

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



Звіт

про виконання лабораторних та практичних робіт блоку № 1

На тему: «Програмування: алгоритм, програма, код. Системи числення.
Двійкова система числення. Розробка та середовище розробки програми.»
з дисципліни: «Основи програмування»

до:

Практичних Робіт до блоку № 1

Виконав:
Студент групи ШІ-12
Стефанович Євгеній

Львів 2024

Тема роботи:

- Налаштування VSCode (компілятора для C++)
- Освоєння Linux команд
- Створення дошок в Trello
- Робота з Git та GitHub
- Створення блок-схем в Draw.io
- Виконання задач на сайті Algotester.com
- Робота в команді

Мета роботи:

Навчитись створювати блок-схеми в Draw.io, користуватись дошками в Trello, вивчити Linux команди для роботи з консоллю, налаштувати Visual Studio Code для роботи з C++, працювати з Git та GitHub, виконувати задачі на Algotester.com, освоїти двійкову систему числення та операції над нею. Покращити навички роботи в команді (SoftSkills).

Теоретичні відомості:

1. Для налаштування компілятора у VSCode використовував сайт: <https://code.visualstudio.com/docs/cpp/config-mingw>.

Витрачений час: близько 10 годин, бо була проблема з шляхом куди встановлювався розширення

2. Вивчав Linux команди тут: <https://kinsta.com/blog/linux-commands/>, також звертався до ChatGPT для детальніших роз'яснень.

Витрачений час: 1.5 години.

3. Для освоєння Git та GitHub використовував міні-курс Андрія Кравця на YouTube:

https://www.youtube.com/watch?v=oofAm4x6oOE&list=PL3o5sNxukLFCKfvxez4nEZQ_cpWcXtb06&pp=iAQB, а також сайт <https://git-scm.com/book/uk/v2> де є пояснення до всіх речей пов'язаних з Git.

Витрачений час: 2 дні, бо були проблеми з клонуванням репозиторію за допомогою ключа SSH.

4. Створення дошок в Trello не зайняло багато часу, адже це інтуїтивно зрозуміла платформа для контролю виконання завдань.

Витрачений час: 30 хвилин.

5. Створення блок-схем в Draw.io - я вже маю досвід роботи
Витрачений час: 5 хвилин .

Виконання роботи:

1. Опрацювання завдання та вимог до програм та середовища:

Завдання № 1 Practice Task

Обчислення складних відсотків за депозитом

Задача: Обчислити складні відсотки для депозиту, який був відкритий в банку на певний період часу під фіксовані відсотки з різними варіантами виплати відсотків.

Вимоги:

1. Використати функції `scanf` та `printf` для зчитування і форматування вводу/виводу;
2. В кінці програма має вивести повну інформацію про вкладені кошти, загальну суму інвестиції і суму самого заробітку.

Завдання № 2 Self Practice Task

Марічка і печиво

Задача: Зібралися Зеник і Марічка разом з пластунами в похід. Похід — серйозна справа. Потрібно запаситись продуктами харчування та розподілити їх споживання по днях так, щоб всім вистачило. Цього разу Зеник слідкує за тим, щоб печива вистачило аж до останнього дня походу. Зеник чітко знає, скільки пачок печива повинно залишитись кожного дня, і щовечора перераховує їх. Якщо Зеник побачить, що залишилось менше пачок, ніж повинно залишитись за його розрахунками, він неодмінно знайде того, хто з'їв забагато печива, і покарає його.

Марічка дуже любить печиво. Сьогодні, коли всі пластуни покинуть свої намети і підуть купатися в річку, Марічка планує непомітно з'їсти трохи печива. Звісно, Марічка не хоче бути покараною і дуже боїться, щоб Зеник не помітив пропажу.

Марічка підгледіла, скільки пачок печива є в рюкзаку Зеника. Також вона знає, скільки штук в кожній пачці. Марічці не терпиться дізнатися, скільки ж печива вона зможе з'їсти так, щоб Зеник не помітив. Зеник помітить пропажу печива з деякої пачки тоді і тільки тоді, коли Марічка повністю спустошить її.

Вимоги:

1. Обмеження в часі та розмірі програми: 4 сек., 256 MiB
2. $1 \leq n \leq 10^5$; $0 \leq a_i \leq 10^9$

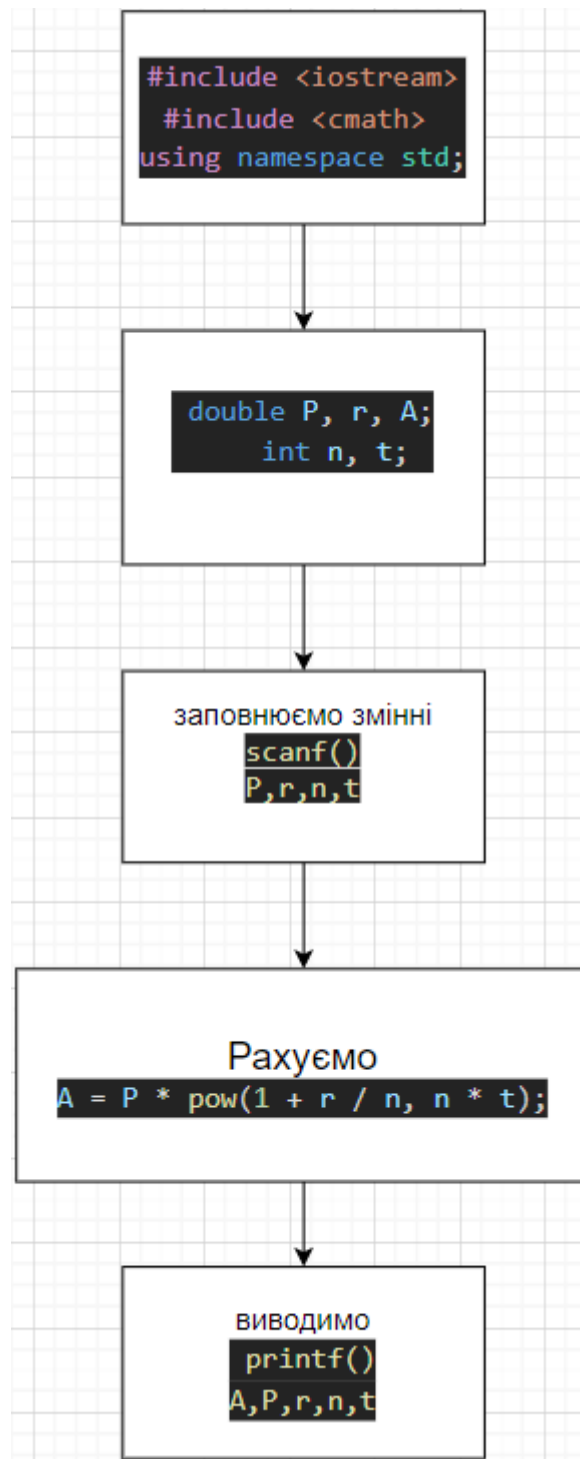
Завдання № 3 Калькуляції у двійковій системі

	Завдання на калькуляції в двійковій системі
1	Згенерувати в рандомайзері десяткове число у від 20 до 99
2	Згенерувати в рандомайзері десяткове число x від 20 до 99
3	Перевести у двійкову систему числення
4	Перевести x у двійкову систему числення
5	Додати два двійкових числа x та y
6	Відняти від більшого двійкового числа менше двійкове число

7	Більше двійкове число поділити на менше двійкови число число
8	Більше двійкове число помножити на менше двійкови число число
9	Згенерувати в рандомайзері десяткове число k від 20 до 99
10	Перевести k у 16-ву систему числення

2. Графічне Practice Task Draw.io

представлення
за допомогою



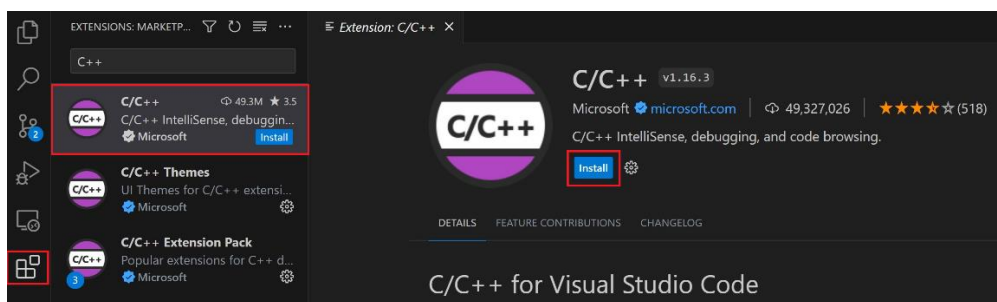
Виконав це завдання за 45 хвилин, враховуючи ознайомлення з функціями scanf, printf та бібліотеки <math.h>.

3. Конфігурація середовища до виконання завдань:

Скріншоти конфігурації середовища з підписами, що на цьому скріншоті

1.У мене вже був встановлений VScode:

2.Встановлення розширення для C/C++:



3.Встановлюємо MSYS2 і вводимо туди таку командну:
pacman -S --needed base-devel mingw-w64-ucrt-x86_64-toolchain

```
$ pacman -S --needed base-devel mingw-w64-ucrt-x86_64-toolchain
warning: base-devel-2022.12-2 is up to date -- skipping
:: There are 19 members in group mingw-w64-ucrt-x86_64-toolchain:
:: Repository ucrt64
 1) mingw-w64-ucrt-x86_64-binutils  2) mingw-w64-ucrt-x86_64-crt-git
 3) mingw-w64-ucrt-x86_64-gcc      4) mingw-w64-ucrt-x86_64-gcc-ada
 5) mingw-w64-ucrt-x86_64-gcc-fortran 6) mingw-w64-ucrt-x86_64-gcc-libgfortran
 7) mingw-w64-ucrt-x86_64-gcc-libs  8) mingw-w64-ucrt-x86_64-gcc-objc
 9) mingw-w64-ucrt-x86_64-gdb     10) mingw-w64-ucrt-x86_64-gdb-multiarch
11) mingw-w64-ucrt-x86_64-headers-git 12) mingw-w64-ucrt-x86_64-libgccjit
13) mingw-w64-ucrt-x86_64-libmangle-git 14) mingw-w64-ucrt-x86_64-libwinpthread-git
15) mingw-w64-ucrt-x86_64-make     16) mingw-w64-ucrt-x86_64-pkgconf
17) mingw-w64-ucrt-x86_64-tools-git 18) mingw-w64-ucrt-x86_64-winpthread-git
19) mingw-w64-ucrt-x86_64-winstorescompat-git
Enter a selection (default=all): |
```

4.Встановлюємо шлях до нашого компілятора

В пошуку шукаємо: Змінити змінні оточення для вашого облікового запису

Заходимо в Path

Створюємо новий шлях

Вказуємо C:\msys64\mingw64\bin

Компілятор встановлено та налаштовано!

Завдання №1 Practice Task:

```
Users > Eugene > Desktop > labs > Epic-1 > Epic1deposit.cpp > ...
1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4
5  int main() {
6
7      double P, r, A;
8      int n, t;
9
10
11      printf("Введіть початкову суму інвестиції (P): ");
12      scanf("%lf", &P); // double
13
14      printf("Введіть річну процентну ставку (r): ");
15      scanf("%lf", &r); // double
16
17      printf("Введіть кількість нарахувань відсотків на рік (n): ");
18      scanf("%d", &n); // int
19
20      printf("Введіть кількість років (t): ");
21      scanf("%d", &t); // int
22
23
24      A = P * pow(1 + r / n, n * t);
25
26
27      printf("\n-----\n");
28      printf("Початкова сума інвестиції (P): %.2f грн\n", P);
29      printf("Річна процентна ставка (r): %.2f%%\n", r * 100);
30      printf("Кількість нарахувань відсотків на рік (n): %d\n", n);
31      printf("Кількість років інвестиції (t): %d\n", t);
32      printf("-----\n");
33      printf("Загальна сума інвестиції (A): %.2f грн\n", A);
34      printf("Сума заробітку: %.2f грн\n", A - P);
35
36      return 0;
37  }
```

ОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ КОНСОЛЬ ОТЛАДКИ ТЕРМИНАЛ ПОРТЫ

```
Річна процентна ставка (r): 16.00%
Кількість нарахувань відсотків на рік (n): 4
Кількість років інвестиції (t): 2
-----
Загальна сума інвестиції (A): 13685.69 грн
Сума заробітку: 3685.69 грн
C:\Users\Eugene\Downloads\output> □
```

0 0 0 ▶ Compile & Run 0 Compile 0 Debug

Завдання №2 Self Practice Task:

Завдання з сайту Algotester.com

```

: > Users > Eugene > Desktop > labs > Epic-1 > Epic1algpecvivo.cpp > main
1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5  int main() {
6
7      int min, n ;
8      long max;
9      cin >> n;
10     min = 0;
11     max = 0;
12     vector<int>A(n);
13
14     for(int i = 0;i < n ;i++)
15         cin >>A[i] ;
16
17     for(int i = 0;i < n ;i++)
18         if(A[i]>0){
19             min = A[i] - 1;
20             max = max + min;
21         }
22     cout << max;
23
24
25     return 0;
26 }
27

```

Посилання на мою гілку де знаходяться ці програми:
<https://github.com/Eugenie2708/Epic1>

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

Завдання №1 Practice Task

```
PS C:\Users\Eugene\Desktop\labs\Epic-1\output> & .\'Epic1deposit.exe'  
Введіть початкову суму інвестиції (P): 10000  
Введіть річну процентну ставку (r): 0.14  
Введіть кількість нарахувань відсотків на рік (n): 4  
Введіть кількість років (t): 1  
  
-----  
Початкова сума інвестиції (P): 10000.00 грн  
Річна процентна ставка (r): 14.00%  
Кількість нарахувань відсотків на рік (n): 4  
Кількість років інвестиції (t): 1  
-----  
Загальна сума інвестиції (A): 11475.23 грн  
Сума заробітку: 1475.23 грн  
PS C:\Users\Eugene\Desktop\labs\Epic-1\output> 100
```

На це завдання я витратив близько години, бо ознайомлювався з новими для мене функціями.

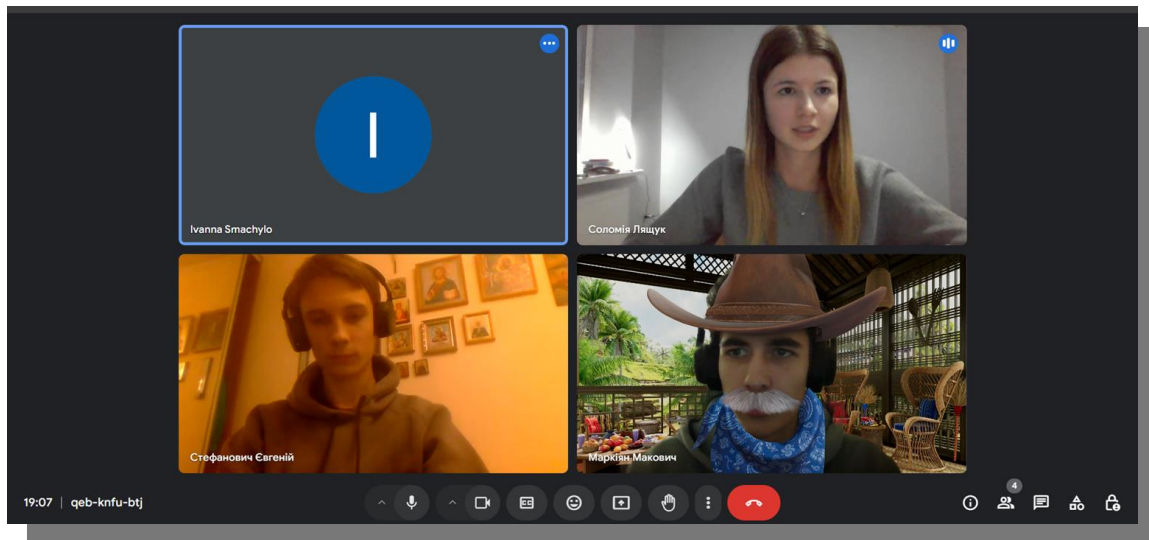
Завдання №2 Self Practice Task

```
PS C:\Users\Eugene\Desktop\labs\Epic-1\output> & .\'Epic1algpecvivo.exe'  
5  
1 2 3 4 55  
60  
PS C:\Users\Eugene\Desktop\labs\Epic-1\output>
```

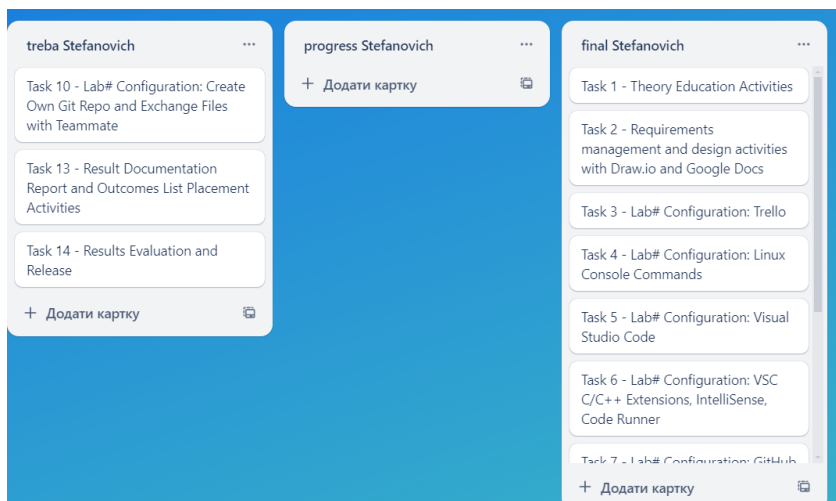
Виконуючи цю задачу на Algotester.com, я витратив близько двох годин, оскільки під час виконання виникала помилка, яку я довгий час не міг помітити.

6. Кооперація з командою:

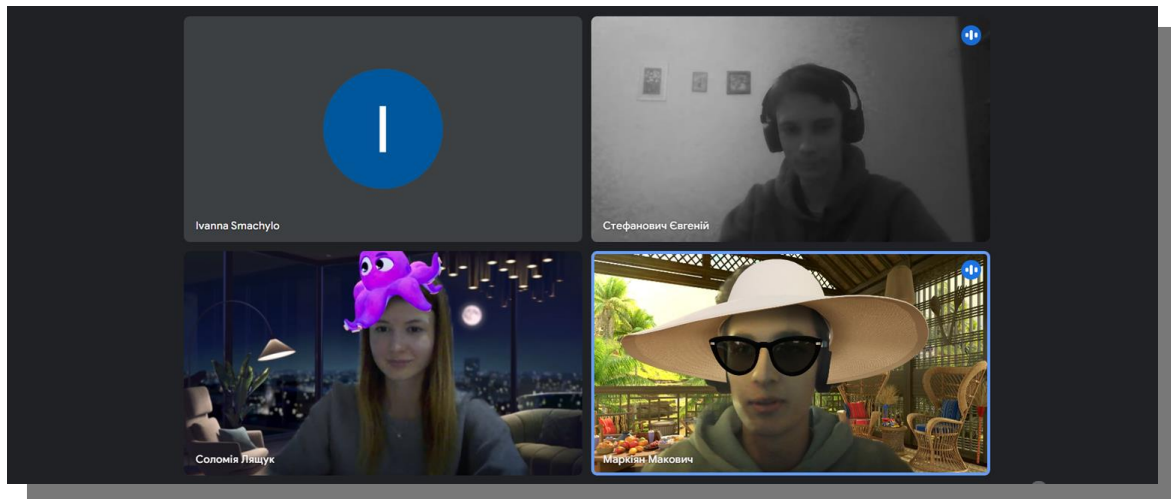
Зустріч з командою №1:



Прогрес Trello:



Зустріч з командою №2:



Висновки:

Працюючи над **Еріс-1** я налаштував середовище для програмування, дізнався про Linux команди та освоїв частину з них. Створив дошку в Trello. Дізнався про сайт Draw.io, та навчився створювати блок-схеми. Зареєструвався на сайті Algotester.com, та виконав задачу. Налаштував Git та розібрався з потрібними мені командами. Створив свою гілку та Pull Request на сайті GtiHub. Навчився проводити калькуляції над числами в двійковій системі, та розвинув свої SoftSkills працюючи в команді з одногрупниками.