

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-11
Бубельник Юрій Олегович

Львів 2024

Тема роботи:

Налаштування робочого середовища (Visual Studio Code, GitHub, MSYS2, Trello, Algotester). Системи числення. Програмування на C++. Створення блок-схем.

Мета роботи:

Ознайомитись з git та його командами, принципами роботи з GitHub та створення блок-схем. Конфігурація VS Code та налаштування компілятора для C++. Використання на практиці знань з систем числення. Налаштування дошки завдань та робота в команді.

Теоретичні відомості:

Тема №1.

Git tutorial for Beginners:

<https://www.youtube.com/watch?v=8JJ101D3knE>

Тема №2

Уроки програмування на C++:

<https://acode.com.ua/uroki-po-cpp/>

Тема №3

How To Use TRELLO for Beginners:

<https://www.youtube.com/watch?v=6drUzoeHZkg>

Тема №4

Debugging C++ Program in Visual Studio Code:

https://www.youtube.com/watch?v=2VokW_Jt0oM&t=610s

Тема №5

Просто про Системи числення: двійкова, шістнадцяткова, десяткова:

<https://www.youtube.com/watch?v=yGmVLDenVpE>

Тема №6

Арифметичні дії в різних системах числення

<https://studfile.net/preview/1582350/page:6/>

Тема №7

Складання блок-схем:

Дізнався від старшокурсника.

Індивідуальний план опрацювання теорії:

Тема №1.

Ознайомився з git та його командами.
Витратив 50 хвилин.

Тема №2.
Ознайомився з базовим синтаксисом C++.
Витратив 4-5 годин.

Тема №3.
Ознайомився з Trello.
Витратив 15хвилин.

Тема №4.
Налаштував VS Code для C++.
Витратив 30 хвилин.

Тема №5-6.
Ознайомився з системами числення та операціями з ними.
Витратив 20-30 хвилин.

Тема №7.
Ознайомився з інтерфейсом Draw.io та базовими блок-схемами.

Виконання роботи:

1) Опрацювання завдання та вимог до програм та середовища

Завдання №1:

Обчислення складних відсотків за депозитом:

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

Вимоги:

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:

Депутатські гроші:

Часто-густо громадяни намагаються з'ясувати, наскільки багатими є депутати. Дехто вірить, що матеріальні статки окремих депутатів є необмеженими.

Тож уявіть собі депутата, у якого є необмежена кількість купюр усіх номіналів (1, 2, 5, 10, 20, 50, 100, 200 та 500 гривень). Він хоче придбати подарунок для своєї тещі, що коштує np гривень.

Незважаючи на те, що наш герой-олігарх є нескінченно багатим, він також є нескінченно скупим та педантичним. Саме тому він хоче оплатити покупку готівкою без решти. Яка мінімальна кількість купюр йому для цього знадобиться?

Вимоги:

У єдиному рядку задано одне натуральне число n — вартість подарунку.

$$1 \leq n \leq 10^9$$

Завдання №3:

-Перевести число з десяткової системи числення у двійкову та подати це число з мінусом.

Завдання №4:

Завдання на калькуляції в двійковій системі:

- Операції над числами у двійковій системі числення.

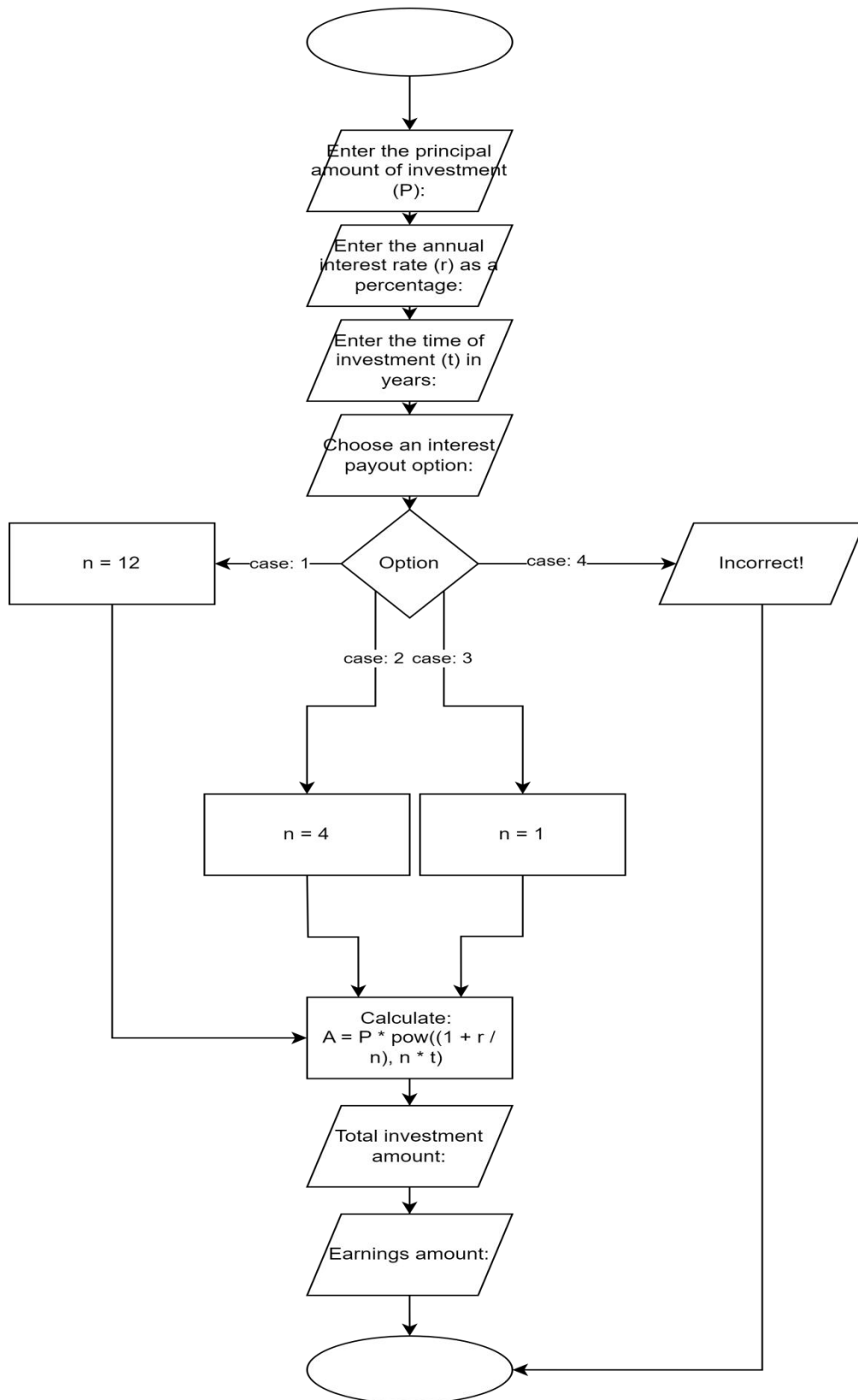
Вимоги:

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:

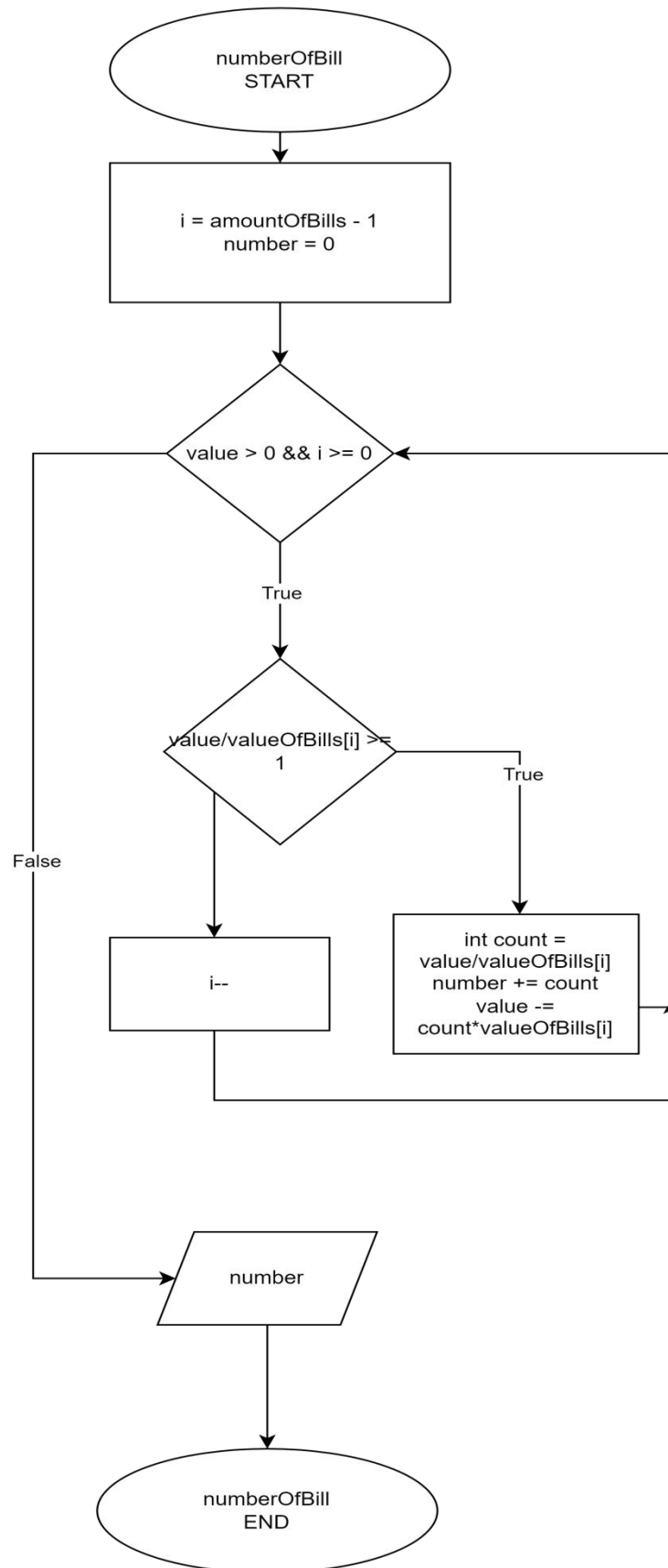
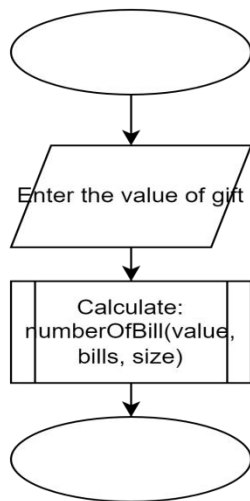
Завдання №1:

Планований час виконання: 30-35 хвилин.



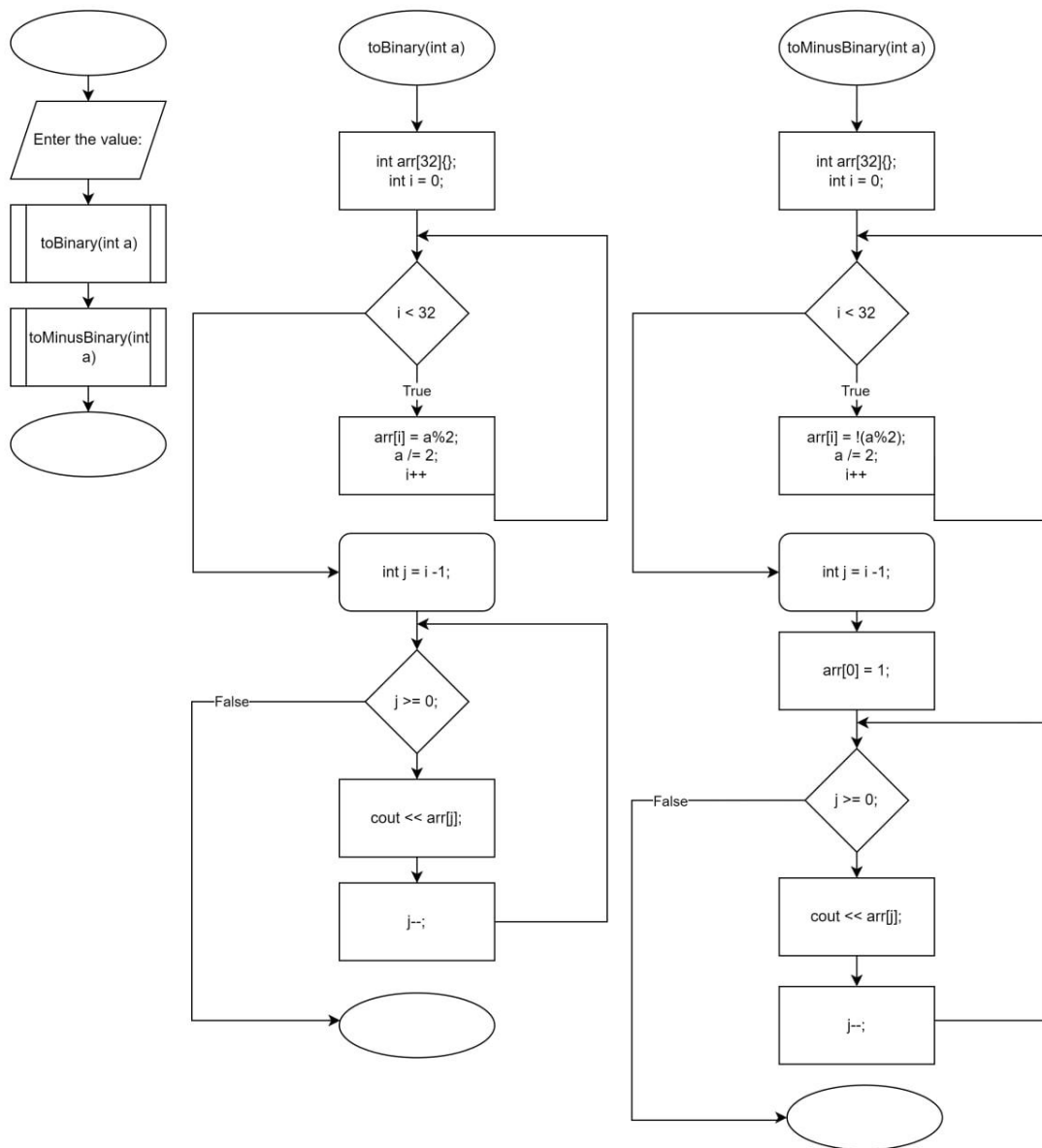
Завдання №2:

Планований час виконання: 25 хвилин.



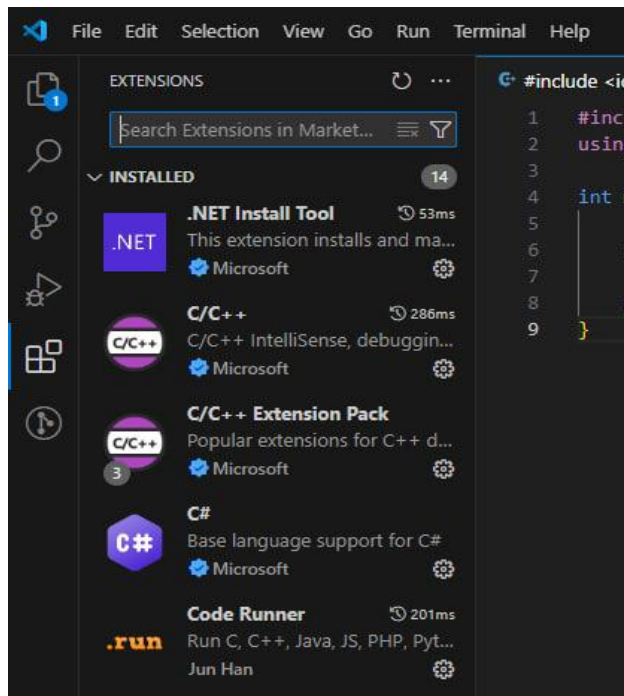
Завдання №3:

Планований час виконання: 1 година.

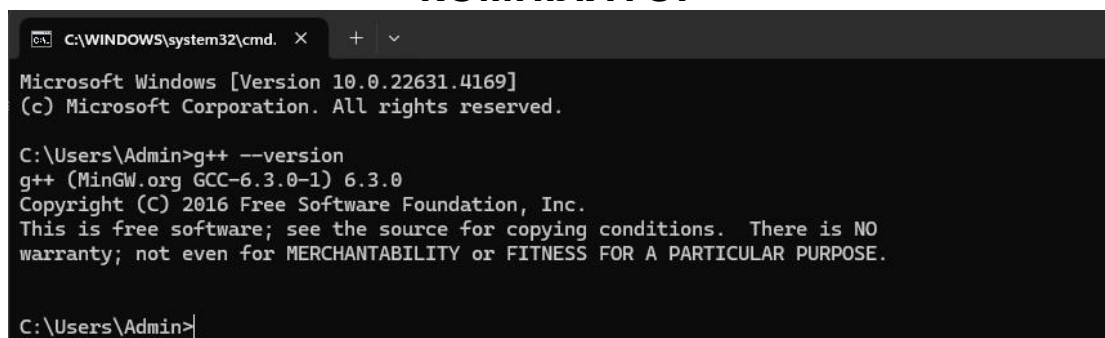


3) Конфігурація середовища до виконання завдань:

РОЗШИРЕННЯ



КОМПІЛЯТОР



Git та його базові команди




```
MINGW64/C:/Users/Admin

Admin@DESKTOP-AHHEQ55 MINGW64 ~ (master)
$ git --help
usage: git [-v | --version] [-h | --help] [-C <path>] [-c <name>=<value>]
[--exec-path=<path>] [--html-path] [--man-path] [--info-path]
[-p | --paginate | -P | --no-pager] [--no-replace-objects] [--no-lazy
-Fetch]
[--no-optional-locks] [--no-advice] [--bare] [--git-dir=<path>]
[--work-tree=<path>] [--namespace=<name>] [--config-env=<name>=<envva
r>]
<command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone Clone a repository into a new directory
  init Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add Add file contents to the index
  mv Move or rename a file, a directory, or a symlink
  restore Restore working tree files
  rm Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)
  bisect Use binary search to find the commit that introduced a bug
  diff Show changes between commits, commit and working tree, etc
  grep Print lines matching a pattern
  log Show commit logs
  show Show various types of objects
  status Show the working tree status

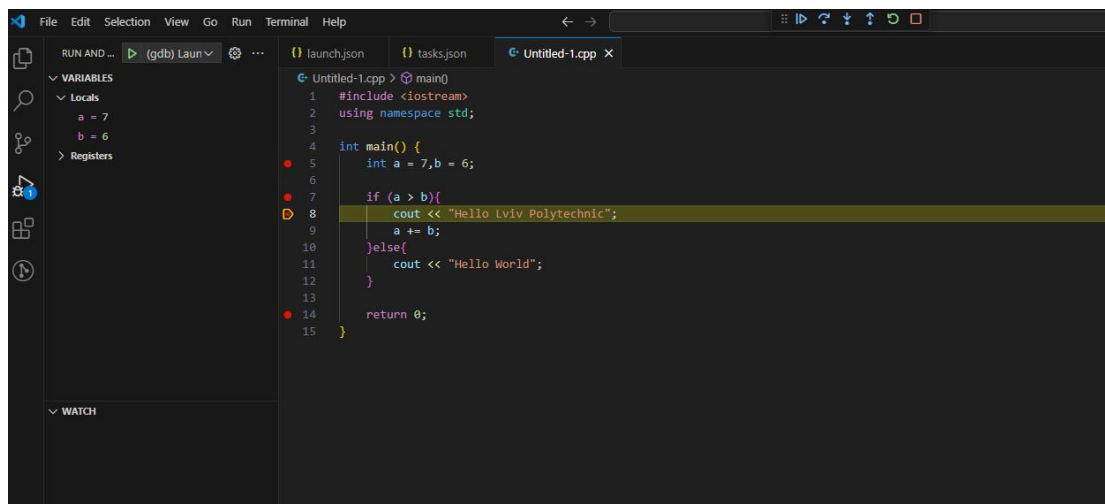
grow, mark and tweak your common history
  branch List, create, or delete branches
  commit Record changes to the repository
  merge Join two or more development histories together
  rebase Reapply commits on top of another base tip
  reset Reset current HEAD to the specified state
  switch Switch branches
  tag Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)
  fetch Download objects and refs from another repository
  pull Fetch from and integrate with another repository or a local branch
  push Update remote refs along with associated objects

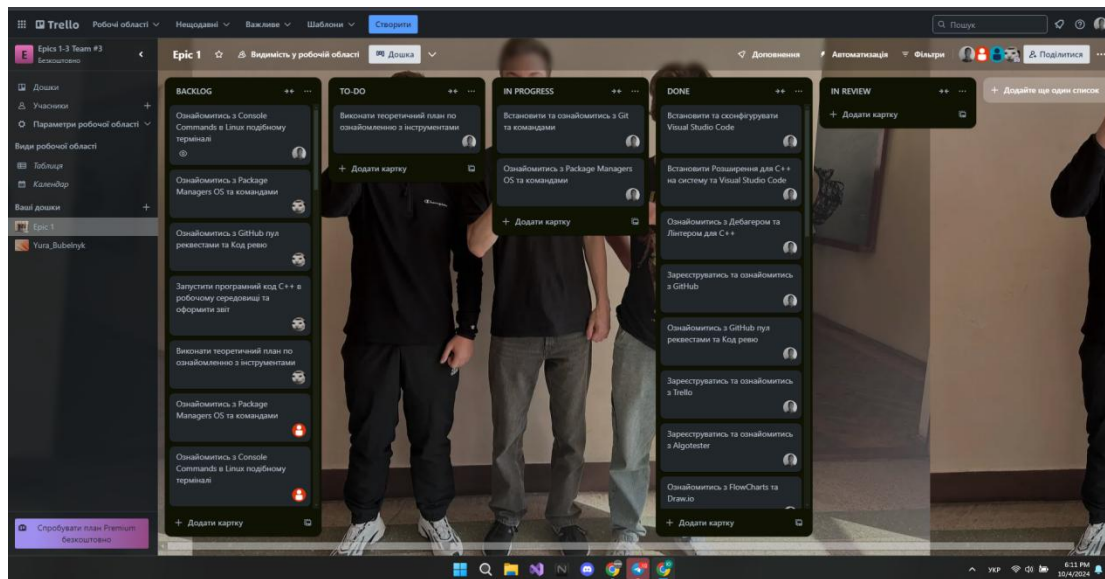
'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
See 'git help git' for an overview of the system.

Admin@DESKTOP-AHHEQ55 MINGW64 ~ (master)
$
```

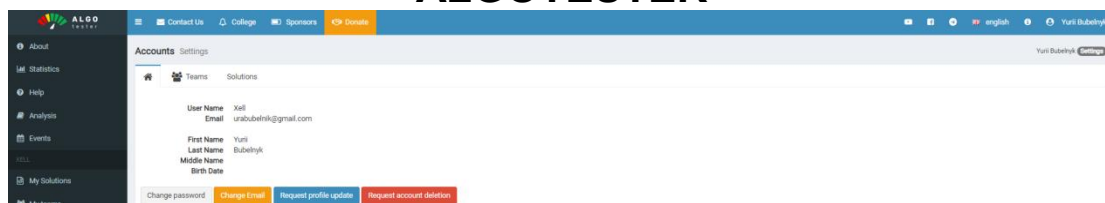
DEBUGGER



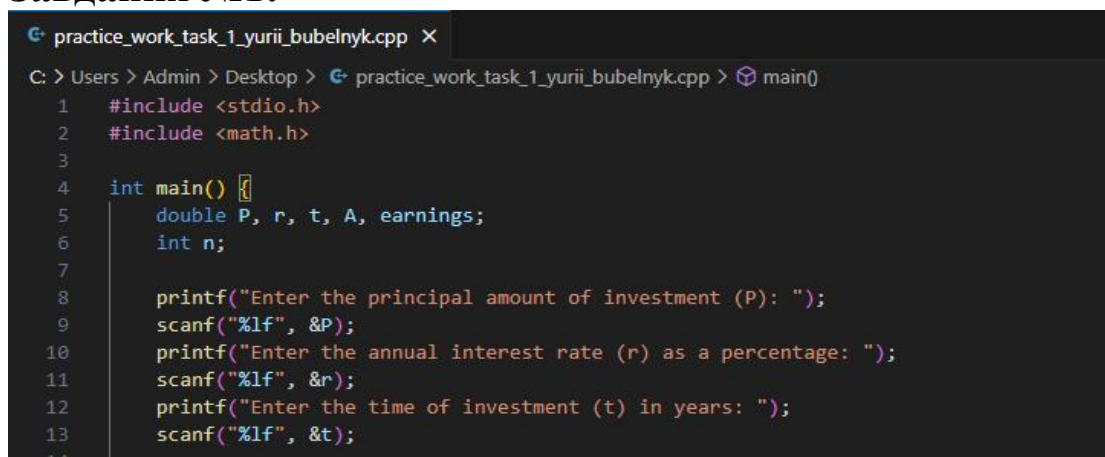
TRELLO



ALGOTESTER



4) Код програм з посиланням на зовнішні ресурси: Завдання №1:



```
printf("Choose an interest payout option:\n");
printf("1 - Monthly\n");
printf("2 - Quarterly\n");
printf("3 - Annually\n");
int option;
scanf("%d", &option);
```

```
switch (option) {
case 1:
    n = 12;
    break;
case 2:
    n = 4;
    break;
case 3:
    n = 1;
    break;
default:
    printf("Incorrect!\n");
    return 1;
}
```

```
r /= 100;
```

```
A = P * pow((1 + r / n), n * t);
```

```
earnings = A - P;
```

```
printf("Total investment amount: %.2lf\n", A);
printf("Earnings amount: %.2lf\n", earnings);
```

```
return 0;
```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/blob/epic_1_practice_and_labs_yurii_bubelnyk/ai_1_1/yurii_bubelnyk/epic_1/practice_work_task_1_yurii_bubelnyk.cpp

Завдання №2:

```

C: > Users > Admin > Desktop > Epic_1 > self_practice_work_algotester_task_1_yurii_bubelnyk > self_practice
1  #include <iostream>
2  using namespace std;
3
4  void numberOfBill(int value, int valueOfBills[],int amountOfBills){
5
6      int i = amountOfBills - 1;
7      int number = 0;
8
9      while (value > 0 && i >= 0){
10
11          if(value/valueOfBills[i] >= 1){
12
13              int count = value/valueOfBills[i];
14              number += count;
15              value -= count*valueOfBills[i];
16
17              }
18              i--;
19          }
20      cout << number;
21  }
22
23  int main(){
24
25      int bills[] = {1, 2, 5, 10, 20, 50, 100, 200, 500};
26      int size = sizeof(bills)/sizeof(bills[0]);
27      int value;
28
29      cin >> value;
30
31      numberOfBill(value, bills, size);
32
33      return 0;
34  }

```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/blob/epic_1_practice_and_labs_yurii_bubelnyk/ai_1_1/yurii_bubelnyk/epic_1/self_practice_work_algotester_task_1_yurii_bubelnyk.cpp

Завдання №3:

```

#include <iostream>
using namespace std;

void toBinary(int a){

    int arr[32]{};
    int i = 0;

    while (i < 32){
        arr[i] = a % 2;
        a /= 2;
        i++;
    }

    for (int j = i - 1; j >= 0; j--){
        cout << arr[j];
    }

}

void toMinusBinary (int a){

    int arr[32]{};
    int i = 0;

    while (i < 32){
        arr[i] = !(a % 2);
        a /= 2;
        i++;
    }

    arr[0] = 1;
    for (int j = i - 1; j >= 0; j--){
        cout << arr[j];
    }

}

int main() {
    int value;
    cout << "Enter the value: ";
    cin >> value;

    toBinary(value);
    cout << endl;
    toMinusBinary(value);

    return 0;
}

```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/blob/epic_1_practice_and_labs_yurii_bubelnyk/ai_1_1/yurii_bubelnyk/epic_1/self_practice_work_yura_bubelnyk.cpp

5) **Результат виконання завдань, тестування та фактично затрачений час:**

Завдання №1:

```
PS C:\Users\Admin> cd "c:\Users\Admin\Desktop\" ; if ($?) { g++ practice_work.cpp
Enter the principal amount of investment (P): 40000
Enter the annual interest rate (r) as a percentage: 7
Enter the time of investment (t) in years: 4
Choose an interest payout option:
1 - Monthly
2 - Quarterly
3 - Annually
1
Total investment amount: 52882.16
Earnings amount: 12882.16
PS C:\Users\Admin\Desktop>
```

Фактично затрачений час: 25-30 хвилин.

Завдання №2:

Created	Problem	Compiler	Result	Time (sec.)	Memory (MiB)	#
9 days ago	0021 - Депутатські гроші	C++ 23	Accepted	0.002	1.262	1781339

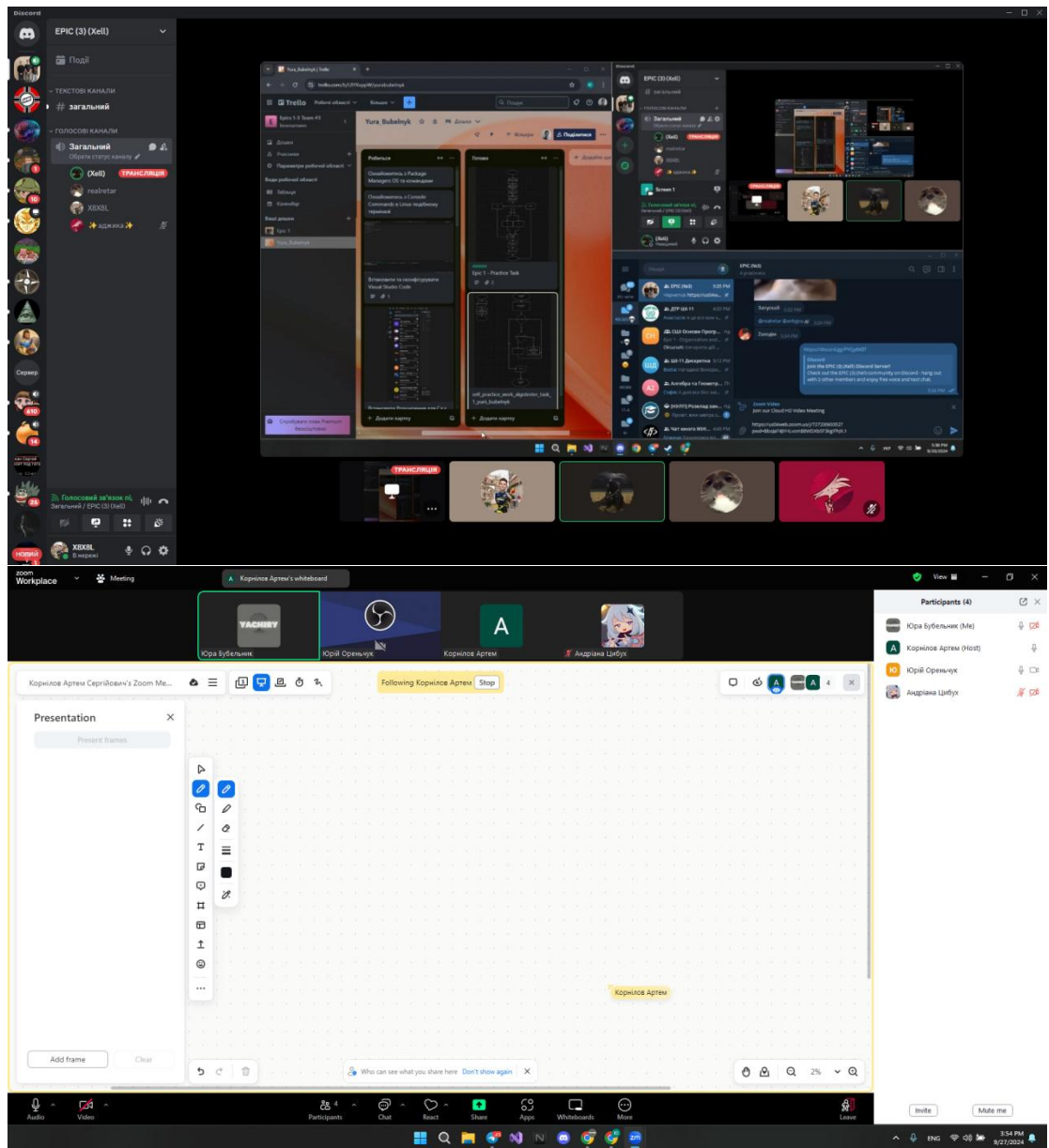
Фактично затрачений час: 30 хвилин.

Завдання №3:

[illegible]

Фактично затрачений час: 1.5 години.

Робота з командою:





Висновки:

Отже, в межах цього епіку я ознайомився з Git та GitHub, налаштував середовище для написання коду, а саме Visual Studio Code, скачав плагіни необхідні для компіляції коду на C++, ознайомився з базовим синтаксисом та написав перші програми. Використовував Draw.io для складання блок-схем, що допомогло краще зрозуміти написаний код. Дізнався про системи числення та операції над ними. Створено дошку завдань в Trello для кращої роботи в команді, також були проведені зустрічі на яких ми допомагали один одному.

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/29/files