

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Кафедра систем штучного інтелекту



Звіт

про виконання

Лабораторних та практичних робіт № 1

з дисципліни: «Мови та парадигми програмування»

з розділу: Вступ до Розробки: Налаштування та Використання Середовища"»

Виконав(ла):

студент групи ШІ-14

Вульчин Володимир Сергійович

Тема роботи:

Конфігурація та ознайомлення з середовищем Visual Studio Code, встановлення необхідних утиліт під мову C++, ознайомлення з Git та GitHub, Trello, Algotester, Draw.IO, Flowcharts

Мета роботи:

1. Ознайомитись з Package Managers OS та командами
2. Ознайомитись з Console Commands в Linux подібному терміналі
3. Встановити та сконфігурувати Visual Studio Code
4. Встановити Розширення для C++ на систему та Visual Studio Code
5. Ознайомитись з Дебагером та Лінером для C++
6. Встановити та ознайомитись з Git та командами
7. Зареєструватись та ознайомитись з GitHub
8. Ознайомитись з GitHub пул реквестами та Код ревью
9. Зареєструватись та ознайомитись з Trello
10. Зареєструватись та ознайомитись з Algotester
11. Ознайомитись з FlowCharts та Draw.io
12. Ознайомитись з Word та створенням Звітів на Практичні та Лабораторні
13. Запустити програмний код C++ в робочому середовищі та оформити звіт
14. Виконати теоретичний план по ознайомленню з інструментами”

Теоретичні відомості: (таски з трело)

1) Теоретичні відомості з переліком важливих тем:

- Тема №1: Управління вимогами та проектна діяльність з Draw.io та Google Docs
- Тема №2: Лаб# Конфігурація: Trello
- Тема №3: Лаб# Конфігурація: VSC C/C++ Extensions, IntelliSense, Code Runner
- Тема №4: Лаб# Конфігурація: GitHub
- Тема №5: Лаб# Конфігурація: Git
- Тема №6: Лаб# Конфігурація: Algotester
- Тема №7: Експериментальні вправи діяльності - запустити першу програму

2) Індивідуальний план опрацювання теорії:

- Тема №1: Управління вимогами та проектна діяльність з Draw.io та Google Docs
 - Джерела Інформації:
 - Стаття (Draw.io): <https://drawio-app.com/tutorials/step-by-step-guides/>
 - Практичне/Лабораторне заняття.

- Що опрацьовано:
 - Ознайомився з Draw.io та зрозумів що це – сайт, який дає можливість утворювати схеми, на яких зображено етапи певного процесу. Шаблони блок-схем містять різні фігури, за допомогою яких, особливо зручно, легко відобразити основі процеси коду, та алгоритм їх дій.
 - Google Docs - також не менш важливий сайт/застосунок, у якому можна зберігати різні типи інформації. Наприклад: покрокові цілі до виконання задачі, план дій до виконання того, чи іншого завдання, а також списки різних форматів. (Список груп, команд, особистих задач, тощо.)
 - Статус: **Ознайомлений** / Ознайомлений частково / Не ознайомлений
 - Початок опрацювання теми: 1.10.2023
 - Звершення опрацювання теми: 1.10.2023
- Тема №2: Лаб# Конфігурація: Trello
- Джерела Інформації:
 - Практичне/Лабораторне заняття
 - Стаття: <https://trello.com/guide/create-project#create-a-board>
 - Що опрацьовано:
 - Ознайомився з Trello та зрозумів що це – допоміжний сайт, який дає можливість викладачам та куратора команди (в рамках курсу) або Тех. Ліду (в рамках робочого процесу в компанії), відслідковувати на якій стадії виконання роботи знаходиться команда.
 - Доповнення – також Trello дає чудову можливість розподілити категорії процесу по групам, для зручності виконання. До прикладу:
- A screenshot of the Trello web interface. It shows a horizontal board with five columns: 'To Do', 'Blocked', 'In Progress', 'In Review', and 'Done'. Each column has a three-dot menu icon to its right. The columns are dark blue with white text.
- Статус: **Ознайомлений** / Ознайомлений частково / Не ознайомлений
 - Початок опрацювання теми: 1.10.2023
 - Звершення опрацювання теми: 3.10.2023
- Тема №3: Лаб# Конфігурація: VSC C/C++ Extensions, IntelliSense, Code Runner.
- Джерела Інформації:
 - Телеграм канал (Epic 1 – Practice Topics Discussion)
 - Практичне/Лабораторне заняття.
 - Що опрацьовано:
 - Так як я користуюсь ОС Windows, мені потрібно встановити програму MSYS2, з допомогою якої, я зміг встановити компілятори для мови C++. При цьому, використовував пакет менеджерів PACMAN та вводив команди для встановлення компіляторів. (gcc –version, g++ --version, gdb – version)

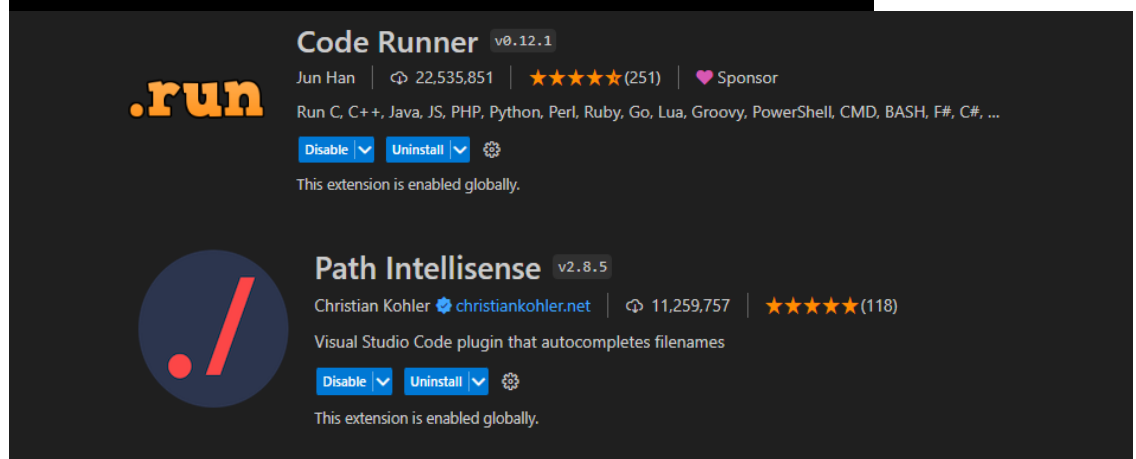
```

MS@DESKTOP-3VJBN1D MSYS ~
$ g++ --version
g++ (GCC) 13.2.0
Copyright (C) 2023 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

MS@DESKTOP-3VJBN1D MSYS ~
$ gdb --version
GNU gdb (GDB) 13.2
Copyright (C) 2023 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

MS@DESKTOP-3VJBN1D MSYS ~
$ gcc --version
gcc (GCC) 13.2.0
Copyright (C) 2023 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

```



- Статус: **Ознайомлений** / Ознайомлений частково / Не ознайомлений
- Початок опрацювання теми: 5.10.2023
- Звершення опрацювання теми: 6.10.2023

- Тема №4: Лаб# Конфігурація: GitHub

- Джерела Інформації:
 - Практичне/Лабораторне заняття.
 - Однорупники
 - Відео на YouTube.
 -
- Що опрацьовано:
 - Опрацьовав матеріал по GitHub, та зрозумів, що він пов'язаний з GitHub.

```

5) git status - команда перевіряє кількість та наявність незакомічених файлів.
6) git add . - команда яка дає дозвіл на коміт (символ . вибирає всі файли)
7) git remote add НАЗВА HTTPS - кажу гіту що "назва" = ссилці HTTPS (ця дія означає, що локальний репозиторій підключив до гітхабу)
8) git commit -m "Назва коміту" - кажу гіту, що закомічені файли хочу надіслати на віддалений репозиторій.
9) git push -u "назва заміни HTTPS" master - загрузив файли на віддалений репозиторій.

```

- Статус: **Ознайомлений** / Ознайомлений частково / Не ознайомлений
- Початок опрацювання теми: 5.10.2023
- Звершення опрацювання теми: 5.10.2023

- Тема №5: Лаб# Конфігурація: Git.

- Джерела Інформації:
 - Практичне/Лабораторне заняття.

- Одногрупники
- Відео на YouTube.

- Що опрацьовано:
 - Зрозумів, що таке Git та для чого він потрібен.

```
1) cd "шлях папки"
2) git init (команда init створює локальний репозиторій в цій папці)
3) git config --global user.name "Назва" - вказати ГІТУ своє ім'я.
4) git config --global user.email "Пошта" - вказати ГІТУ свою пошту.
```

- Статус: **Ознайомлений** / Ознайомлений частково / Не ознайомлений
- Початок опрацювання теми: Дата
- Звершення опрацювання теми: Дата

- Тема №6: Лаб# Конфігурація: Algotester.

- Джерела Інформації:
 - Практичне/Лабораторне заняття.
- Що опрацьовано:
 - Створив аккаунт, та зрозумів, що сайт Algotester теоретично заміняє самостійні та контрольні роботи, адже сайт автоматично тестує студента та перевіряє роботоздатність програми.
 - Коментар 2
- Статус: Ознайомлений / **Ознайомлений частково** / Не ознайомлений
- Початок опрацювання теми: 2.10.2023
- Звершення опрацювання теми: 2.10.2023

- №7: Експериментальні вправи діяльності - запустити першу програму.

- Що опрацьовано:

```
testzvit.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5
6      int a, b, sum;
7
8      cout << "Enter two integers: ";
9      cin >> a >> b;
10
11
12      sum = a * b;
13
14
15      cout << a << " * " << b << " = " << sum << endl;
16
17      return 0;
18  }
```

Enter two integers: 1305
1305
1305 * 1305 = 1703025

Написав код, який дає можливість користувачу ввести через термінал значення А та Б. Після чого програма вираховує їх добуток.

- Статус: **Ознайомлений** / Ознайомлений частково / Не ознайомлений
- Початок опрацювання теми: 7.10.2023

- Звершення опрацювання теми: 7.10.2023

Виконання роботи:

1. Опрацювання завдання та вимог до програм та середовища:

Завдання №1 Створення першої власної програми

- Варіант завдання (відсутній)
- Деталі завдання:
- Закодити першу програму, яка буде вирішувати задачу $a*b$

Завдання №2 Створення задачі, яка буде вираховувати прибуток від вкладення на певний період.

- Варіант завдання (відсутній)
- Деталі завдання:
- Закодити програму, через змінні `double` (для числових значень з плаваючою комою(наприклад грошей у відсотках)), `int` (для проміжку часу). Після чого програма має запитатись у користувача:
 - 1) Вкажіть своє ім'я:
 - 2) Вкажіть к-сть вкладених грошей:
 - 3) Вкажіть відсоткову ставку:
 - 4) Вкажіть термін часу:
 - 5) Вкажіть кількість відсотків за певний перелік часу

Завдання №3 Створення програми, яка в залежності від погоди, вкаже вам, що вам потрібно одягнути.

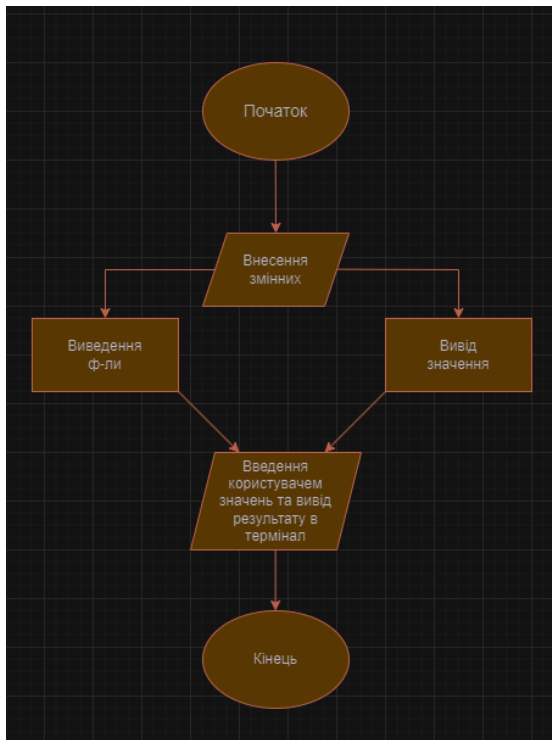
- Варіант завдання (відсутній)
- Деталі завдання:

Закодити програму, через оператор `switch case`, `if else`, `if`, `else`, що програма змогла вивести при вказаній користувачем погоди, які речі йому слід одягнути.

2. Дизайн та планована оцінка часу виконання завдань:

Програма №1 Створення першої власної програми

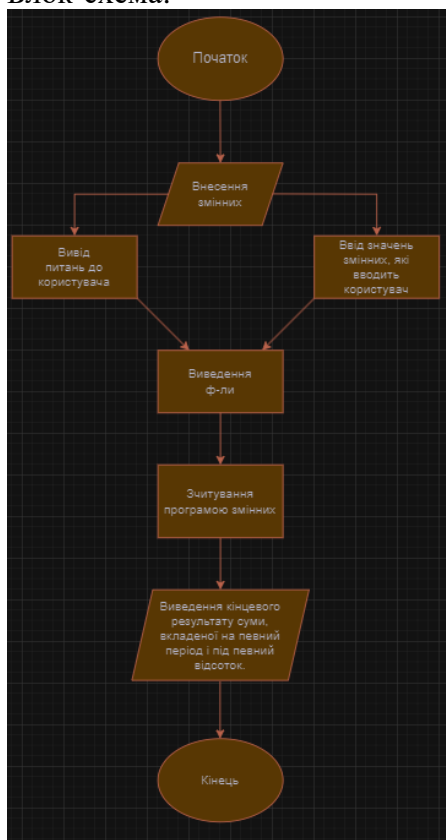
Блок-схема:



- Планований час на реалізацію: 15 хвилин.

Програма №2 Створення задачі, яка буде вираховувати прибуток від вкладення на певний період.

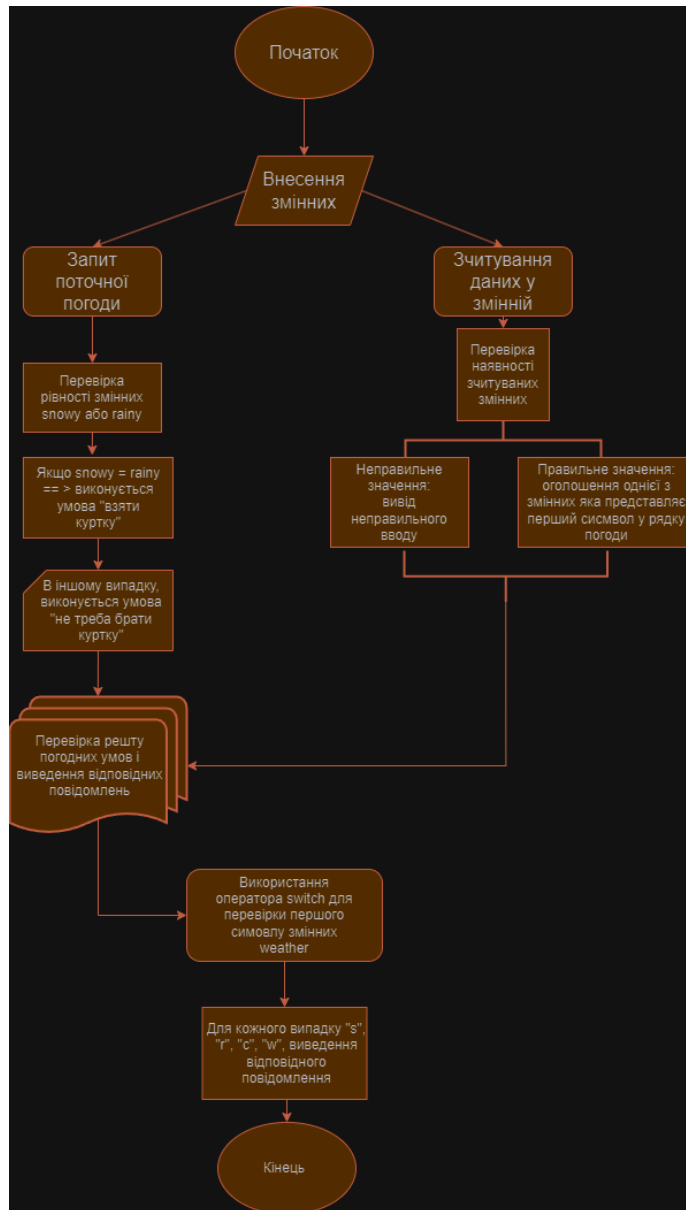
- Блок-схема:



- Планований час на реалізацію: 1 година.

Програма №3 Створення програми, яка в залежності від погоди, вкаже вам, що вам потрібно одягнути.

- Блок-схема



- Планований час на реалізацію 1,5 години.

3. Конфігурація середовища до виконання завдань: (не потрібно)

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

Підпис та № до блоку з скріншотами до конфігурації

4. Код програм з посиланням на зовнішні ресурси:

Завдання №1 Створення першої власної програми

Посилання на файл програми у пул-запиті GitHub (https://github.com/artificial-intelligence-department/ai_programming_playground/pull/12)


```

#include <iostream>
using namespace std;

int main() {

    int a, b, sum;

    cout << "Enter two integers: ";
    cin >> a >> b;

    sum = a * b;

    cout << a << " * " << b << " = " << sum << endl;

    return 0;
}

```

Підпис та № до блоку з кодом програми

Завдання №2 Створення задачі, яка буде вираховувати прибуток від вкладення на певний період.
Посилання на файл програми у пул-запиті GitHub (https://github.com/artificial-intelligence-department/ai_programming_playground/pull/12)

```

#include <iostream>
#include <string>
#include <math.h>

int main(){
    std::string name;
    double pricipal, rate, total_amount, a ;
    int year, period ;

    std::cout << "Enter your name: " ;
    std::cin >> name;

```

```

std::cout << "Enter the principal amount: ";
std::cin >> pricipal;

std::cout << "Enter the rate of interest (in percentage): ";
std::cin >> rate;

std::cout << "Enter the number of years: ";
std::cin >> year;

std::cout << "Enter the number of times interest is compounded per year (e.g., 1
for annually, 4 for quarterly, 12 for monthly): ";
std::cin >> period;

total_amount = pricipal * pow((1+ (rate/100) / period) , (period * year));
a = total_amount - pricipal;

std::cout << "Hello, " << name << "\n";
std::cout << "The compound interest for a principal of " << pricipal << " at a
rate of " << rate << "% compounded " << period << " times a year for " << year << "
years is " << a << "\n";
std::cout << "The total amount after 3 years is: " << total_amount << "\n";

return 0;
}

```

Завдання №3 Створення програми, яка в залежності від погоди, вкаже вам, що вам потрібно одягнути.

Посилання на файл програми у пул-запиті GitHub (https://github.com/artificial-intelligence-department/ai_programming_playground/pull/12)

```

#include <iostream>
#include <string>
using namespace std;

int main()
{
    string weather;

    cout << "Enter the weather conditions: ";
    getline(std::cin, weather);
    if (weather != "cloudy" && weather != "snowy" && weather != "rainy" && weather !=
"windy" && weather != "sunny")

```

```

{
    cout << "You entered inappropriate data";
}

else
{
    char i = weather[0];
    if (weather == "snowy" || weather == "rainy")
    {
        cout << "You must take the jacket" << endl;;
    }
    else
    {
        cout << "You may not take the jacket" << endl;;
    }

    if (weather == "sunny")
    {
        cout << "Nice day for a picnic!" << endl;
    }
    else if (weather == "rainy")
    {
        cout << "An ideal for reading a book" << endl;
    }
    else if (weather == "cloudy")
    {
        cout << "Maybe, visit the museum?" << endl;
    }
    else if (weather == "snowy")
    {
        cout << "What about making snowman?" << endl;
    }
    else
    {
        cout << "What about flying a kite?" << endl;
    }

    switch(i)
    {
    case 's':
        if (weather == "sunny")
        {

```

```

        cout << "Wear your favourite shoes" << endl;
    }
    else
    {
        cout << "Snow boots will warm up your foots" << endl;
    }
    break;
case 'r':

    cout << "Rainy boots is an ideal shoes" << endl;
    break;
case 'c':
    cout << "You can wear any shoes" << endl;
    break;
case 'w':
    cout << "Wear something strong" << endl;
    break;

}

}
}

```

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

Завдання №1 Деталі по виконанню і тестуванню програми

Output:

Enter two integers: 3

4

3 * 4 = 12

Затрачений час: 15 хв.

Завдання №2 Створення задачі, яка буде вираховувати прибуток від вкладення на певний період.

Input:

Enter your name: Volodymyr

Enter the principal amount: 50000

Enter the rate of interest (in percentage): 5

Enter the number of years: 3

Enter the number of times interest is compounded per year (e.g., 1 for annually, 4 for quarterly, 12 for monthly): 4

Output:

Hello, Volodymyr!

The compound interest for a principal of 50000.00 at a rate of 5.00% compounded 4 times a year for 3 years is: 8037.73

The total amount after 3 years is: 58037.73

Затрачений час: 1 година.

Завдання №3 Створення програми, яка в залежності від погоди, вкаже вам, що вам потрібно одягнути. (в залежності від умови)

Enter the weather conditions: sunny

You may not take the jacket

Nice day for a picnic!

Wear your favourite shoes

Затрачений час: 1,5 годин.

Висновки:

Під час вивчення цих матеріалів було здійснено значний крок у підготовці до роботи з програмуванням та розробки перший програм. Ось короткий огляд кожного пункту:

Ознайомлення з Package Managers OS та командами: Розуміння систем управління пакетами та вміння використовувати їхні команди для встановлення та оновлення програм та бібліотек.

Ознайомлення з Console Commands в Linux подібному терміналі: Володіння (деякими) командами терміналу важливо для роботи з Linux і серверами.

Встановлення та налаштування Visual Studio Code: Visual Studio Code - це потужний і зручний редактор для програмування, який допомагає в розробці. (Я навчився, самостійно встановлювати компілятор на C++)

Встановлення розширень для C++ на систему та Visual Studio Code: Розширення допомагають зробити розробку на C++ більш зручною та продуктивною.

Ознайомлення з Дебагером та Лінером для C++: Дебаггер і лінер допомагають виявляти та виправляти помилки в коді.

Встановлення та ознайомлення з Git та командами: Git є найпопулярнішою системою керування версіями, і знання його команд важливо для спільної роботи над проектами. (Наприклад: повернутись до минулої версії коду, якщо виникли якісь проблеми)

Зареєструватись та ознайомитись з GitHub: GitHub - це платформа для спільної роботи над кодом, і вона допомагає зберігати та відстежувати зміни в проектах за допомогою Git`а.

Ознайомлення з GitHub пул реквестами та Код ревью: ці процеси важливі для спільної роботи в команді та покращення якості коду.

Зареєструватись та ознайомитись з Trello: Trello допомагає організувати робочі завдання та проекти, та відслідковувати етап розвитку команди.

Зареєструватись та ознайомитись з Algotester: сайт, який дає змогу перевірити володіння мовою програмування, та автоматично перевірити задані задачі.

Ознайомлення з FlowCharts та Draw.io: розібрався з цими сайтами, які надають змогу зробити блок-схему до програмного коду або якогось івенту.

Ознайомлення з Word та створенням звітів на практичні та лабораторні роботи: Володіння текстовим редактором і здатність створювати звіти є важливими навичками для документування роботи, наприклад, як зараз.

Запуск програмного коду C++ в робочому середовищі та оформлення звіту: Практична реалізація коду і створення звіту є важливими етапами в програмуванні, тому я зміг написати 3 різні програми і успішно їй задебажати.

Загалом, ці пункти сприяли розвитку різних аспектів моїх навичок і знань, що необхідні для ефективної роботи розробника як в команді, так і самотійно

