

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(ла):

Студент групи ШІ-12

Бобровицький Олександр Сергійович

Львів 2024

Тема роботи:

“Програмування: алгоритм, програма, код. Системи числення. Двійкова система числення. Розробка та середовище розробки програми.”

Мета роботи:

Навчитися працювати з ресурсами які знадобляться в подальшому навчанні. Налаштувати середовище розробки. Ознайомитися з інструментами які допомагають оптимізувати роботу в цьому середовищі. На практиці пропрацювати процес створення власної програми.

Теоретичні відомості:

- 1) Теоретичні відомості з переліком важливих тем:
 - Тема №1: Draw.io and FlowCharts
 - Тема №2: Trello
 - Тема №3: Linux Console Commands.
 - Тема №4: VSC C/C++ Extensions, IntelliSense, Code Runner
 - Тема №5: GitHub/ Git
 - Тема №6: Algotester
 - Тема №7: Binary Calculations
 - Тема №8: C++ Basics
- 2) Індивідуальний план опрацювання теорії:
 - Тема №1: Draw.io and FlowCharts
 - Джерела Інформації
 - Відео <https://www.youtube.com/watch?v=w3zm-wbmlpc>.
 - Що опрацьовано:
 - Створення власних FlowCharts за допомогою Draw.io
 - Статус: Ознайомлений
 - Початок опрацювання теми: 28.09
 - Звершення опрацювання теми: 02.10
 - Тема №2: : Trello.
 - Джерела Інформації:
 - Відео <https://www.youtube.com/watch?v=geRKHfzTxNY>
 - Що опрацьовано:
 - Ознайомлення з дошкою команди в Trello
 - Робота з дошкою команди
 - Зміна статусу тасків залежно від їх дійсного стану опрацювання
 - Статус: Ознайомлений
 - Початок опрацювання теми: 16.09
 - Звершення опрацювання теми: 16.09
 - Тема №3: : Linux Console Commands.
 - Джерела Інформації:
 - Відео <https://www.freecodecamp.org/news/the-linux-commands-handbook/#heading-the-linux-man-command>.
 - Що опрацьовано:

- Створення директорій та робота з ними за допомогою терміналу.
 - Виконання різноманітних операцій за допомогою цих команд
 - Статус: Ознайомлений
 - Початок опрацювання теми: 16.09
 - Звершення опрацювання теми: 04.10
- Тема №4 VSC C/C++ Extensions, IntelliSense, Code Runner.
 - Джерела Інформації:
 - <https://code.visualstudio.com/docs>
 - <https://code.visualstudio.com/docs/cpp/config-mingw>.
 - Що опрацьовано:
 - Встановив VSC та розширення для нього
 - Встановив та ознайомився на практиці з GCC C++ compiler (g++) and GDB debugger за допомогою MinGW-w64.
 - Статус: Ознайомлений
 - Початок опрацювання теми: 01.09
 - Звершення опрацювання теми: 20.09
- Тема №5 GitHub/ Git.
 - Джерела Інформації:
 - Відео <https://www.youtube.com/watch?v=RGOj5yH7evk&t=70s>.
 - Відео <https://www.youtube.com/watch?v=a0CftUjkrbs&t=2364s>
 - Що опрацьовано:
 - Ознайомився з командами git
 - Створив власний віддалений репозиторій завдяки Github
 - Вивів файли власних проєктів на віддалений репозиторій.
 - Статус: Ознайомлений
 - Початок опрацювання теми: 16.09
 - Звершення опрацювання теми: 04.10
- Тема №6 Algotester.
 - Джерела Інформації:
 - <https://algotester.com/uk/Home/Help>.
 - Що опрацьовано:
 - Зареєстрував власний аккаунт
 - Почав виконувати вільно доступні завдання
 - Статус: Ознайомлений
 - Початок опрацювання теми: 01.09
 - Звершення опрацювання теми: 03.09
- Тема №7 Binary Calculations.
 - Джерела Інформації:
 - <https://www.youtube.com/watch?v=dBRBPKq3IJY>.
 - Лекції з основ програмування
 - Що опрацьовано:
 - Навчився виконувати базові арифметичні дії в двійковій системі.

- Статус: Ознайомлений
- Початок опрацювання теми: 08.09
- Звершення опрацювання теми: 28.09
- Тема №8 C++ Basics.
 - Джерела Інформації:
 - <https://ravesli.com/>
 - <https://acode.com.ua/>
 - Лекції з основ програмування
 - Що опрацьовано:
 - Створення власних програм на C++
 - Запуск цих програм
 - Статус: Ознайомлений
 - Початок опрацювання теми: 01.09
 - Звершення опрацювання теми: 13.09

Виконання роботи:

1. Опрацювання завдання та вимог до програм та середовища:

Завдання №1 Обчислення складних відсотків за депозитом

Варіант завдання

- Деталі завдання :
 - Обчислити складні відсотки для депозиту, який був відкритий в банку на певний період часу під фіксовані відсотки з різними варіантами виплати відсотків.
- Важливі деталі для врахування в імплементації програми :
 - Використати функції *scanf* та *printf* для зчитування і форматування вводу/виводу;
 - В кінці програма має вивести повну інформацію про вкладені кошти, загальну суму інвестиції і суму самого заробітку.

Завдання №2 Власний код по темі з задач алготестеру

- 0022(Верховна Рада)
- В останніх виборах до Верховної Ради взяли участь n партій. Вам відома кількість голосів, що отримала кожна з них. Допоможіть народові мінімізувати загальну кількість депутатських місць..
- Важливі деталі для врахування в імплементації програми :
 - Вхідні дані
 - У першому рядку задано одне натуральне число n — кількість партій.
 - У другому рядку задано n натуральних чисел a_i — кількість голосів, що отримала i -та партія на останніх виборах.
 - Вихідні дані
 - У єдиному рядку виведіть одне ціле число — мінімальну загальну кількість депутатських місць у парламенті.
 - Обмеження
 - $1 \leq n \leq 1000$,
 - $1 \leq a_i \leq 10^9$.

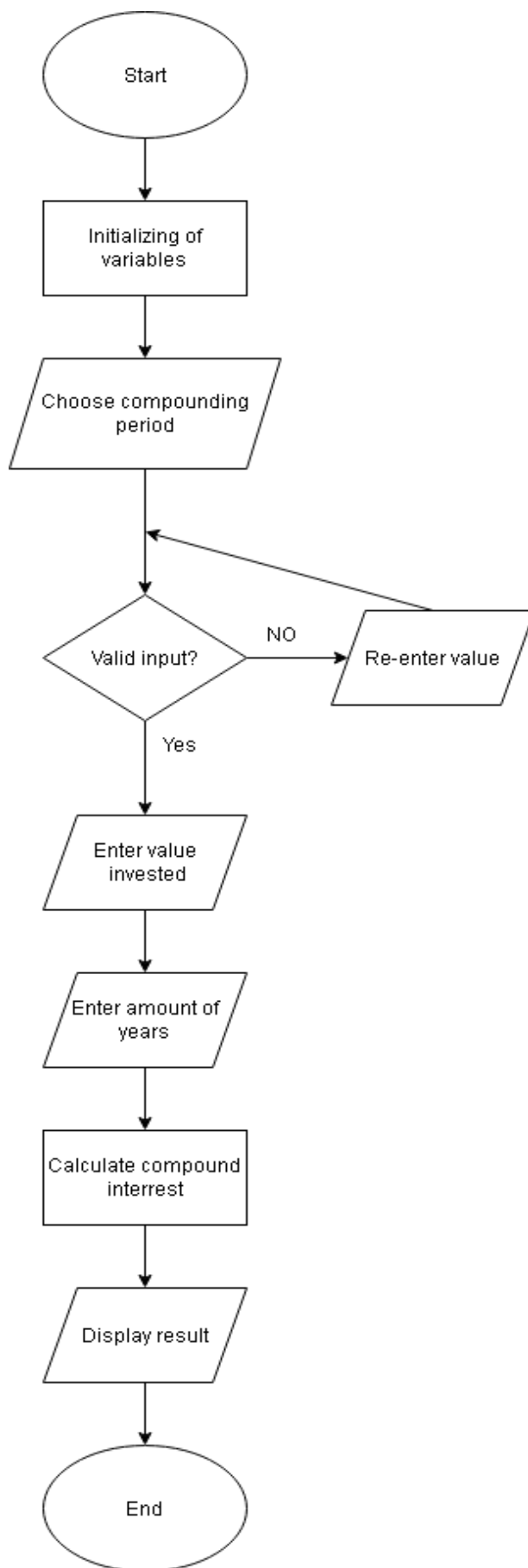
Завдання №3 Завдання на калькуляції в двійковій системі

- 1 Згенерувати в рандомайзері десяткове число y від 20 до 99
- 2 Згенерувати в рандомайзері десяткове число x від 20 до 99
- 3 Перевести y 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 двійкове число
- 8 Більше двійкове число помножити на менше двійкове число
- 9 Згенерувати в рандомайзері десяткове число k від 20 до 99
- 10 Перевести k у 16-ву систему числення

2. Дизайн та планована оцінка часу виконання завдань:

Програма №1 Обчислення складних відсотків за депозитом.

- Блок-схема

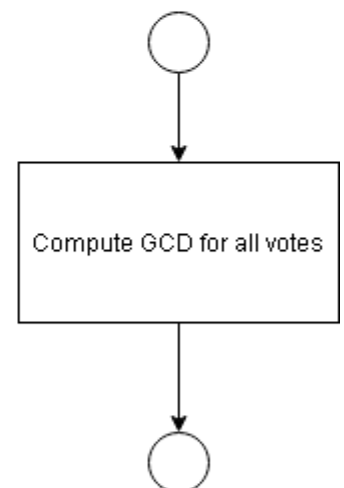
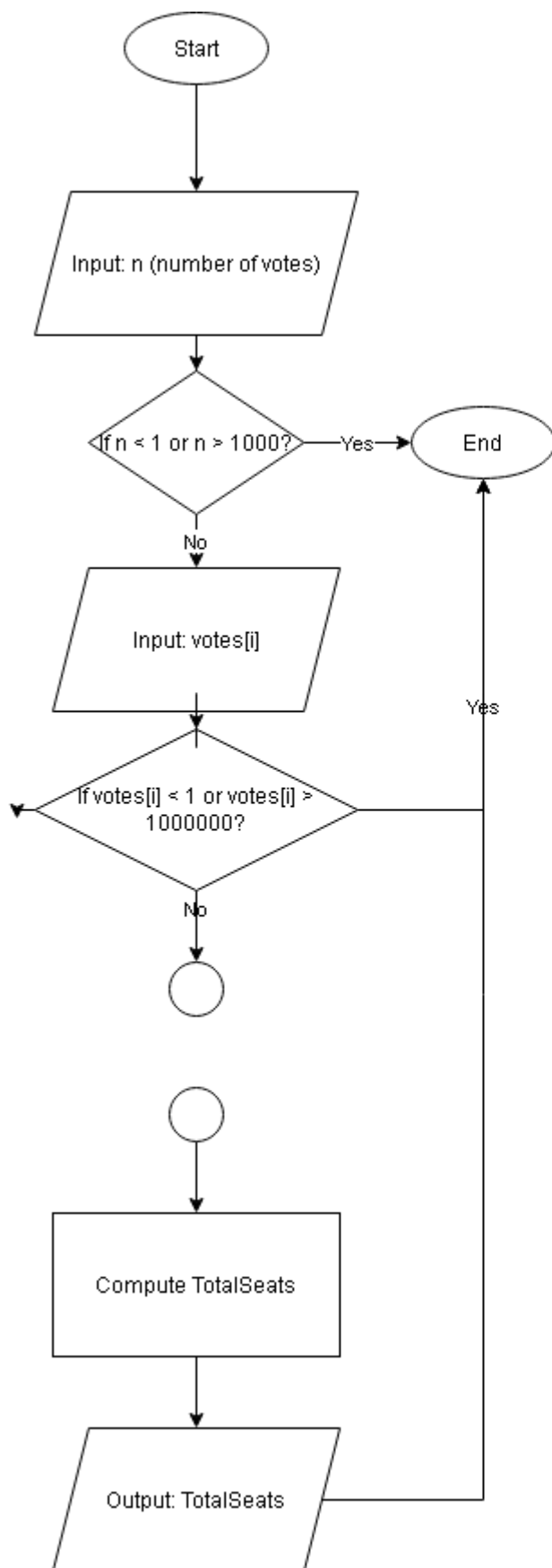


- Планований час на реалізацію(1год.)
- Важливі деталі для врахування в імплементації

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

Програма №2 Власний код по темі з задач алготестеру.(Задача 0022 Верховна Рада)

- Блок-схема



- Планований час на реалізацію(1год.)
- Важливі деталі для врахування в імплементації
 - Вхідні дані
 - У першому рядку задано одне натуральне число n — кількість партій.

- У другому рядку задано n натуральних чисел a_i — кількість голосів, що отримала i -та партія на останніх виборах.
- Вихідні дані
 - У єдиному рядку виведіть одне ціле число — мінімальну загальну кількість депутатських місць у парламенті.
- Обмеження
 - $1 \leq n \leq 1000$,
 - $1 \leq a_i \leq 10^9$.

Програма №3 Завдання на калькуляції в двійковій системі.

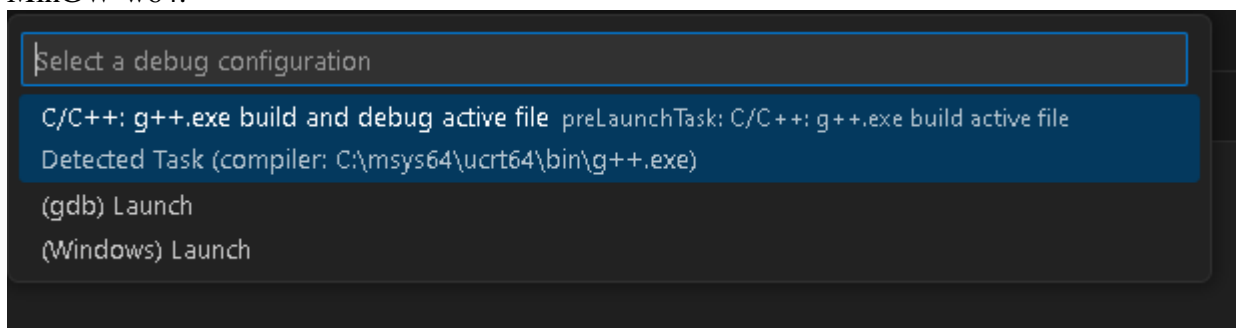
- 1 Згенерувати в рандомайзері десяткове число y від 20 до 99
- 2 Згенерувати в рандомайзері десяткове число x від 20 до 99
- 3 Перевести y у двійкову систему числення
- 4 Перевести x у двійкову систему числення
- 5 Додати два двійкових числа x та y
- 6 Відняти від більшого двійкового числа менше двійкове число
- 7 Більше двійкове число поділити на менше двійкове число
- 8 Більше двійкове число помножити на менше двійкове число
- 9 Згенерувати в рандомайзері десяткове число k від 20 до 99
- 10 Перевести k у 16-ву систему числення

Планований час на реалізацію(30хв)

3. Конфігурація середовища до виконання завдань:

Завдання №1

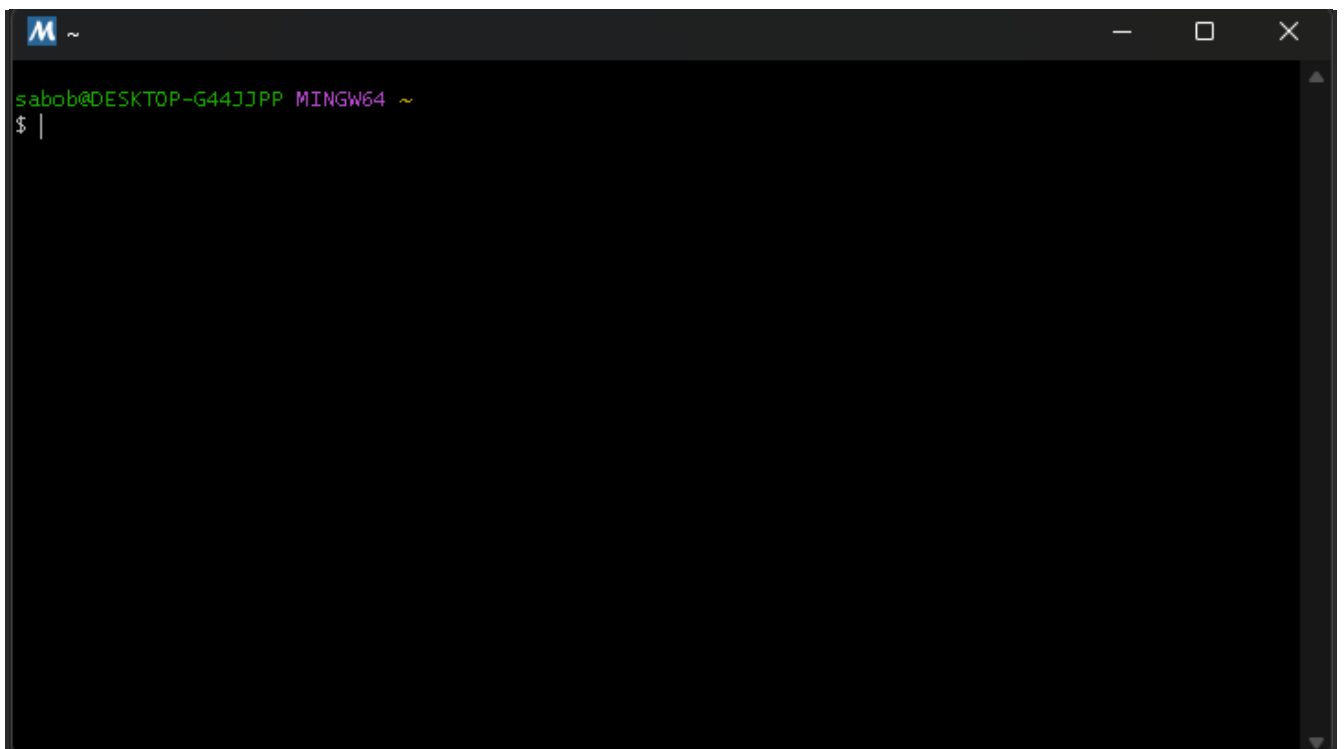
- Встановлення VSCode та GCC C++ compiler (g++) і GDB debugger за допомогою MinGW-w64.



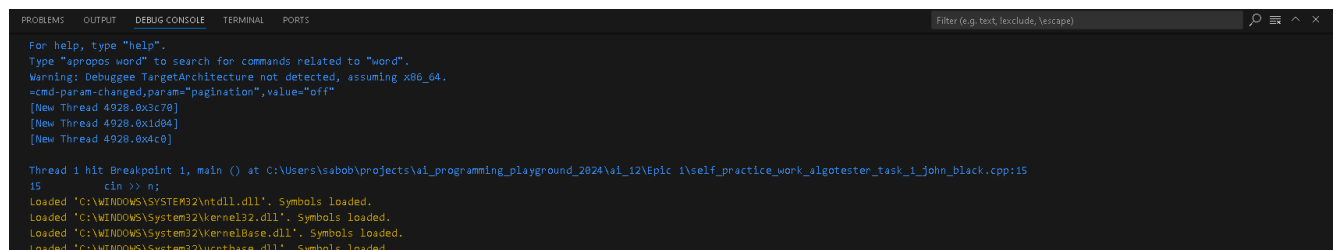
Опції зборки та налагодження програми

```
1 #include <iostream>
2 #include <vector>
3 #include <numeric>
4
5 using namespace std;
6
7 int gcd(int a, int b)
8 {
9     return b == 0 ? a : gcd(b, a % b);
10 }
11
12 int main()
13 {
14     int n;
15     cin >> n;
16     if (1 > n || n > 1000)
17     {
18         cout << "Invalid value" << endl;
19         return 1;
20     }
21
22     vector<int> votes(n);
23
24     for (int i = 0; i < n; ++i)
25     {
26         cin >> votes[i];
27         if (1 > votes[i] || votes[i] > 1000000)
28         {
29             cout << "Invalid value" << endl;
30             return 1;
31         }
32     }
33
34     int GreatestCommonDevide = votes[0];
35
36     for (int i = 1; i < n; ++i)
37     {
38         GreatestCommonDevide = gcd(GreatestCommonDevide, votes[i]);
39     }
40
41     int TotalSeats;
42     for (int i = 0; i < n; ++i)
43     {
44         TotalSeats += votes[i] / GreatestCommonDevide;
45     }
46
47     cout << TotalSeats << endl;
48
49     return 0;
50 }
```

VSCode



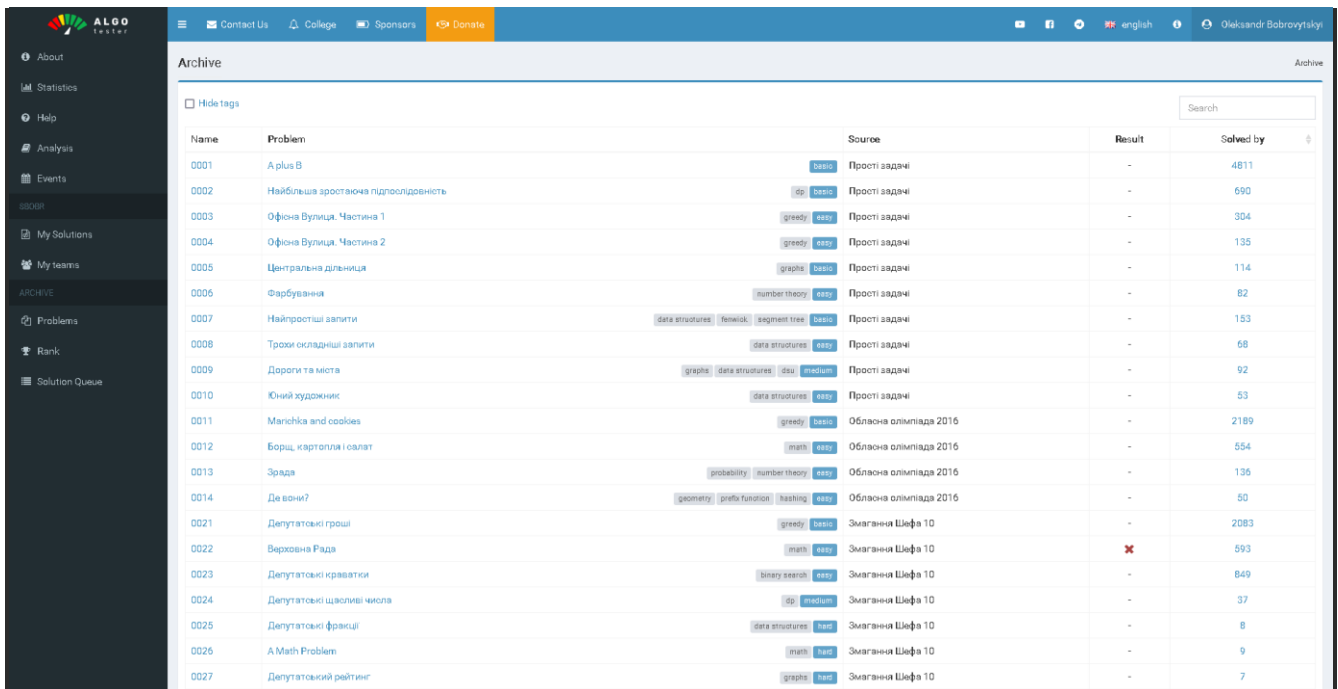
MINGw64 terminal



Debug console

Завдання №2

Використання того ж самого середовища що й для першого завдання + реєстрація акаунта на Алготестері.



Name	Problem	Source	Result	Solved by
0001	A plus B	basics	-	4811
0002	Найбільша зростаюча підпоследовність	dp	-	690
0003	Офісна Вулиця, Частина 1	greedy	-	304
0004	Офісна Вулиця, Частина 2	greedy	-	135
0005	Центральна ділянка	graphs	-	114
0006	Фарбування	number theory	-	82
0007	Найпростіші запити	data structures	-	153
0008	Трохи складніші запити	data structures	-	68
0009	Дороги та міста	graphs	-	92
0010	Юний художник	data structures	-	53
0011	Marichka and cookies	greedy	-	2189
0012	Борщ, картопля і салат	math	-	554
0013	Зрада	probability	-	136
0014	Де вони?	geometry	-	50
0021	Депутатські гроші	greedy	-	2083
0022	Верховна Рада	math	-	593
0023	Депутатські краватки	binary search	-	849
0024	Депутатські щасливі числа	dp	-	37
0025	Депутатські фракції	data structures	-	8
0026	A Math Problem	math	-	9
0027	Депутатський рейтинг	graphs	-	7

Акаунт на Алготестері

4. Код програм з посиланням на зовнішні ресурси:

Завдання №1 Деталі по програмі

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

Посилання на зовнішні ресурси

- <https://www.programiz.com/c-programming/c-input-output>
- <https://www.programiz.com/c-programming/c-switch-case-statement>
- https://en.wikipedia.org/wiki/C_data_types

Посилання на файл програми у пул-запиті GitHub https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/34/commits/e877ea6226792d1d201fb325efaac9e66bdec827#diff-2d2025bf05d28c131c0eae5396175eff3171e920fe5fc4c8e2048ecd8d5bb79e

Вставка з кодом з підписами до вставки.

```
#include <stdio.h>
#include <math.h>

int main()
{
    float AnnualInterest = 0.05;
    short int NumOfCompoundingPeriods;
    double AmountInvested;
    int AmountOfYears;
    short int temp;

    printf("Choose the number corresponding to the compounding period:\n");
```

```

printf("Every month = 1\n");
printf("Every quarter = 2\n");
printf("Every year = 3\n");

do
{
    scanf("%d", &temp);
    switch (temp)
    {
        case 1:
            NumOfCompoundingPeriods = 12;
            break;
        case 2:
            NumOfCompoundingPeriods = 4;
            break;
        case 3:
            NumOfCompoundingPeriods = 1;
            break;
        default:
            printf("Invalid value, try again.\n");
    }
} while (!(temp == 1 || temp == 2 || temp == 3));

printf("Enter the amount that you want to put in = ");
scanf("%lf", &AmountInvested);

printf("Enter the amount of time(in whole years) for which money will be invested = ");
scanf("%d", &AmountOfYears);

double InterestRate = 1 + (AnnualInterest/NumOfCompoundingPeriods);

int TotalNumOfCompoundingPeriods = AmountOfYears * NumOfCompoundingPeriods ;

double FutureAmountOfMoney = AmountInvested * pow(InterestRate , TotalNumOfCompoundingPeriods);

printf("Initial amount invested = %lf\n", AmountInvested );
printf("Initial amount of money plus interest = %lf\n", FutureAmountOfMoney );
printf("Clear profit = %lf\n", FutureAmountOfMoney - AmountInvested );

return 0;
}

```

Код для обчислення складних відсотків для депозиту, який був відкритий в банку на певний період часу під фіксовані відсотки з різними варіантами виплати відсотків. Для завдання №1.

Завдання №2 Деталі по програмі

- В останніх виборах до Верховної Ради взяли участь n партій. Вам відома кількість голосів, що отримала кожна з них, відношення кількості голосів до кількості місць у Верховній Раді має бути однаковим для усіх партій. Допоможіть народові мінімізувати загальну кількість депутатських місць.
- У першому рядку задано одне натуральне число n — кількість партій.
- У другому рядку задано n натуральних чисел a_i — кількість голосів, що отримала i -та партія на останніх виборах.
- У єдиному рядку виведіть одне ціле число — мінімальну загальну кількість депутатських місць у парламенті.

Посилання на зовнішні ресурси

- <https://en.cppreference.com/w/cpp/numeric/gcd>

- https://www.w3schools.com/cpp/cpp_conditions_shorthand.asp
- <https://ravesli.com/urok-95-vvedenie-v-std-vector-vektory/>

Посилання на файл програми у пул-запиті GitHub https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/34/commits/9b4a6d48764d100dc2006fbdfae9127155c4c51#diff-f37868a6d85a45905b6b45eab0d190df5108d25d3b5694f915a4db1be227e83a

Вставка з кодом з підписами до вставки.

```
#include <iostream>
#include <vector>
#include <numeric>

using namespace std;

int gcd(int a, int b)
{
    return b == 0 ? a : gcd(b, a % b);
}

int main()
{
    int n;
    cin >> n;
    if (1 > n || n > 1000)
    {
        cout << "invalid value" << endl;
        return 1;
    }

    vector<int> votes(n);

    for (int i = 0; i < n; ++i)
    {
        cin >> votes[i];
        if (1 > votes[i] || votes[i] > 1000000)
        {
            cout << "invalid value" << endl;
            return 1;
        }
    }

    int GreatestCommonDeviver = votes[0];

    for (int i = 1; i < n; ++i)
    {
        GreatestCommonDeviver = gcd(GreatestCommonDeviver, votes[i]);
    }

    int TotalSeats = 0;
    for (int i = 0; i < n; ++i)
    {
        TotalSeats += votes[i] / GreatestCommonDeviver;
    }

    cout << TotalSeats << endl;

    return 0;
}
```

Код для обчислення мінімальної загальної кількості депутатських місць у парламенті. Для завдання №2

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

Завдання №1 Деталі по виконанню і тестуванню програми

```
Choose the number corresponding to the compounding period:
Every month = 1
Every quarter = 2
Every year = 3
2
Enter the amount that you want to put in = 10000
Enter the amount of time(in whole years) for which money will be invested = 15
Initial amount invested = 10000.000000
Initial amount of money plus interest = 21071.873012
Clear profit = 11071.873012
```

Блок №1. Результат виконання програми

```
The program 'C:\Users\sabob\projects\ai_programming_playground_2024\ai_12\Epic
1\practice_work_task_1_oleksandr_bobrovytskyi.exe' has exited with code 0 (0x00000000).
```

Блок №2. Результат тестування програми в Debug консолі.

Час затрачений на виконання завдання(1.5 год)

Завдання №2 Деталі по виконанню і тестуванню програми

```
4
1000
4000
5500
620
556
```

Блок №1. Результат виконання програми(556- відповідь)

```
The program 'C:\Users\sabob\projects\ai_programming_playground_2024\ai_12\Epic
1\self_practice_work_algotester_task_1_oleksandr_bobrovytskyi.exe' has exited with code 0 (0x00000000).
```

Блок №2. Результат тестування програми в Debug консолі.

Час затрачений на виконання завдання(2. год)

Завдання №3 Деталі по виконанню

9) ~~K_{16}~~ $K_{10} = 39_{10}$

10) $K_{16} = 24_{16}$ ~~16~~

$$\begin{array}{r|l} \cancel{39} & 16 \\ \hline 16 & \cancel{1} \\ \hline 23 & \end{array}$$

$$\begin{array}{r|l} 39 & 16 \\ \hline 32 & 24 \\ \hline 7 & \end{array}$$

$$1) y_2 = 32$$

$$2) x_2 = 80$$

$$3) y_2 = \cancel{00100000} 00100000_2$$

$$4) x_2 = 01010000_2$$

$$5) x_2 + y_2 = 00100000 + 01010000 =$$

$$= 01110000$$

$$\begin{array}{r} \cancel{00100000} + 00100000 \\ 01010000 \\ \hline 01110000 \end{array}$$

$$6) x_2 - y_2 = 01010000 - 00100000 =$$

$$= 00110000$$

$$\begin{array}{r} -1 \\ 01010000 \\ - 00100000 \\ \hline 00110000 \end{array}$$

$$7) \frac{x_2}{y_2} = 10$$

$$\begin{array}{r} \cancel{00100000} \cancel{00100000} \\ \cancel{00100000} \cancel{00100000} \\ \cancel{00100000} \cancel{00100000} \\ \cancel{00100000} \cancel{00100000} \\ \hline 00100000 \end{array}$$

$$\begin{array}{r} 00100000 \overline{) 00000010} \\ \underline{00100000} \\ 00000000 \end{array}$$

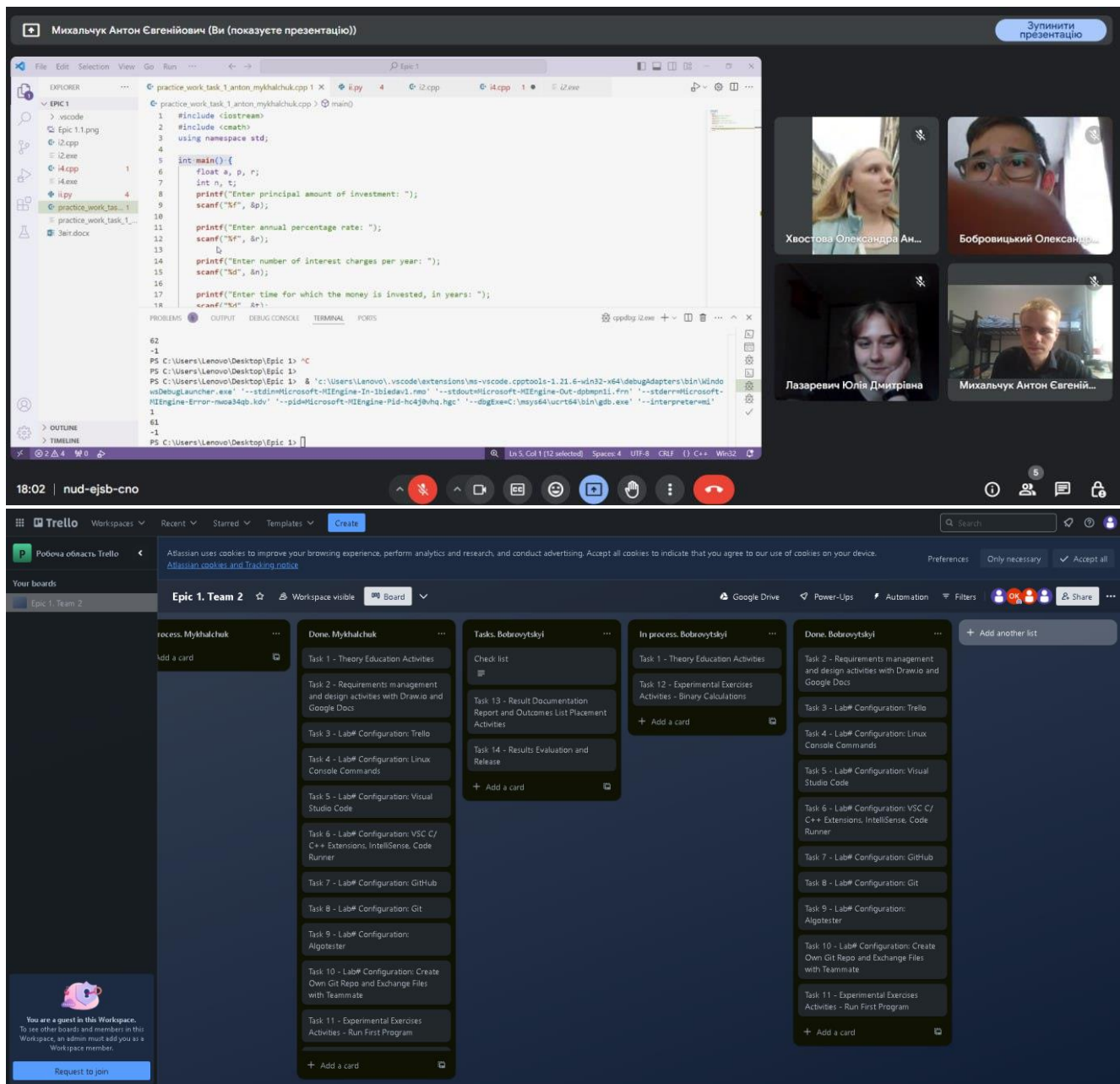
$$8) x_2 \cdot y_2 = 01010000 \cdot 00100000 =$$

$$\begin{array}{r} \times \quad 01010000 \\ \quad 100000 \\ \hline 000000000 \\ 0000000000 \\ 00000000000 \\ 000000000000 \\ \hline 010100000000 \end{array}$$

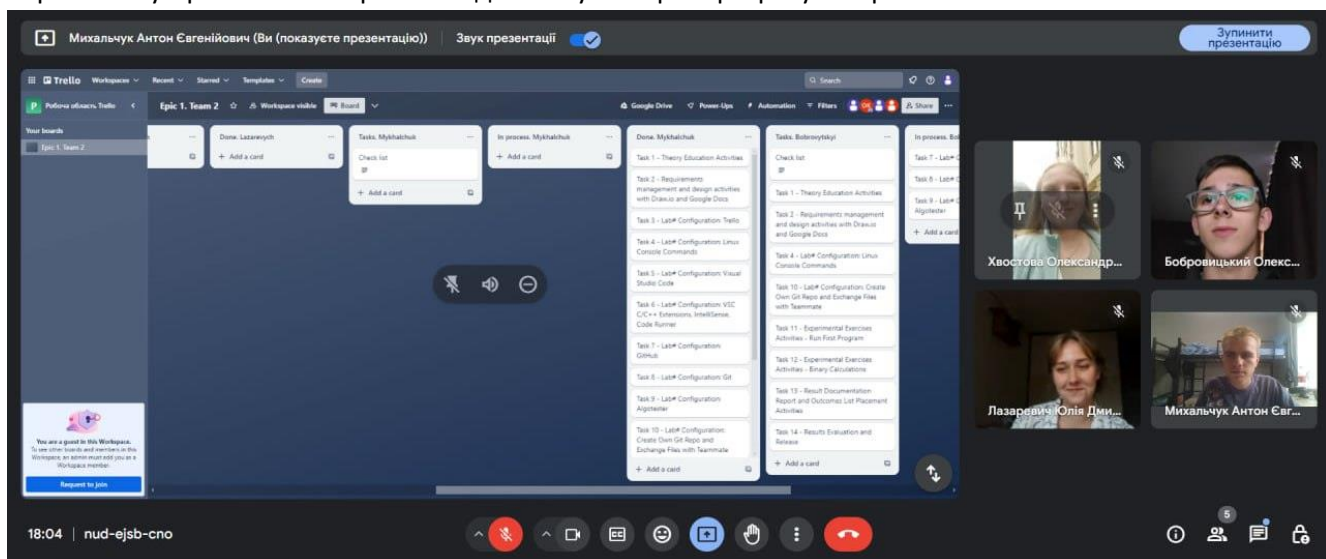
$$= 101000000000$$

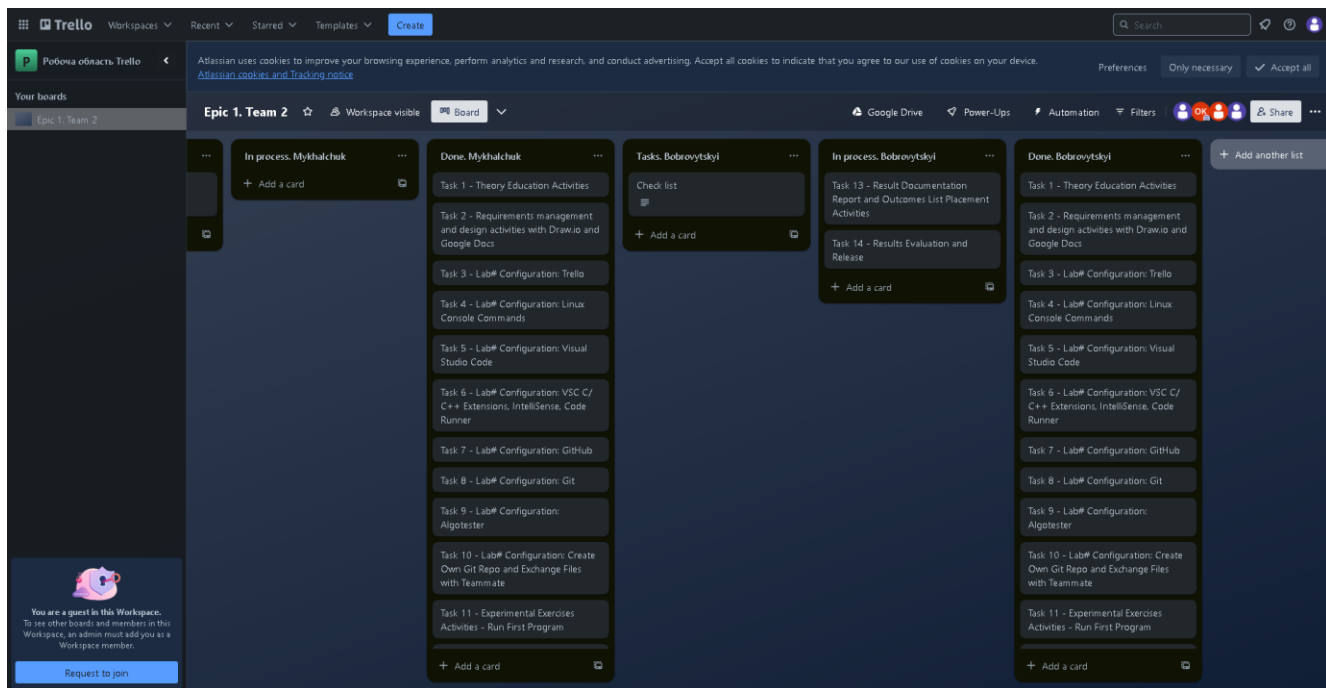
6. Кооперація з командою:

- Скрін з 1-ї зустрічі по обговоренню задач Епіку та Скрін прогресу по Тrello

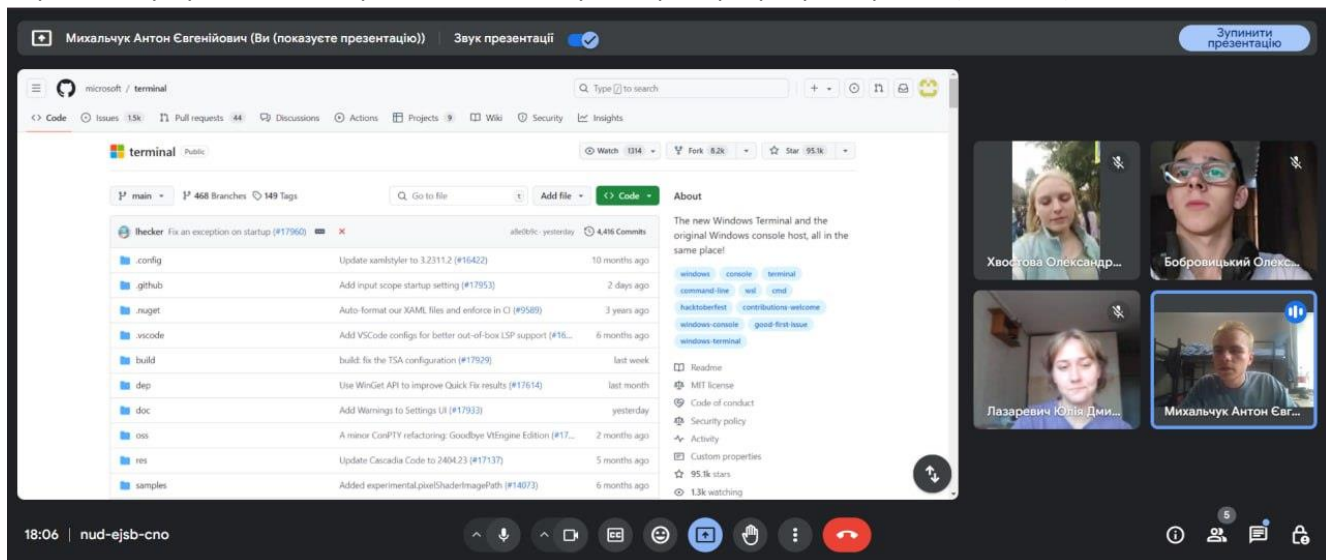


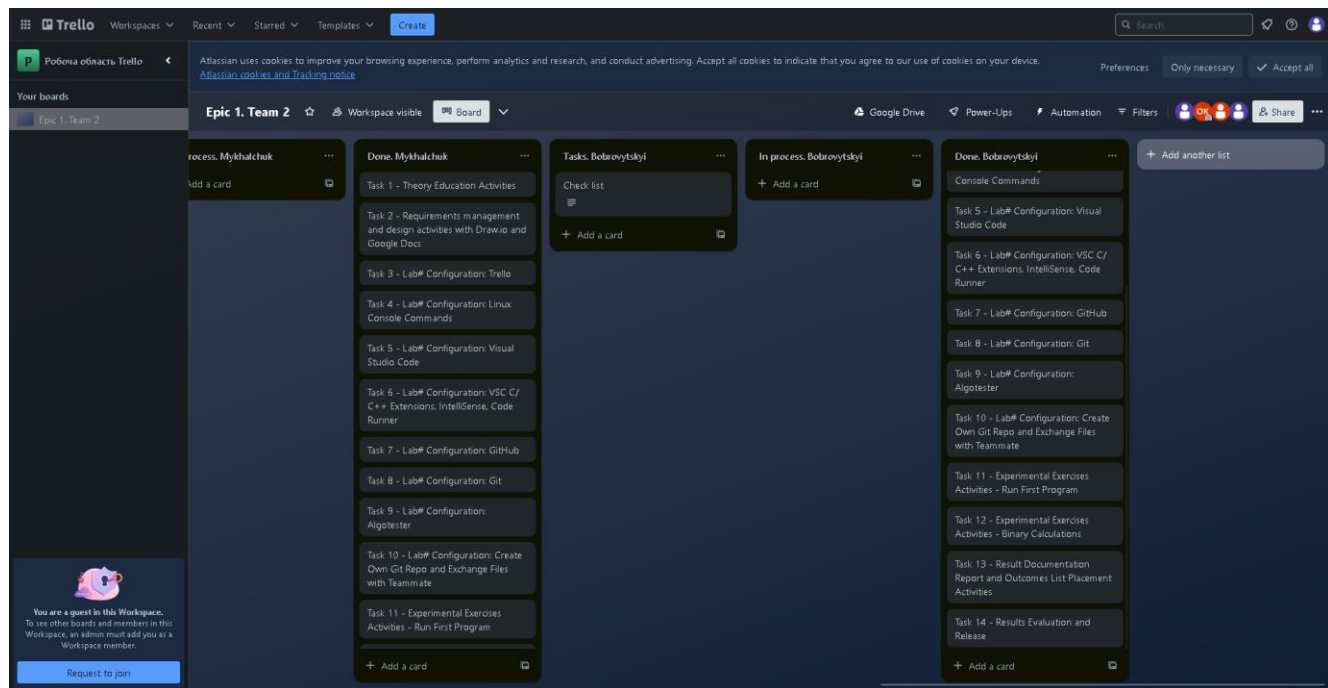
- Скрін з 2-ї зустрічі по обговоренню задач Епіку та Скрін прогресу по Тrello



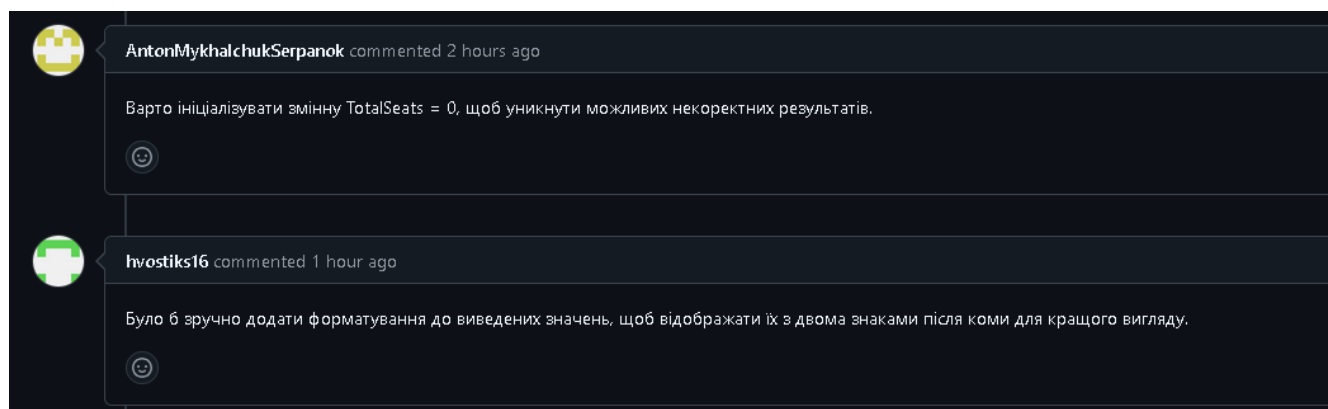


- Скрін з 3-ї зустрічі по обговоренню задач Епіку та Скрін прогресу по Тrello (опційно)





- Скрін з 2-му коментарями від учасників команди на пул реквесті з Ревю Роботи



Висновки:

Я навчився працювати з ресурсами які можуть знадобитися в моєму подальшому навчанні. Налаштував середовище розробки. Ознайомився з інструментами які допомагають оптимізувати роботу в цьому середовищі. На практиці пропрацював процес створення власної програми.