

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 ІІІ-14

Кісіль Олександр Володимирович

Тема роботи:

Програмування: алгоритм, програма, код. Системи числення. Двійкова система числення. Розробка та середовище розробки програми: Visual Studio Code, FlowCharts та Draw.io, Git, Github, Algotester, Trello, Linux Console Commands.

Мета роботи:

Завантажити середовище розробки Visual Studio Code, встановити розширення для C++ на систему та Visual Studio Code, ознайомитись з дебагером та лінтером для C++, ознайомитись з основними поняттями мови c/c++ Встановити та ознайомитись з Git та командами, Зареєструватись та ознайомитись з GitHub, ознайомитись з GitHub пул реквестами та Код ревью, зареєструватись та ознайомитись з Trello, зареєструватись та ознайомитись з Algotester, ознайомитись з FlowCharts та Draw.io, ознайомитись з Word та створенням Звітів на Практичні та Лабораторні, ознайомитись з Системами числення та попрактикуватись з роботою в двійковій системі числення, запустити програмний код C++ в робочому середовищі та оформити звіт.

Теоретичні відомості:

- 1) Теоретичні відомості з переліком важливих тем:
 - Тема №1: Основні поняття мови c/c++.
 - Тема №2: Середовище розробки Visual Studio Code.
 - Тема №3: Системи числення.
 - Тема №4: FlowCharts та Draw.io.
 - Тема №5: Git та Github.
 - Тема №6: Algotester.
 - Тема №7: Trello.
 - Тема №8: Linux Console Commands.
- 2) Індивідуальний план опрацювання теорії:
 - Тема №1: Основні поняття мови C/C++
 - o Джерела Інформації
 - Лекції О. Пшеничного
 - o Що опрацьовано:
 - Ознайомлен з структурою коду програми в мові C++, з основними типами даних, з використанням змінних, бібліотекою iostream, з поняттям алгоритму.
 - Структура коду мови C++
 - Основні типи даних
 - Змінні
 - Бібліотека iostream, використання spacenames
 - Базові алгоритми
 - o Статус: Ознайомлен
 - o Початок опрацювання теми: 08.09.25
 - o Звершення опрацювання теми: 29.09.25
 - Тема №2: Середовище розробки Visual Studio Code
 - o Джерела Інформації:
 - [How to set up C++ in Visual Studio Code](#)
 - [How to use a linter](#)
 - [Debug a C++ project in VS Code](#)
 - o Що опрацьовано:
 - встановлено VS Code
 - Встановлено набір інструментів MinGW на систему та Visual Studio Code

- ознайомлен з лінтером і дебагером
 - Статус: Ознайомлен
 - Початок опрацювання теми: 11.09.25
 - Звершення опрацювання теми: 14.09.25
- Тема №3: Системи числення
 - Джерела Інформації:
 - Лекції О. Пшеничного
 - Практичні заняття
 - Що опрацьовано:
 - переведення чисел із десяткової системи в двійкову, вісімкову та шістнадцяткову, а також у зворотному напрямку
 - виконання додавання, віднімання, множення та ділення чисел, поданих у двійковій, вісімковій або шістнадцятковій формах
 - Статус: Ознайомлен
 - Початок опрацювання теми: 15.09.25
 - Звершення опрацювання теми: 22.09.25
- Тема №4 FlowCharts та Draw.io
 - Джерела Інформації:
 - Практичні заняття
 - [Flowcharts](#)
 - Що опрацьовано:
 - ознайомлен з поняттям FlowChart
 - навчився створювати FlowChart у draw.io
 - побудова блок-схем до практичних задач
 - Статус: Ознайомлен
 - Початок опрацювання теми: 16.09.25
 - Звершення опрацювання теми: 19.09.25
- Тема №5 Git та Github
 - Джерела Інформації:
 - [git-scm.com](#)
 - Практичні та лабораторні заняття
 - [Git vs. GitHub: What's the difference?](#)
 - [Git Tutorial for Beginners: Learn Git in 1 Hour](#)
 - [Git Basics - Getting a Git Repository](#)
 - Що опрацьовано:
 - Зареєструвався на Github
 - Завантажив Git
 - Вивчив базові git команди
 - навчився створювати репозиторії, бренчі, додавати, комітити та пушити зміни, робити пул реквест
 - Статус: Ознайомлен
 - Початок опрацювання теми: 11.09.25
 - Звершення опрацювання теми: 14.09.25
- Тема №6 Algotester
 - Джерела Інформації:
 - Лекції Пшеничного
 - Що опрацьовано:
 - зареєстровано на Algotester
 - виконано кілька завдань
 - Статус: Ознайомлена
 - Початок опрацювання теми: 15.09.25
 - Звершення опрацювання теми: 15.09.25

- Тема №7 Trello
 - o Джерела Інформації:
 - trello.com
 - o Що опрацьовано:
 - створено board на Trello
 - створено список моїх завдань
 - надав членам моєї команди можливість відслідковувати мій прогрес та вносити зміни
 - o Статус: Ознайомлен
 - o Початок опрацювання теми: 19.09.25
 - o Звершення опрацювання теми: 19.09.25
- Тема №8 Linux Console Commands
 - o Джерела Інформації:
 - [Ubuntu.com](https://ubuntu.com)
 - o Що опрацьовано:
 - запустила термінал на дистрибутиві Ubuntu
 - освоїв декілька базових команд
 - o Статус: Ознайомлен
 - o Початок опрацювання теми: 26.10.25
 - o Звершення опрацювання теми: 28.10.25

Виконання роботи:

1. Опрацювання завдання та вимог до програм та середовища:

Завдання №1 Requirements management and design activities with Draw.io and Google Docs

- Навчитись створювати блок схеми в [Draw.io](https://draw.io)

Завдання №2 Configuration: Trello

- Створити і налаштувати аккаунт в Trello
- Створити дошку для команди, додати свої завдання на дошку
- Відслідковувати свій прогрес та прогрес команди

Завдання №3 Configuration: Linux Console Commands

- Налаштувати MSYS консоль на Windows системі
- Навчитись користуватись базовими консольними командами

Завдання №4 Configuration: Visual Studio Code, VSC C/C++ Extensions, IntelliSense, Code Runner, debugger

- Налаштувати Visual Studio Code та розширення для мови C/C++
- Налаштувати конфігураційні файли tests.json та launch.json.
- Запустити свою першу програму.
- Навчитись запускати та користуватись дебаггером.

Завдання №5 Configuration: Git

- Налаштувати Git на комп'ютері. Вивчити базові git команди та навчитись ними користуватись.

Завдання №6 Configuration: GitHub

- Створити GitHub аккаунт
- Приєднати Git та GitHub
- Навчитись створювати репозиторії, пулреквести, пушати та пулати код.

Завдання №7 Configuration: Algotester

- Створити Algotester аккаунт.
- Приєднатись до контексту.

Завдання №8 Configuration: Create Own Git Repo and Exchange Files with Teammate

- Створити свій репозиторій.

Завдання №9 Binary Calculations

- *Деталі завдання:* Згенерувати в рандомайзері десяткове число у від 20 до 99
 - o Згенерувати в рандомайзері десяткове число x від 20 до 99
 - o Перевести y у двійкову систему числення
 - o Перевести x у двійкову систему числення
 - o Додати два двійкових числа x та y

Завдання №10 Run First Program - задача про депозит

- *Деталі завдання:* Обчислити складні відсотки для депозиту, який був відкритий в банку на певний період часу під фіксовані відсотки з різними варіантами виплати відсотків. Виплати можливі: кожного місяця, кожного кварталу, кожного року
- *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 №11 Run Second Program – задача про зарплату працівника

Деталі завдання: Реалізувати програму для обчислення зарплати працівника за певний період часу. Наприкінці програма повинна обчислити і вивести суму заробітної плати працівника

Завдання №12 Run Third Program – задача про приріст переглядів відео

Деталі завдання: Реалізувати програму для обчислення лінійного приросту переглядів відео за певний період часу. Наприкінці програма має порахувати очікувану кількість переглядів через t днів.

Завдання №13 Run a Program From Algotester– задача з алготестера

Деталі завдання: Реалізувати програму з Алготестера на власний вибір та пройти перевірку. Я вибрав $a + b$

2. Дизайн та планована оцінка часу виконання завдань:

Програма №10 Run First Program - задача про депозит

- Блок-схема

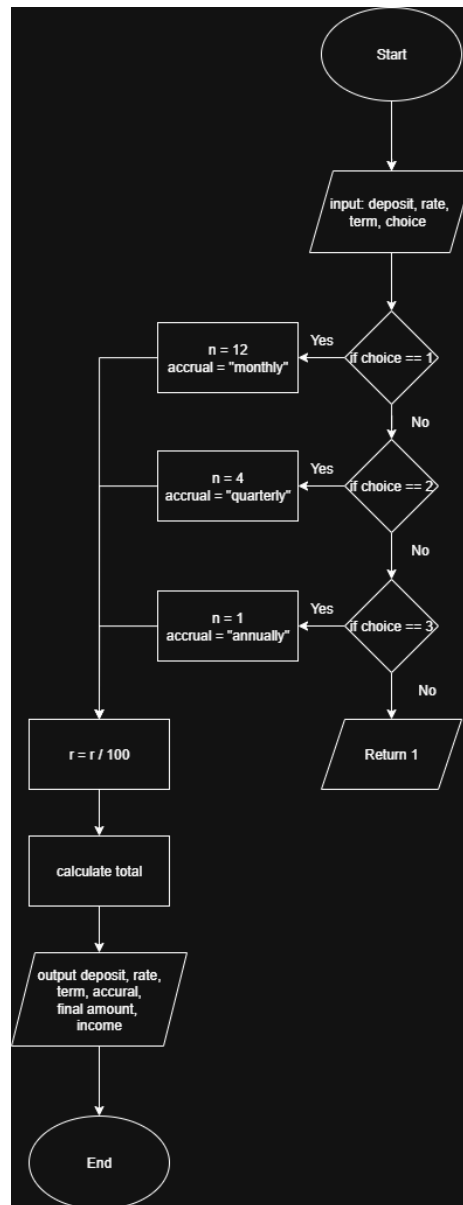


Рис 1. Блоксхема до задачі про депозит

- Планований час на реалізацію: 20хв
- Важливі деталі для врахування в імплементатії: Необхідно використати *scanf* та *printf*

Завдання №11 Run Second Program - задача про зарплату працівника

- Блок-схема

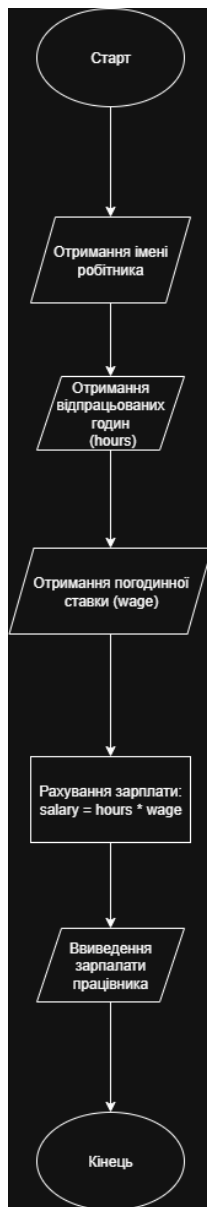


Рис 2. Блоксхема до задачі про зарплату працівника

- Планований час на реалізацію: 10 хв

Завдання №12 Run Third Program - задача про прогноз переглядів відео

- Блок-схема



Рис 3. Блоксхема до задачі про приріст переглядів відео

- Планований час на реалізацію: 5 хв

Завдання №13 Run a Program from Algotester - задача $a + b$

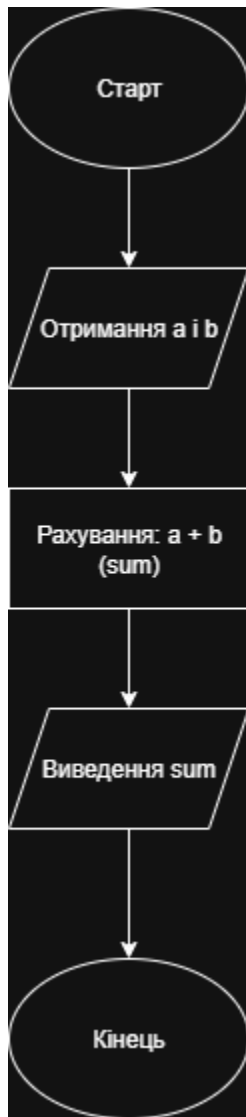


Рис 4. Блоксхема до задачі $a + b$ з алготестера

3. Конфігурація середовища до виконання завдань:

Завдання №1 Requirements management and design activities with Draw.io and Google Docs

- Встановив Draw.io на мій локальний комп'ютер
- Навчився створювати блоксхеми:

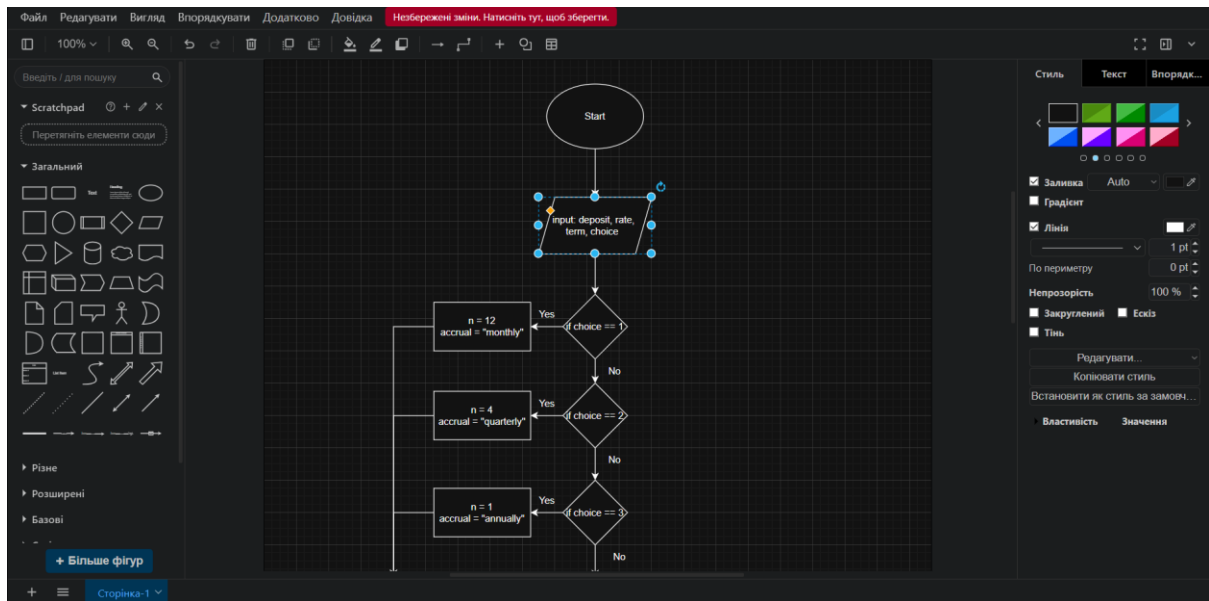


Рис 5. Створення блоксхеми в Draw.io

Завдання №2 Configuration: Trello

- Створено борду в трелло
- Надав тимейтам можливість переглядати та оцінювати мій прогрес.

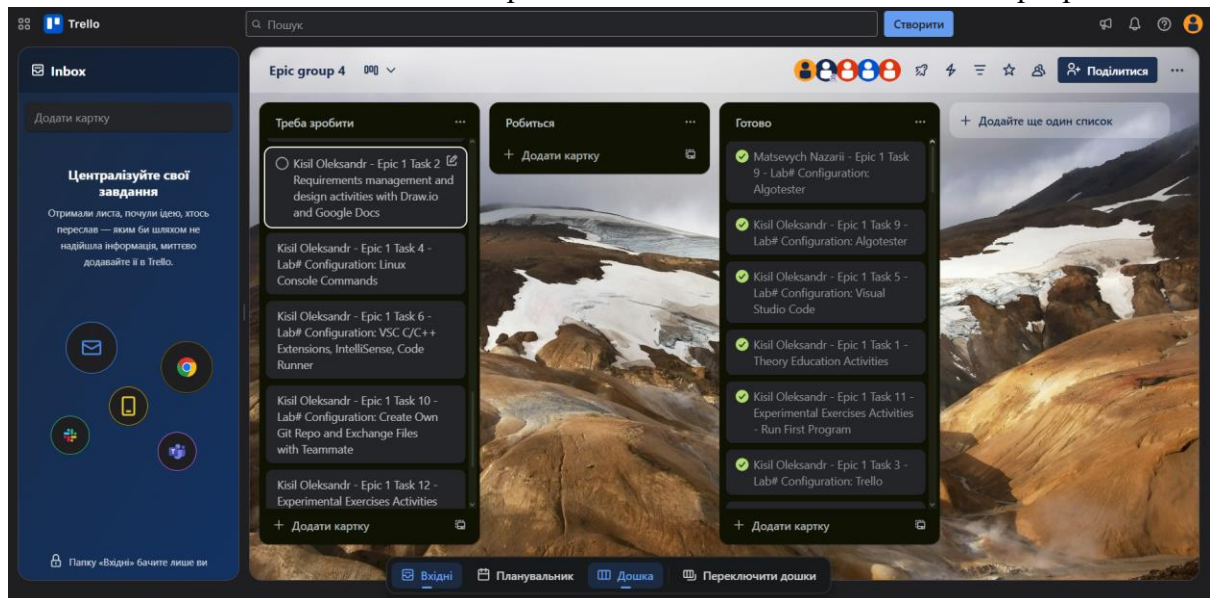


Рис 6. Створена дошка Trello

Завдання №3 Configuration: Linux Console Commands

- Попрактикував використання команд в терміналі.

```
MINGW64:/c

sasha@Sanek MINGW64 ~
$ pwd
/c/Users/sasha

sasha@Sanek MINGW64 ~
$ cd C://

sasha@Sanek MINGW64 /c
$ mkdir test_dir

sasha@Sanek MINGW64 /c
$ rm -r test_dir

sasha@Sanek MINGW64 /c
$ echo "Hello Word"
Hello Word

sasha@Sanek MINGW64 /c
$ |
```

Рис 7. Виконання лінукс команд

Завдання №4 Configuration: Visual Studio Code, VSC C/C++ Extensions, IntelliSense, Code Runner, debugger

- Встановив розширення для C/C++

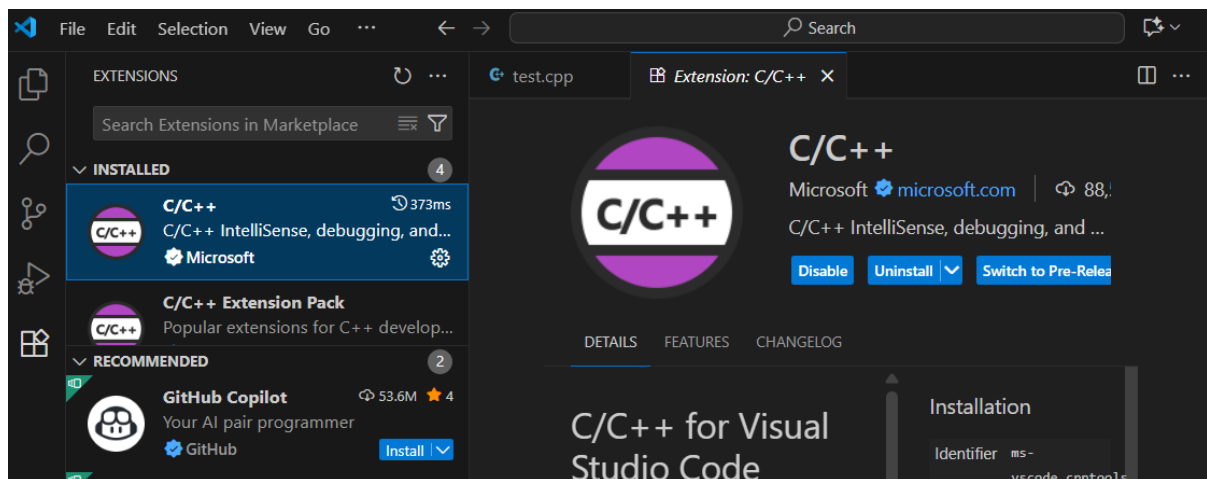


Рис 8. Встановлені розширення у Visual Studio Code

- Встановлено MSYS2 для компіляції та створення програм.

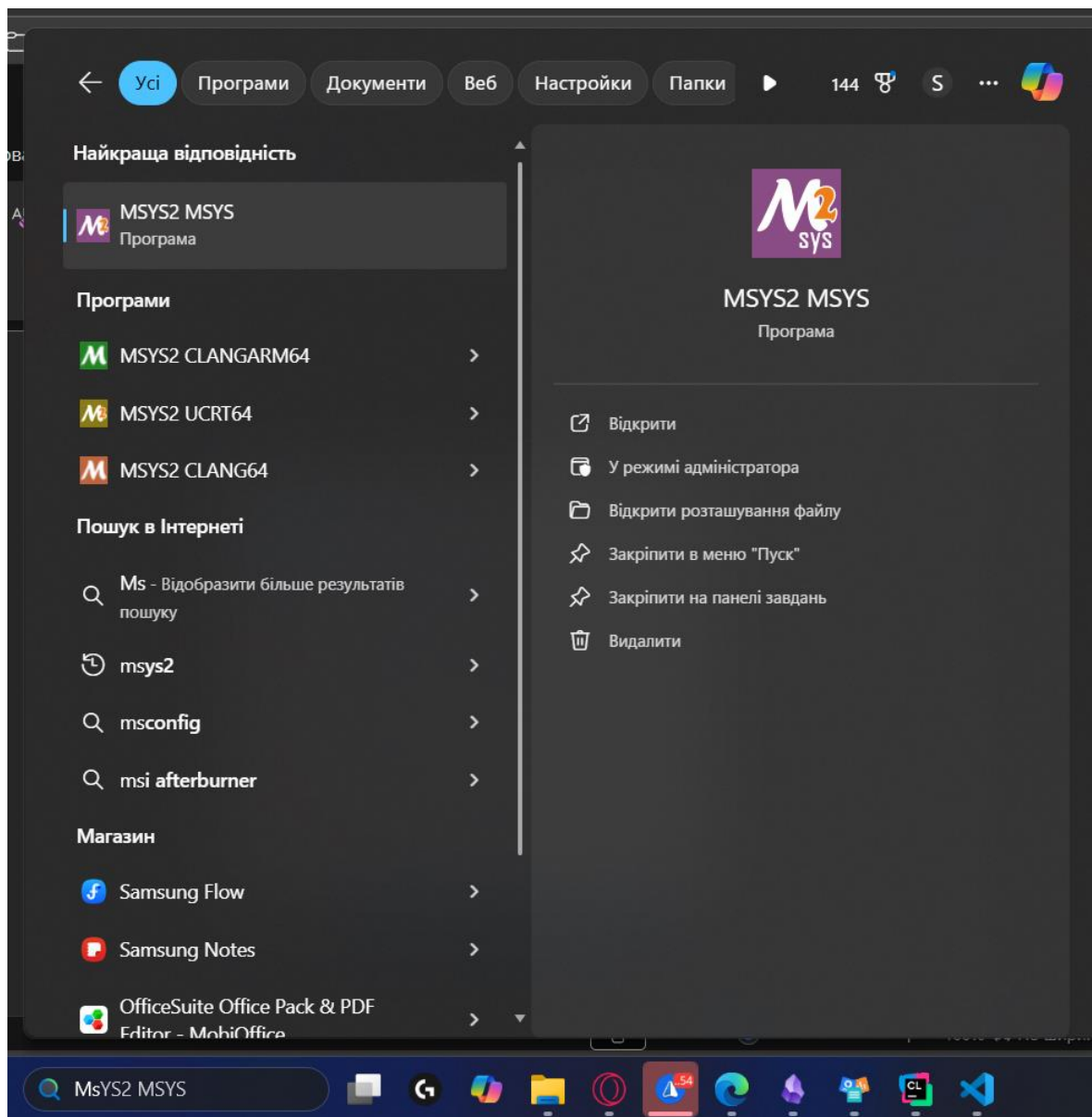


Рис 9. Встановлена програма MSYS2

Завдання №5 Configuration: Git

- приєднав Git до свого Github. На фото показана перевірка приєднаного гіта.

```
MINGW64:/c/Users/sasha
sasha@Sanek MINGW64 ~
$ git
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
  backfill   Download missing objects in a partial clone
  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.

sasha@Sanek MINGW64 ~
$
```

Рис 10. Перевірка встановленого гіта

Завдання №6 Configuration: GitHub

- Зареєструвався на GitHub:

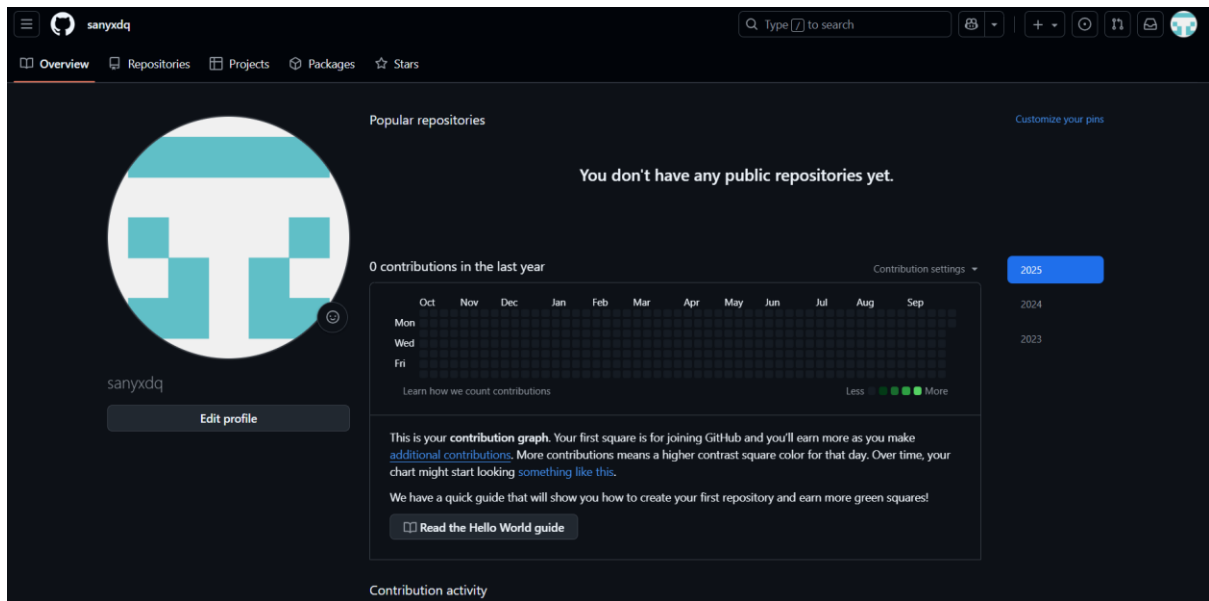


Рис 11. Створений Github аккаунт

- Створив ssh ключ

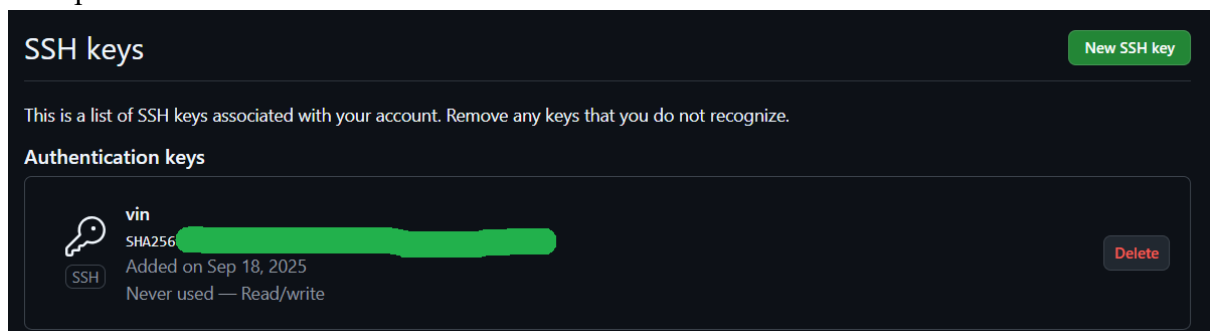


Рис 12. Створений ssh ключ

Завдання №7 Configuration: Algotester

- Зареєструвався на алготестері та виконав кілька завдань:

Створено	Задача	Компілятор	Результат	Час (сек.)	Пам'ять (МіБ)	№
день тому	0001 - А плюс В	C++	Зараховано	0.002	1.215	2160013
2 роки тому	0062 - Велике чаювання	Python 3.8	Ліміт часу 7	2.013	4.352	1573721
2 роки тому	0062 - Велике чаювання	Python 3.8	Неправильна відповідь 2	0.019	1.555	1573714
2 роки тому	0062 - Велике чаювання	Python 3.8	Неправильна відповідь 2	0.018	1.496	1573417
2 роки тому	0062 - Велике чаювання	Python 3.8	Неправильна відповідь 2	0.018	1.488	1573416
2 роки тому	0062 - Велике чаювання	Python 3.8	Неправильна відповідь 1	0.021	1.488	1573414
2 роки тому	0062 - Велике чаювання	Python 3.8	Неправильна відповідь 1	0.020	1.348	1573381
2 роки тому	0062 - Велике чаювання	Python 3.8	Неправильна відповідь 2	0.018	1.336	1573346
2 роки тому	0062 - Велике чаювання	Python 3.8	Неправильна відповідь 2	0.019	1.457	1573344
2 роки тому	0062 - Велике чаювання	Python 3.8	Неправильна відповідь 2	0.019	1.645	1573336
2 роки тому	0062 - Велике чаювання	Python 3.8	Неправильна відповідь 1	0.018	1.234	1573334
2 роки тому	0062 - Велике чаювання	Python 3.8	Помилка часу виконання 2	0.018	1.449	1573331
2 роки тому	0062 - Велике чаювання	Python 3.8	Помилка часу виконання 2	0.018	1.559	1573328
2 роки тому	0062 - Велике чаювання	Python 3.8	Неправильна відповідь 1	0.019	1.359	1573304

Рис 13. Створений Algotester аккаунт

Завдання №8 Configuration: Create Own Git Repo and Exchange Files with Teammate

- Створив свій власний, приватний репозиторій

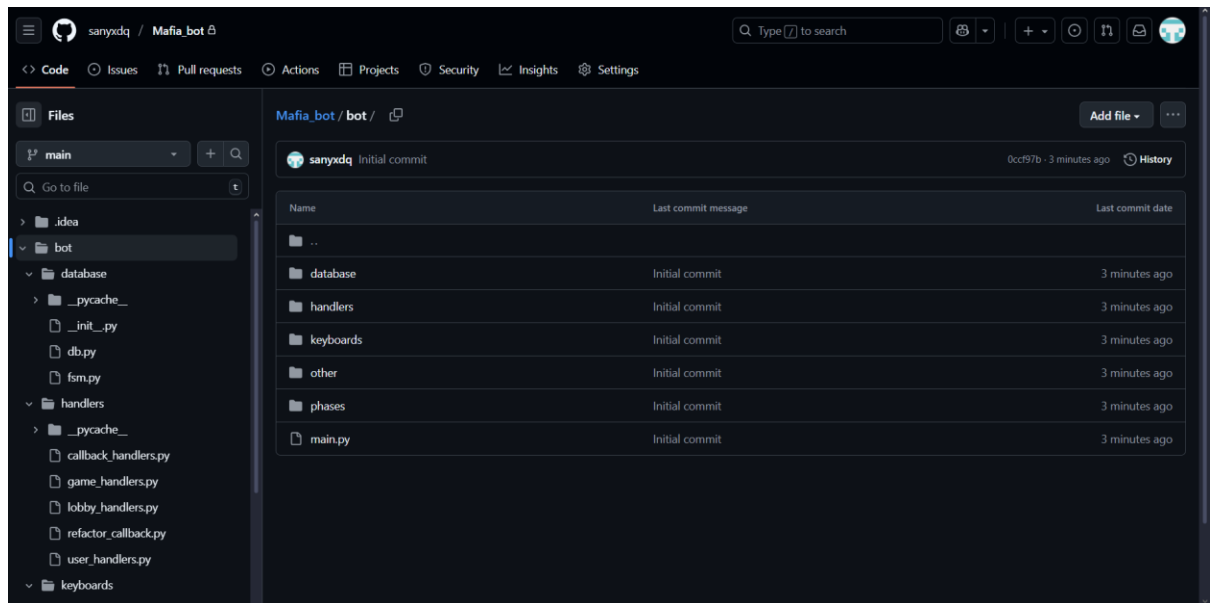


Рис 14. Створений приватний репозиторій

4. Код програм з посиланням на зовнішні ресурси:

Завдання №10 Run First Program - задача про депозит

```

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

int main() {
    double P, r, t, A;
    int n, choice;
    string accrual;

    printf("Enter deposit: ");
    scanf("%lf", &P);

    printf("Enter annual rate (%): ");
    scanf("%lf", &r);

    printf("Enter term (years): ");
    scanf("%lf", &t);

    printf("Choose accrual type:\n1 - Monthly\n2 - Quarterly\n3 - Annually\n");
    scanf("%d", &choice);

    if (choice == 1) {
        n = 12;
        accrual = "monthly";
    } else if (choice == 2) {
        n = 4;
        accrual = "quarterly";
    } else if (choice == 3) {
        n = 1;
        accrual = "annually";
    } else {
        printf("Invalid Choice\n");
        return 1;
    }

    r = r / 100.0;
    A = P * pow(1 + r / n, n * t);
    double income = A - P;

    printf("Your deposit amount: %.2lf\n", P);
    printf("Annual rate: %.2lf%%\n", r * 100);
    printf("Term: %.0lf years\n", t);
    printf("Accrual: %s\n", accrual.c_str());
    printf("Final amount: %.2lf\n", A);
    printf("Income: %.2lf\n", income);

    return 0;
}

```

Code #1. Код до задачі про депозит / Рис 15. Код до задачі про депозит

Посилання на файл коду в публеквесті: https://github.com/artificial-intelligence-department/ai_programming_playground_2025/pull/69/files#diff-84dbe86f41bcfa206fb124b520fece08ef806288d23d063c640c8cda09694983

Завдання №11 Run Second Program - задача про зарплату працівника

```
#include <iostream>
#include <windows.h>
#include <string>
using namespace std;

int main() {
    SetConsoleCP(65001);
    SetConsoleOutputCP(65001);

    string name;
    int hours;
    double wage, salary;

    cout << "Введіть ім'я працівника: ";
    cin >> name;
    cout << "Введіть відпрацьовані години: ";
    cin >> hours;
    cout << "Введіть погодинну ставку: ";
    cin >> wage;

    salary = hours * wage;

    cout << "Працівник " << name << " отримав зарплату " << salary << " грн";

    return 0;
}
```

Code #2. Код до задачі про зарплату працівника / Рис 16. Код до задачі про зарплату

Посилання на файл коду в пулреквесті: https://github.com/artificial-intelligence-department/ai_programming_playground_2025/pull/69/files#diff-4704f1d3a973ecffcf28e4c538bc63d5e94d2e0e093bccce810456e615d77465

Завдання №12 Run Third Program - задача про прогноз переглядів відео

```

#include <cmath>
#include <iostream>
#include <windows.h>
#include <string>
using namespace std;

int main() {
    SetConsoleCP(65001);
    SetConsoleOutputCP(65001);

    double V, p, t;

    printf("Перегляди, зростання (%%), дні: ");
    scanf("%lf %lf %lf", &V, &p, &t);

    double V_t = V * pow(1 + p / 100.0, t);

    printf("\nЗростання: %.2lf\n", V_t);
    printf("Прогноз переглядів через %.0lf днів: %.0lf", t, V_t);

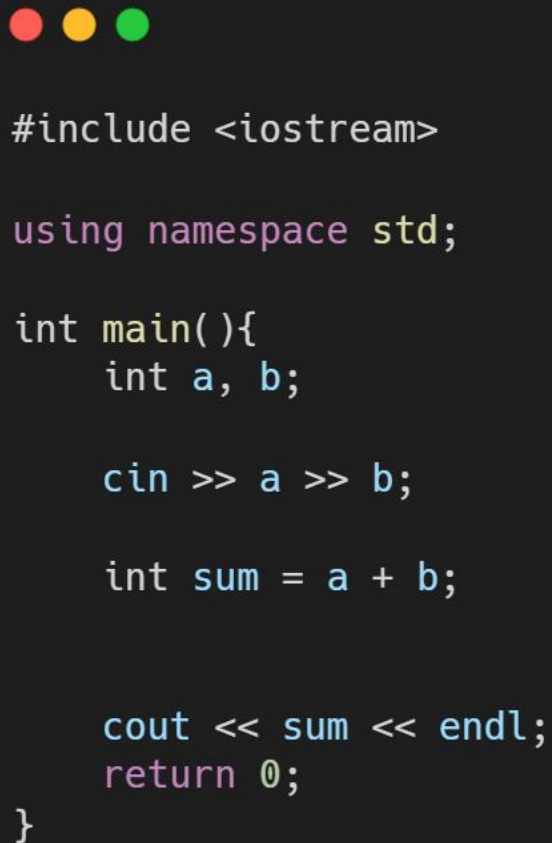
    return 0;
}

```

Code #3. Код до задачі про прогноз переглядів відео / Рис 17. Код до задачі про прогноз переглядів відео

Посилання на файл коду в пулреквесті: https://github.com/artificial-intelligence-department/ai_programming_playground_2025/pull/69/files#diff-a8ef74e3b6cb8f85b457ac4db8f147a4c5a46a21d48ee5b8dd2245bb67ae274c

Завдання №13 Run a Program from Algotester - задача a + b



```
#include <iostream>

using namespace std;

int main(){
    int a, b;

    cin >> a >> b;

    int sum = a + b;

    cout << sum << endl;
    return 0;
}
```

Code #4. Код до задачі a + b / Рис 18. Код до задачі a + b алготестера

Посилання на файл коду в пулреквесті:https://github.com/artificial-intelligence-department/ai_programming_playground_2025/pull/69/files#diff-af51d30b73aae395de4148167ae9e2268ffd1b3b71e29e842a8df431189d8c9d

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

Завдання №9 Binary Calculations

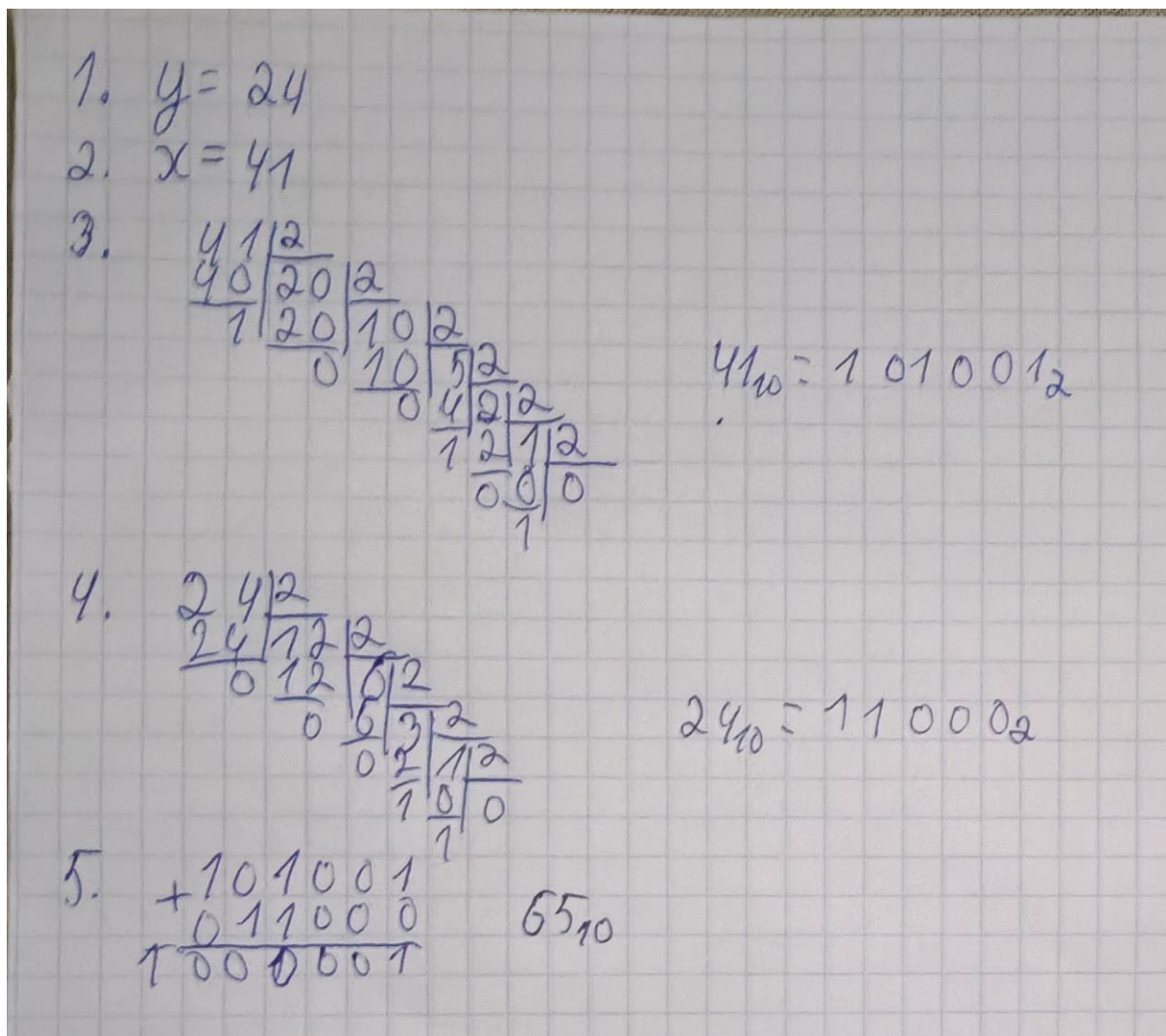


Рис 19. Обчислення в двійковій системі числення

Час затрачений на виконання завдання: 20хв

Завдання №10 Run First Program - задача про депозит

The screenshot shows the Visual Studio IDE with a C++ project named 'practise_work_task_1_oleksandr_kisil'. The code in the editor includes headers for `<iostream>`, `<string>`, and `<string.h>`, and uses the `std` namespace. The `main` function prompts the user for deposit amount, annual rate, and term, then calculates the final amount and income based on the chosen accrual type (Monthly, Quarterly, or Annually).

```
13 scanf("%lf", &P);
14
15 printf("Enter annual rate (%): ");
16 scanf("%lf", &r);
17
18 printf("Enter term (years): ");
19 scanf("%d", &t);
20
21 // Calculate final amount and income
22 double final_amount = P * (1 + r * t);
23 double income = final_amount - P;
24
25 printf("Your deposit amount: %lf\n", P);
26 printf("Annual rate: %lf%%\n", r * 100);
27 printf("Term: %d years\n", t);
28 printf("Accrual: %s\n", accrual_type);
29 printf("Final amount: %lf\n", final_amount);
30 printf("Income: %lf\n", income);
31
32 return 0;
33 }
```

The Run window shows the program's output, including the user's input and the calculated final amount and income.

```
Enter deposit:10000
Enter annual rate (%):200
Enter term (years):5
Choose accrual type:
1 - Monthly
2 - Quarterly
3 - Annually
3
Your deposit amount: 10000.00
Annual rate: 200.00%
Term: 5 years
Accrual: annually
Final amount: 2430000.00
Income: 2420000.00
Process finished with exit code 0
```

Рис 20. Виконання програми про депозит

Завдання №11 Run Second Program - задача про зарплату працівника

The screenshot shows the Visual Studio IDE with a C++ project named 'practise_work_task_2_oleksandr_kisil'. The code in the editor includes headers for `<iostream>`, `<string>`, and `<string.h>`, and uses the `std` namespace. The `main` function prompts the user for employee name, hours, and hourly wage, then calculates the total salary.

```
1 #include <iostream>
2 #include <string>
3 #include <string.h>
4 using namespace std;
5
6 int main()
7 {
8     SetConsoleCP( 65001);
9     SetConsoleOutputCP( 65001);
10
11     string name;
12     int hours;
13     double wage, salary;
14
15     cout << "Введіть ім'я працівника: ";
16     cin >> name;
17     cout << "Введіть відпрацьовані години: ";
18     cin >> hours;
19     cout << "Введіть погодинну ставку: ";
20     cin >> wage;
21
22     salary = hours * wage;
23
24     cout << "Зарплата працівника: " << salary << endl;
25
26     return 0;
27 }
```

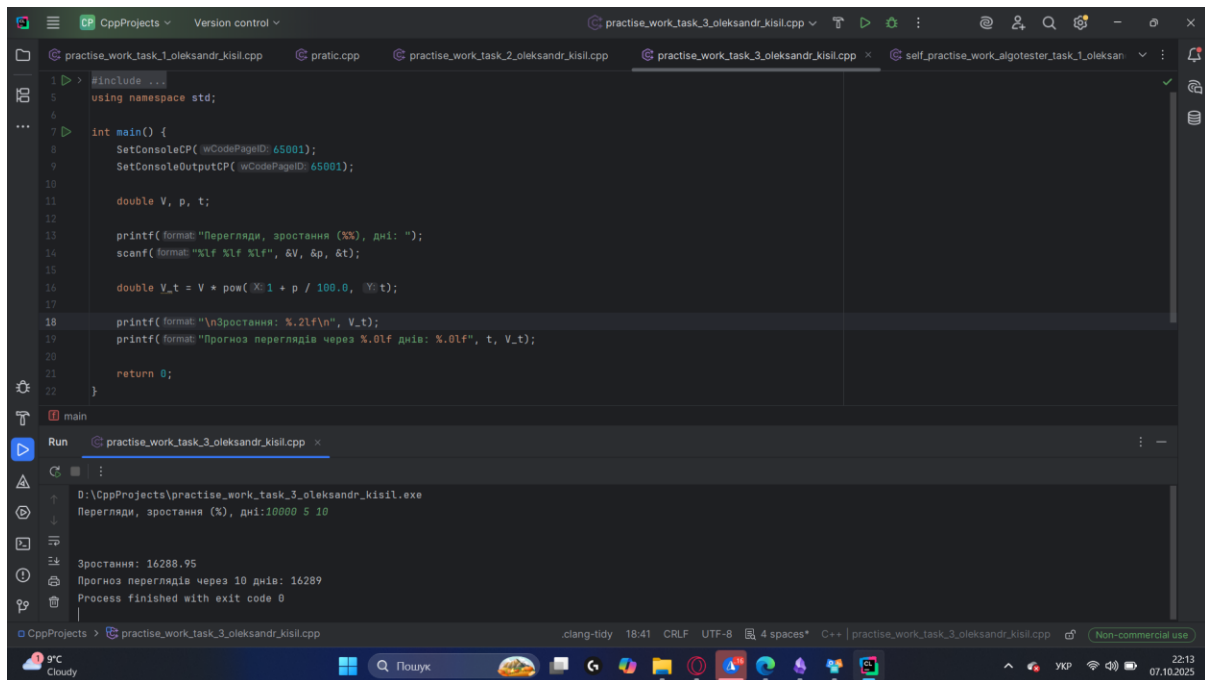
The Run window shows the program's output, including the user's input and the calculated salary.

```
Введіть ім'я працівника:Боб
Введіть відпрацьовані години:12
Введіть погодинну ставку:2000
Працівник Боб отримав зарплату 24000 грн
Process finished with exit code 0
```

Рис 21. Виконання програми про зарплату працівника

Час затрачений на виконання завдання: 10 хвилин

Завдання №12 Run Third Program - задача про прогноз переглядів відео



The screenshot shows the Visual Studio Code interface with a C++ project. The main file, `practise_work_task_3_oleksandr_kisil.cpp`, contains the following code:

```
1 > #include <iostream>
2
3 using namespace std;
4
5
6
7 int main() {
8     SetConsoleCP( wCodePageID: 65001);
9     SetConsoleOutputCP( wCodePageID: 65001);
10
11     double V, p, t;
12
13     printf(format: "Перегляди, зростання (%%), дні: ");
14     scanf(format: "%lf %lf %lf", &V, &p, &t);
15
16     double V_t = V * pow( (1 + p / 100.0, t));
17
18     printf(format: "\nЗростання: %.2lf\n", V_t);
19     printf(format: "Прогноз переглядів через %.0lf днів: %.0lf", t, V_t);
20
21     return 0;
22 }
```

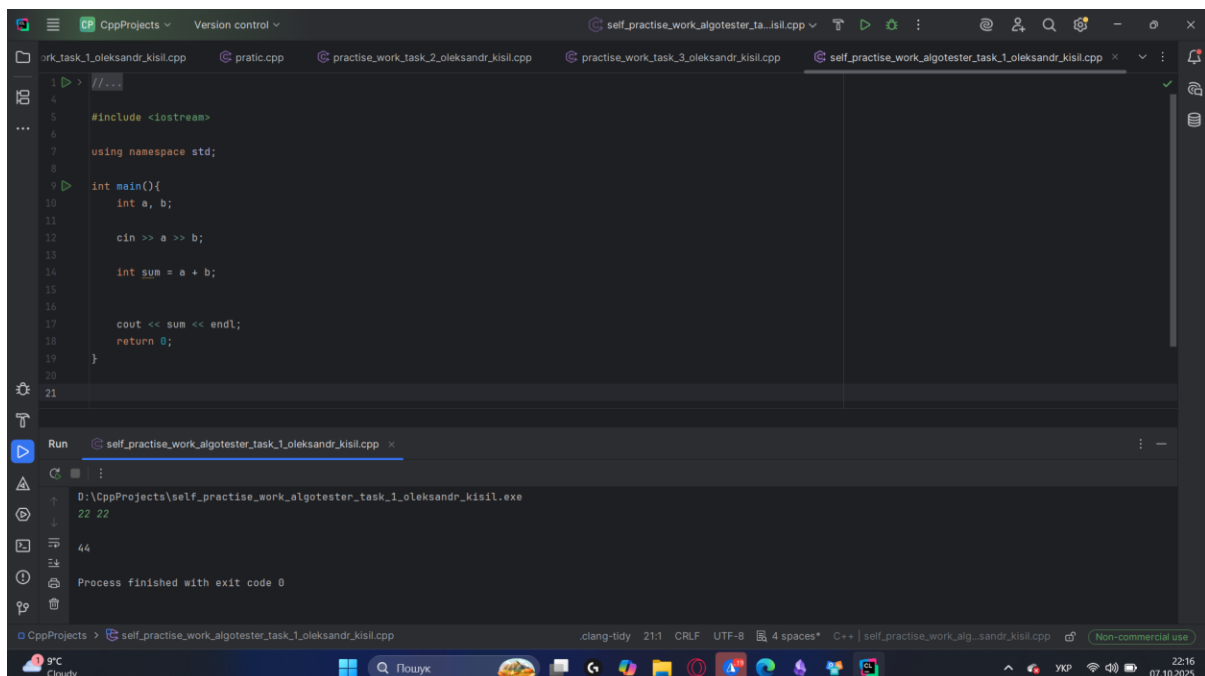
The Run window shows the execution of `practise_work_task_3_oleksandr_kisil.exe` with the following output:

```
Перегляди, зростання (%), дні:10000 5 10
Зростання: 16288.95
Прогноз переглядів через 10 днів: 16289
Process finished with exit code 0
```

Рис 22. Виконання програми про прогноз переглядів відео

Час затрачений на виконання завдання: 15 хвилин

Завдання №13 Run a Program from Algotester - задача a + b



The screenshot shows the Visual Studio Code interface with a C++ project. The main file, `self_practise_work_algotester_task_1_oleksandr_kisil.cpp`, contains the following code:

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

The Run window shows the execution of `self_practise_work_algotester_task_1_oleksandr_kisil.exe` with the following output:

```
22 22
46
Process finished with exit code 0
```

Рис 23. Виконання програми з алготестеру $a + b$

Час затрачений на виконання завдання: 5 хвилин

6. Кооперація з командою:

- Зустріч №1 з командою обговорюємо, які у нас є завдання та як їх виконувати



Рис 24. Скрін першої зустрічі в зум

Код рев'ю пуллреквесту:

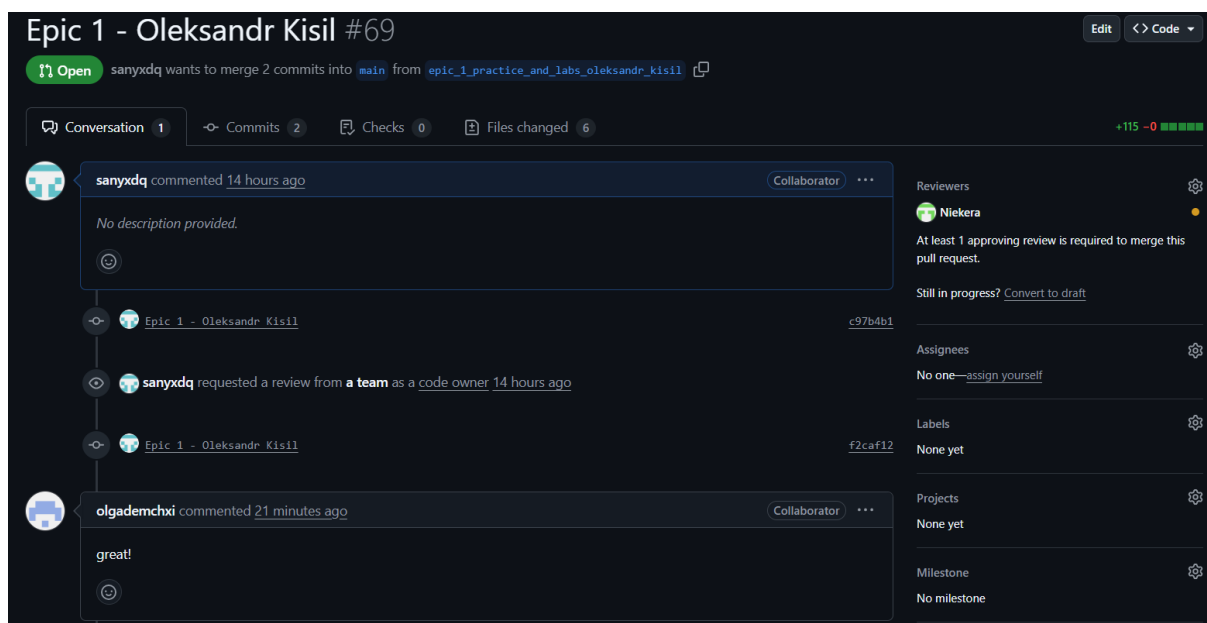


Рис 25. Скрін пуллреквесту

Висновки:

У процесі виконання завдань було здійснено всебічне ознайомлення з основними інструментами, необхідними для ефективної розробки програмного забезпечення мовою C++. Отримані знання та практичні навички створюють міцну основу для подальшого поглибленого вивчення програмування та реалізації більш складних проєктів.

Здобуті навички:

- **Робота в терміналі:** Засвоєно базові команди Linux-подібного терміналу, що дозволяє впевнено працювати з файловою системою та виконувати основні операції без графічного інтерфейсу
- **Середовище розробки:** Встановлено й налаштовано Visual Studio Code та необхідні плагіни для розробки на C++, що забезпечує зручне написання, налагодження та тестування коду
- **Система контролю версій:** Отримано практичні знання з використання Git та GitHub — від створення репозиторію до виконання перших комітів. Це дозволяє організовано зберігати код, відстежувати зміни та ефективно співпрацювати в команді.
- **Організація роботи:** Ознайомлено з Trello як інструментом для планування та управління завданнями, що сприяє кращій структуризації процесу навчання та розробки
- **Тестування алгоритмів:** Зареєструвався на Algotester для перевірки ефективності розроблених алгоритмів
- **Візуалізація:** Освоєно Draw.io для створення блок-схем та інших діаграм.
- **Системи числення:** Вивчено основні принципи роботи з двійковою системою числення.
- **Мова C++:** Отримано базові знання про мову C++ та функції введення/виведення даних. Написано та запущено програмний код на C++.
- **Робота у Word:** Створено звіт про виконану роботу.