, генерування SSH ключів, клонування та перше знайомство з репозиторієм: ознайомлення з командами git, за допомогою консолі клонedfyyz файлу на комп’ютер, його змінена і відправлення всіх змін назад.
* Важливі деталі: вивчити команди для подальшого користування git (git status, git add ., git commit -m, git push).

Завдання № 4 Конфігурація компіляторів, робота з MSYS2, основні команди консолі Linux

* Деталі завдання:

Знайомство з компонентами MSYS2(MinGW, Bash Shall),

Ознайомлення з основними компонентами MSYS2(MinGW, Bash Shall),встановлення та налаштування MSYS2 для роботи з VS Code. Інсталяція g++, gcc, gdb за допомогою MSYS2. Вивчення та використання ключових команд для командного рядка.

Завдання №5 Проектування за допомогою Draw.io

* Деталі завдання:

Ознайомлення з інформацією, щодо блок-схем, їх побудови та різноманітних позначень. Вивчення інтерфейсу заданих програм. Створення першого FlowChart у Draw.io.

Завдання №6 Конфігурація Trello

* Деталі завдання:
* Ознайомлення з Trello .Додавання списків на кожну дошку, що представляють різні етапи чи категорії завдань. Наприклад, "To Do", "In progress", "In review", "Done".Переміщення картки між списками, коли завдання переходить в інший стан (наприклад, з "In progress" в "Done") для відстеження прогресу проекту.

Завдання №7 реєстрація та знайомство з Algotester. Написання першої програми

* Деталі завдання

Створення облікового запису та виконання базової програми з Algotester за допомогою VS Code. Створення проекту, написання коду, збереження проекту, запуск програми, перевірка результату, фікс багів (при потребі)

Завдання №8 Виконання практичної роботи №1

* Деталі завдання:

Написання програми для обчислення суми депозиту (за допомогою формули складних відсотків)

* Важливі деталі:

Врахування того, що можливе обчислення не тільки щорічного нарахування відсотків, а й щомісячного та щоквартального.

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

Програма №1 Обчислення Складеного депозиту

* Блок-схема
* Планований час на реалізацію – 6 годин;
* Важливі деталі для врахування в імплементації

Використання функцій printf(), scanf() обчислення введених даних

Програма №2 Фібоначчі

* Блок-схема
* Планований час на реалізацію 1 година;
* Використання циклу for()

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

Завдання №2 Деталі по конфігурації середовища + скріншоти з підписами до скріншотів.

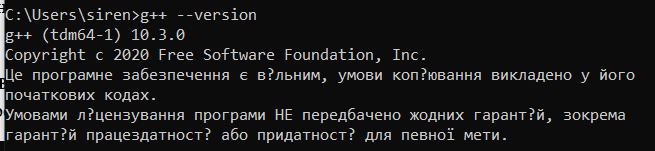


Рисунок Налаштування середовища

Версія С++ компілятора № до блоку з скріншотами до конфігурації

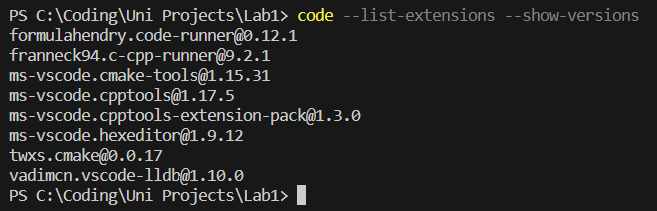


Рисунок Інсталяція необхідних розширень

Список встановлених розширень

Завдання №3 Коннект Git з GitHub

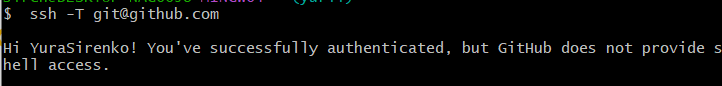


Рисунок Авторизація у свій аккаунт на гіт

Авторизація акаунту GitHub в Git

Завдання №4 налаштування змінних середовища

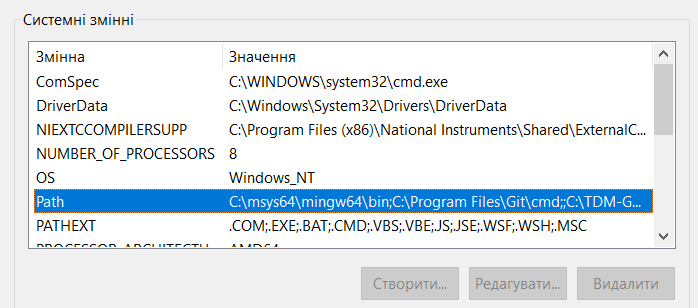


Рисунок Встановлення посилання для системи до компіляторів

Налаштування дебагера та робота з ним.

launch.json було налаштовано таким чином:

{

  "configurations": [

    {

      "name": "C/C++ Runner: Debug Session",

      "type": "cppdbg",

      "request": "launch",

      "args": [],

      "stopAtEntry": false,

      "externalConsole": true,

      "cwd": "c:/Coding/uni\_projects/epic2",

      "program": "c:/Coding/uni\_projects/epic2/algotester\_lab\_1\_task\_yurii\_sirenko.exe",

      "MIMode": "gdb",

      "miDebuggerPath": "gdb",

      "setupCommands": [

        {

          "description": "Enable pretty-printing for gdb",

          "text": "-enable-pretty-printing",

          "ignoreFailures": true

        }

      ]

    }

  ]

}

Цим дебагером було опрацьовано програму з алготестера про ману та хітпоінти:

Вхідні дані:

90

10

89

0

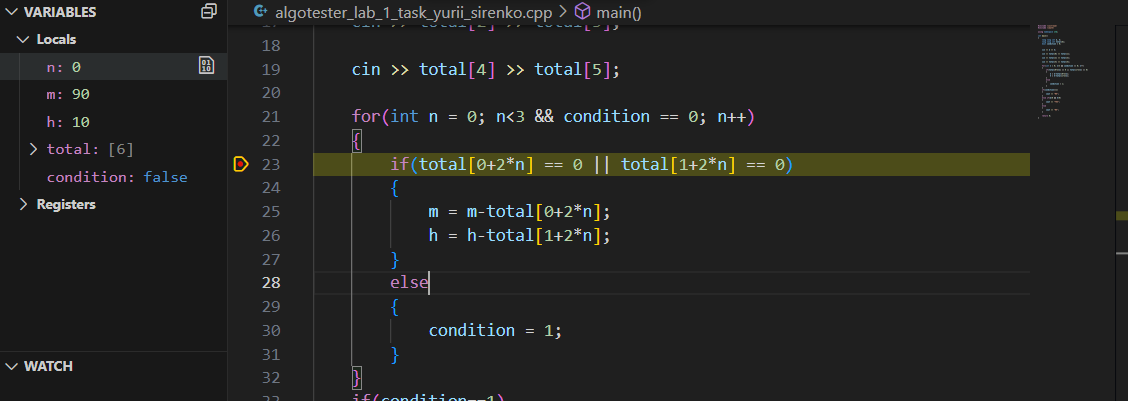
9

0

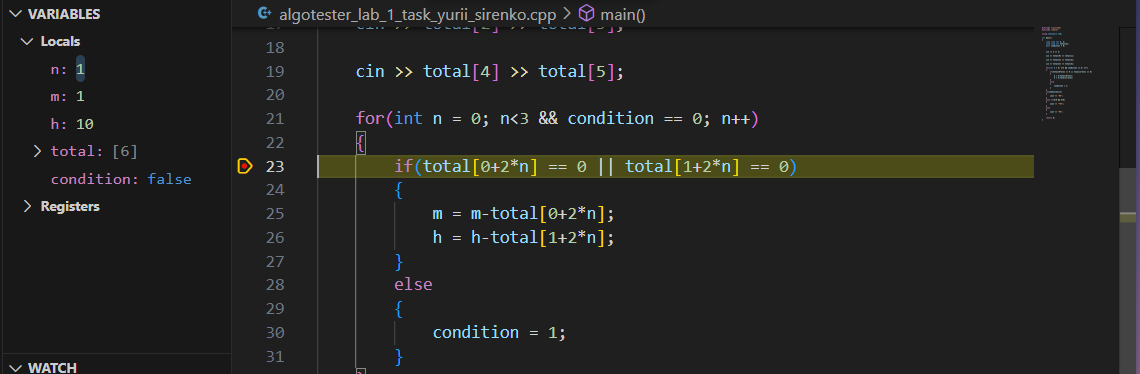
0

0

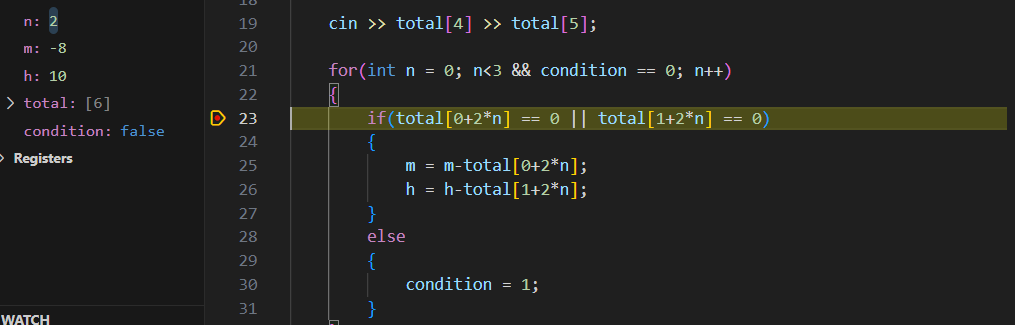
Start



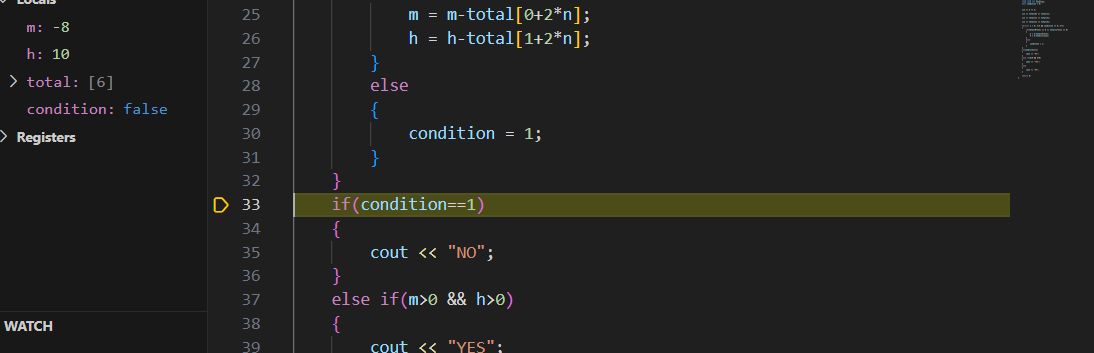
Itter 1



Itter 2



False condition



Output: NO

Налаштування до tasks.json

{

    "tasks": [

        {

            "type": "cppbuild",

            "label": "C/C++: gcc.exe build active file",

            "command": "C:/TDM-GCC-64/bin/gcc.exe",

            "args": [

                "-fdiagnostics-color=always",

                "-g",

                "${file}",

                "-o",

                "c:/Coding/uni\_projects/epic2.exe",

                ""

            ],

            "options": {

                "cwd": "C:/TDM-GCC-64/bin"

            },

            "problemMatcher": [

                "$gcc"

            ],

            "group": {

                "kind": "build",

                "isDefault": true

            },

            "detail": "Task generated by Debugger."

        }

    ],

    "version": "2.0.0"

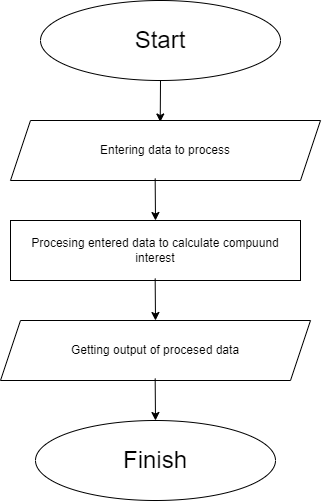
}

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

Завдання №1 Практична №1 Посилання на файл програми у пул-запиті GitHub:

[Pull request link](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/43/commits/48b12d30b2fba0185e3570275a744aebd818fac7#diff-9e7ab39dd60ec5de8f3dbb6f413edba8acf92f995f86e84e241e6a1c2f83578b)

Таблиця Блок-схема до програми складеного відсотка



#include <cstdio>

#include <cmath>

#include <iostream>

int main()

{

    char name[30]; //Оголошуємо змінну для імені

    double principal; //Оголошуємо змінну для початкової суми

    double rate; //Оголошуємо змінну для проценту депозиту

    short years; //Оголошуємо змінну для періоду депозиту

    long result; //Оголошуємо змінну для результату обчислень

    int period; //Оголошуємо змінну для кількості складань в рік

    printf("Enter your name:\n");

    scanf("%s",name); //приймаємо змінну для імені

    printf("Enter the principal amount: \n");

    scanf("%lf", &principal); //приймаємо змінну для відсотку

    printf("Enter the rate of interest (in percentage): \n");

    scanf("%lf", &rate); //приймаємо змінну для відсотку

    printf("How times per year your principal would increase: print 1 - yearly, 4 - by season, 12 - monthly");

    scanf("%i", &period); //приймаємо змінну для періодичності відсотку

    printf("Enter the number of years: \n");

    scanf("%hd", &years); //приймаємо змінну для років

    result = principal\*pow((1+(rate/100)),years\*period); // Обчислюємо складений депозит за формулою

    printf("Hello, %s", name," ! \n");

    printf(" The compound interest for a principal of %lf", principal);

    printf("at a rate of %lf", rate);

    printf(" %i times ", period);

    printf("for %hd ", years);

    printf("years is: %lu", result); //Виводимо фінальний результат

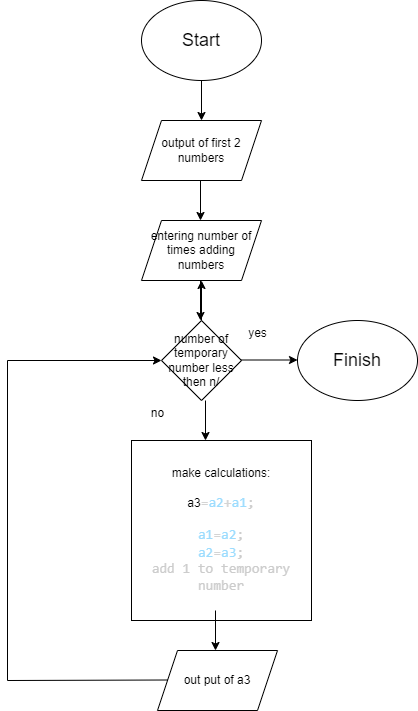
    return 0;

}

Завдання №2 Числа Фібоначчі

[Pull request link](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/43/commits/48b12d30b2fba0185e3570275a744aebd818fac7#diff-ad90f38a2934e10fce77e014033b6a66124063d49bb8db3ea15cd7de798153f5)

Таблиця Блок-схема до своєї задачі



#include <iostream>

#include <windows.h>

using namespace std;

int main()

{

    HANDLE console\_color;

    console\_color = GetStdHandle(STD\_OUTPUT\_HANDLE);

    SetConsoleTextAttribute(console\_color, 2); //Для створення кольору консолі

    unsigned long long int a1 = 1; //Оголошення змінних

    unsigned long long int a2 = 1;

    unsigned long long int a3;

cout <<"1 \n1 \n";

    for(int n = 0; n<10; n++) //Створення циклу спнцифічного для чисел Фібоначчі

    {

    a3=a2+a1;

    a1=a2;

    a2=a3;

    cout << a3 <<"\n"; //Вивід значень

    }

}

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

Завдання №1 Деталі по виконанню і тестуванню програми

Enter your name:

Yura

Enter the principal amount:

10000

Enter the rate of interest (in percentage):

12

How times per year your principal would increase: print 1 - yearly, 4 - by season, 12 - monthly1

Enter the number of years:

4

Hello, Yura The compound interest for a principal of 10000.000000at a rate of 12.000000 1 times for 4 years is: 15735

Час затрачений на виконання завдання – 3 години

Завдання №2 Деталі по виконанню і тестуванню програми

Програма виводить перші 12 чисел Фібоначчі

1

1

2

3

5

8

13

21

34

55

89

144

Час затрачений на виконання завдання – 40 хвилин

# **Висновки:**

В ході цього епіку я ознайомився з програмуванням в загальному. Загалом, вивчення Visual Studio, ознайомлення з Git та GitHub, Trello, Draw.io, Algotester виявилося корисним та цінним досвідом, який покращив мої навички розробки програмного забезпечення та керування версіями. Ці інструменти стануть невід'ємною частиною мого арсеналу для розробки програм та спільної роботи з іншими розробниками.