

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):**

Студентка групи ШІ-11

Гуменюк Анастасія Олександрівна

Львів 2024

**Тема роботи:** Підготовка та налаштування всіх потрібних інструментів для подальшого навчання. Налаштування робочого середовища Visual Studio Code та ознайомлення з його інструментами. Система контролю версій GitHub, встановлення Git та вивчення його команд. Реєстрація на потрібних для роботи ресурсах (Algotester, Trello, Draw.io) та перша робота з ними. Відпрацювання навичок щодо обчислень над числами в двійковій системі числення. Написання першої програми та її тестування.

**Мета роботи:** Використати на практиці знання, отримані з опрацьованої по темі теорії. Навчитися встановлювати та налаштовувати основні інструменти, необхідні для розробки програм на C/C++ (Runner, Debugger). Отримати досвід в роботі з FlowCharts та організації своєї і командної роботи за допомогою Trello. Опанувати тему систем числення, попрактикувавшись на діях у двійковій системі. Ознайомитись з Linux подібним терміналом для подальшої роботи в ньому. Встановити Git, вивчити основні команди для роботи з ним та створити гілку на GitHub.

### **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:
  - Тема №1: Дизайн з FlowCharts для Simple Algorithms. Word та Draw.io як середовище відображення дизайну.
  - Тема № 2: Trello, як інструмент організації та відстеження роботи команди.
  - Тема № 3: Встановлення та Конфігурація Visual Studio Code. Встановлення Розширень Visual Studio Code для C++. Debugger.
  - Тема № 4: Операційна система Linux. Переваги та основна інформація.
  - Тема № 5: Встановлення Git та його команди.
  - Тема № 6: Двійкова система числення, дії над числами, записаними у вигляді двійкового коду.

### **2. Індивідуальний план опрацювання теорії:**

Тема № 1: Дизайн з FlowCharts для Simple Algorithms. Word та Draw.io як середовище відображення дизайну.

- Джерела Інформації:

<https://www.programiz.com/article/flowchart-programming>

<https://www.visual-paradigm.com/tutorials/flowchart-tutorial/>

- Що опрацьовано:
  - Основні елементи блок-схем, позначення (термінатор, дані, процес, рішення), приклади Flowcharts на конкретних задачах.
  - Зрозуміло мету використання блок-схеми при написанні коду.
- Статус: Ознайомлена
- Початок опрацювання теми: 26.09.2024.
- Звершення опрацювання теми: 26.09.2024 (30 хв.).

Тема № 2: Trello, як інструмент організації та відстеження роботи команди.

- Джерела Інформації:
  - <https://trello.com/guide/trello-101>
- Що опрацьовано:
  - Дошка, список, картка – як з цим працювати, як організувати свою першу дошку та додати до неї учасників команди.
- Статус: Ознайомлена
- Початок опрацювання теми: 26.09.2024.
- Звершення опрацювання теми: 26.09.2024 (15хв.).

Тема № 3: Встановлення та Конфігурація Visual Studio Code.  
Встановлення Розширень Visual Studio Code для C++. Debugger.

- Джерела Інформації:
  - <https://code.visualstudio.com/docs/cpp/config-mingw>
  - [https://www.youtube.com/watch?v=2VokW\\_Jt0oM](https://www.youtube.com/watch?v=2VokW_Jt0oM)
  - <https://www.youtube.com/watch?v=DMWD7wfhgNY&t=4s>
  - [https://www.youtube.com/watch?v=77v-Poud\\_io](https://www.youtube.com/watch?v=77v-Poud_io)
- Що опрацьовано:
  - За інструкцією встановлено Visual Studio Code.
  - У Visual Studio Code за допомогою розділу Extension встановлено Code Runner, C/C++ Extension Pack, C/C++.
  - У налаштуваннях поставлено прапорець “Run in Terminal”, щоб результат виконання виводився у Terminal, а не в Output.
  - C++ debugger: breakpoints, Watch.
- Статус: Ознайомлена
- Початок опрацювання теми: 20.09.2024.
- Звершення опрацювання теми: 22.09.2024 (3 год.).

Тема № 4: Операційна система Linux. Переваги та основна інформація.  
Джерела Інформації:

<https://www.freecodecamp.org/news/the-linux-commands-handbook/>  
<https://acode.com.ua/what-is-linux/>  
<https://acode.com.ua/why-linux-is-more-secure/>  
<https://acode.com.ua/linux-vs-windows/>

- Що опрацьовано:
  - Що таке операційна система.
  - Linux та Windows порівняння. Що обирають звичайні користувачі, а що програмісти.
  - Цікавий вислів під назвою «Закон Лінуса»: “При достатній кількості спостерігачів всі помилки лежатимуть на поверхні”, як аргументація того, чому програмістам краще працювати в команді.
  - Команди Linux: echo – вивід тексту, ls – список файлів і папок у поточному каталозі, cd – перехід в папку, mkdir – створення папки, mv – переміщення файлу, cp – копіювання файлу, touch – створення пустого файлу.
- Статус: Ознайомлена
- Початок опрацювання теми: 7.09.2024.
- Звершення опрацювання теми: 7.09.2024 (1 год.).

Тема № 5: Встановлення Git та його команди.

- Джерела Інформації:
  - <https://javarush.com/ua/groups/posts/uk.2683.pochatok-roboti-z-git-dokladniy-gayd-dlja-novachkv>
  - [https://www.youtube.com/watch?v=vR-y\\_2zWrIE&list=PLWKjhJtqVAbkFqHnNaxpOPhh9tSWMXIF](https://www.youtube.com/watch?v=vR-y_2zWrIE&list=PLWKjhJtqVAbkFqHnNaxpOPhh9tSWMXIF)
  - <https://www.youtube.com/watch?v=a0CftUjkrbs>
- Що опрацьовано:
  - Що таке Git.
  - Встановлено git та локально налаштовано.
  - Згенеровано ключ та під'єднано до GitHub.
  - Опрацьована інформація щодо основних елементів Git (гілка, коміт і т.д.)
- Статус: Ознайомлена
- Початок опрацювання теми: 28.09.2024.
- Звершення опрацювання теми: 01.10.2024 (2 год.)

Тема № 6: Двійкова система числення, дії над числами, записаними у вигляді двійкового коду.

- Джерела Інформації:  
<https://www.calculator.net/binary-calculator.html>  
<https://acode.com.ua/urok-34-tsilochoyselni-typy-danyh-short-int-i-long/>  
<https://acode.com.ua/urok-36-typy-danyh-z-plavayuchoyu-krapkoyu-float-double-i-long-double/>  
<https://www.youtube.com/watch?v=xTmvZvicrXs>
- Що опрацьовано:
  - Переведення з десяткової системи числення в двійкову та навпаки.
  - Дії над числами, записаними двійковим способом.
  - При діленні цілочисельного типу даних, дробова частина в програмуванні просто відкидається (не округлюється).
  - При діленні чисел, записаних у двійковій системі, отримуємо нескінченну послідовність і через подібні розбіжності в представленні чисел в різних системах числення можуть виникати проблеми з точністю.
- Статус: Ознайомлена
- Початок опрацювання теми: 21.09.2024.
- Звершення опрацювання теми: 21.09.2024 (40хв.).

### Виконання роботи:

1. Опрацювання завдання та вимог до програм та середовища:

#### Завдання №11 Епіс 1

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

Формула:

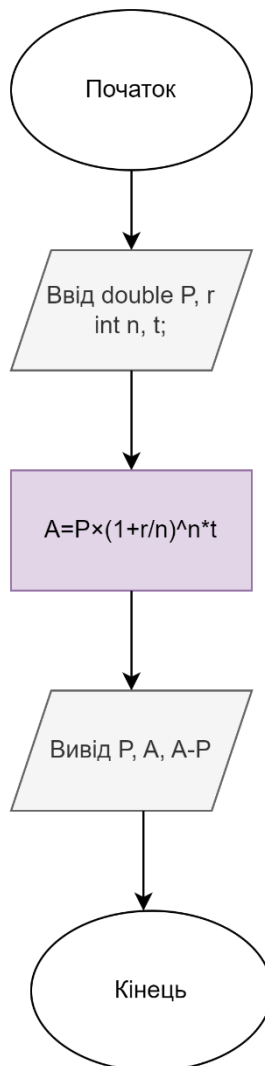
$$A = P \times \left(1 + \frac{r}{n}\right)^{n \times t}$$

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.

Деталі: змінні P та r повинні мати тип double, щоб туди можна було записати дробове число.

### 2.1 Блок схема до задачі:



Запланована витрата часу: 20 хв.

### Завдання «Марічка і печиво» з Algotester Epic 1

Зібралися Зеник і Марічка разом з пластунами в похід. Похід — серйозна справа. Потрібно заpastись 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.

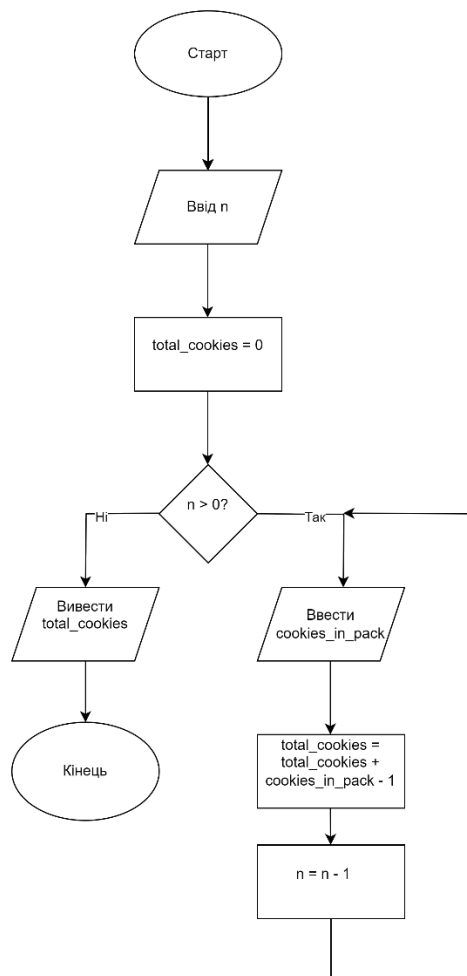
Марічка дуже любить печиво. Сьогодні, коли всі пластуни покинуть свої намети і підуть купатися в річку, Марічка планує непомітно з'їсти трохи

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.

Деталі: щоб пачка рахувалась, потрібно, аби там було хоча б одне печиво.

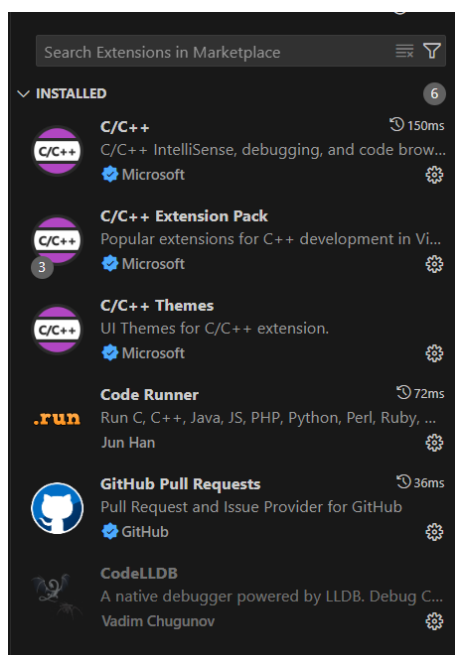
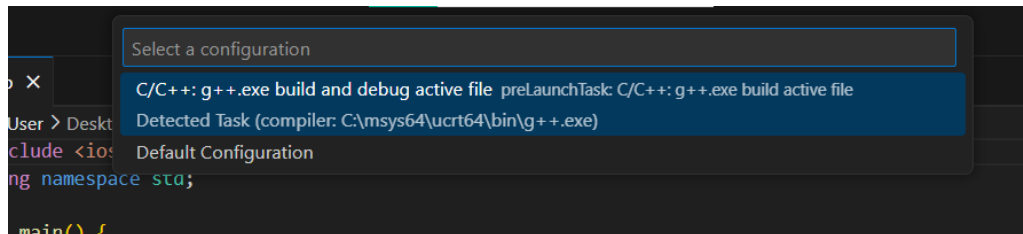
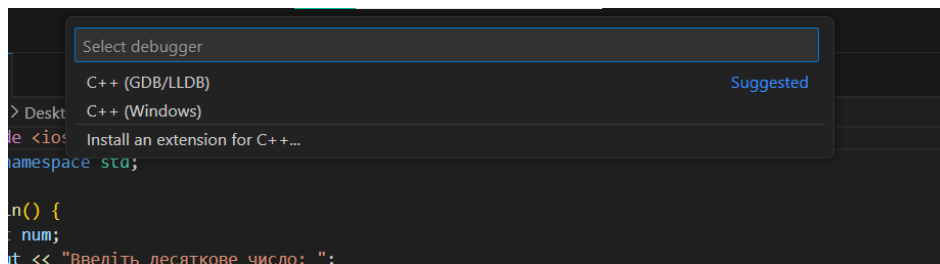
## 2.2 Блок схема до задачі:



Запланована витрата часу: 30 хв.

## 2. Конфігурація середовища до виконання завдань:

### 1. Налаштований Visual Studio Code:



## 2. Зареєстрований акаунт на Algotester:

ALGO tester

Про нас  
Статистика  
Допомога  
Розбір  
Події  
ANASTASIA.HUMENIUK  
Мої розв'язки  
Мої команди  
Архів  
Задачі  
Рейтинг  
Черга розв'язків

Зв'яжіться з нами  
Коледж  
Спонсори  
Підтримати

українська  
Анастасія Гуменюк

Архів

☐ Приховати мітки

Назва	Задача	Джерело	Р
0001	A плюс B	basic	Прості задачі
0002	Найбільша зростаюча підпоследовність	dp, basic	Прості задачі
0003	Офісна Вулиця. Частина 1	greedy, easy	Прості задачі
0004	Офісна Вулиця. Частина 2	greedy, easy	Прості задачі
0005	Центральна дільниця	graphs, basic	Прості задачі
0006	Фарбування	number theory, easy	Прості задачі
0007	Найпростіші запити	data structures, fenwick, segment tree, basic	Прості задачі
0008	Трохи складніші запити	data structures, easy	Прості задачі
0009	Дороги та міста	graphs, data structures, dsu, medium	Прості задачі
0010	Юний художник	data structures, easy	Прості задачі
0011	Maniara і паша	graphs, dsu, medium	Облаштована лінійна 2016

Анастасія Гуменюк  
Зареєстровано 05.09.2024 15:06:49

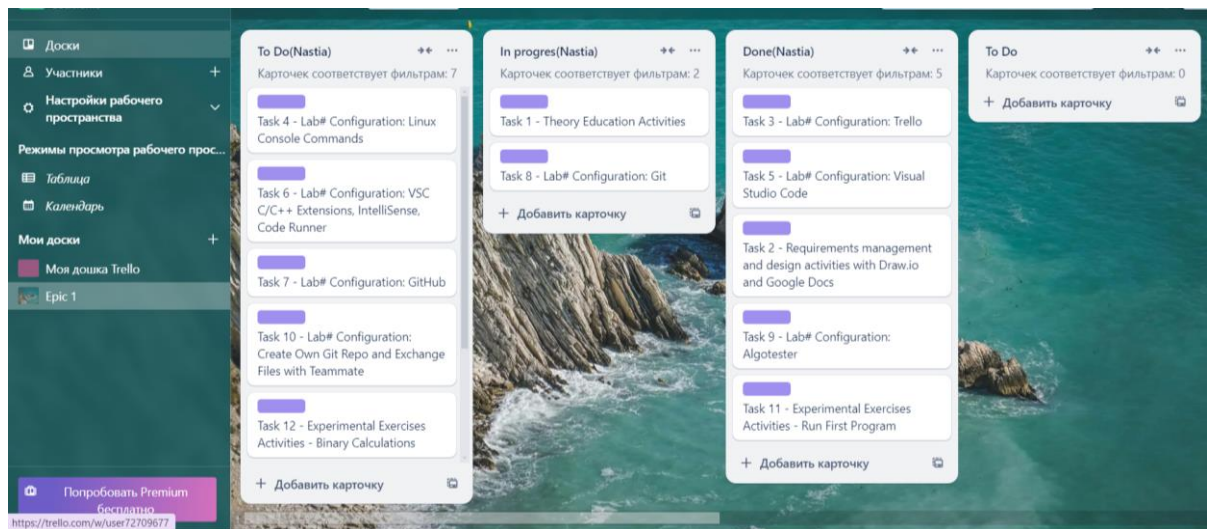
Мої команди  
Мої розв'язки

Обліковий запис  
Вийти

134  
113  
82  
153  
68  
90  
53  
2170



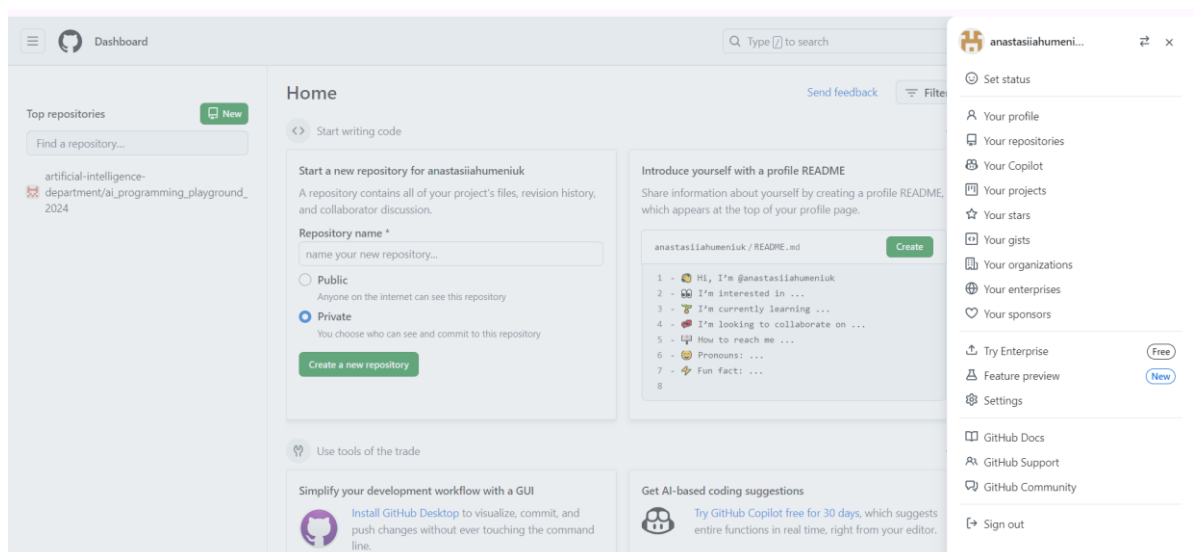
### 3. Дошка на Trello:



### 4. Встановлений Git:



### 5. Зареєстрований акаунт на GitHub:



## 6. Доданий файл в папку гілки на GitHub:

```
User@DESKTOP-7RM97RC MINGW64 ~/ai_programming_playground_2024/ai_11/anastasiia_humeniuk/epic1
$ git add calculations_practice_work_task_1_john_black.docx

User@DESKTOP-7RM97RC MINGW64 ~/ai_programming_playground_2024/ai_11/anastasiia_humeniuk/epic1
$ git add .

User@DESKTOP-7RM97RC MINGW64 ~/ai_programming_playground_2024/ai_11/anastasiia_humeniuk/epic1
$ git status
On branch epic_1_practice_and_labs_anastasiia_humeniuk
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   calculations_practice_work_task_1_john_black.docx

User@DESKTOP-7RM97RC MINGW64 ~/ai_programming_playground_2024/ai_11/anastasiia_humeniuk/epic1
$ git commit -m "Epic 1 - Anastasiia Humeniuk"
[epic_1_practice_and_labs_anastasiia_humeniuk 21071c2] Epic 1 - Anastasiia Humeniuk
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 ai_11/anastasiia_humeniuk/epic1/calculations_practice_work_task_1_john_black.docx

User@DESKTOP-7RM97RC MINGW64 ~/ai_programming_playground_2024/ai_11/anastasiia_humeniuk/epic1
$ git push origin epic_1_practice_and_labs_anastasiia_humeniuk
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 8 threads
Writing objects: 100% (6/6), 1.75 MiB | 1.56 MiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:artificial-intelligence-department/ai_programming_playground_2024.git
 5859124..21071c2  epic_1_practice_and_labs_anastasiia_humeniuk -> epic_1_practice_and_labs_anastasiia_humeniuk

User@DESKTOP-7RM97RC MINGW64 ~/ai_programming_playground_2024/ai_11/anastasiia_humeniuk/epic1
$ |
```

## 4. Код програм з посиланням на зовнішні ресурси:

### 1) Код з практичної по темі:

```
5. #include <iostream>
6. #include <cmath>
7. #include <cstdio>
8. #include <locale.h>
9.
10.int main() {
11.    setlocale(LC_ALL, "ukr");
12.
13.    double P, r, A;
14.    int n, t;
15.
16.    printf("Введіть початкову суму інвестицій (P): ");
17.    scanf("%lf", &P);
18.
19.    printf("Введіть річну процентну ставку : ");
20.    scanf("%lf", &r);
21.
22.    printf("Введіть кількість нарахувань відсотків на рік (n): ");
23.    scanf("%d", &n);
24.
25.    printf("Введіть тривалість інвестицій у роках (t): ");
26.    scanf("%d", &t);
27.
28.    r = r / 100;
29.    A = P * pow(1 + r / n, n * t);
```

```

30.
31.     printf("\nПочаткова сума: %.2f\n", P);
32.     printf("Загальна сума інвестицій після %d років: %.2f\n", t, A);
33.     printf("Зароблена сума: %.2f\n", A - P);
34.
35.     return 0;
36.}

```

Деталі:

- щоб перетворити процентну ставку з відсотків у десятковий дріб, поділимо  $r$  на 100.
- щоб текст виводився українською мовою, було додано локаль `setlocale(LC_ALL, "ukr")` та під'єднано бібліотеку `<locale.h>`.

2) Власний код по темі з задач алготестеру:

```

3. #include <iostream>
4. using namespace std;
5.
6. int main() {
7.     int n;
8.     cin >> n;
9.
10.    long long total_cookies = 0;
11.    int cookies_in_pack;
12.
13.    while (n--) {
14.        cin >> cookies_in_pack;
15.        total_cookies += cookies_in_pack - 1;
16.    }
17.
18.    cout << total_cookies << endl;
19.    return 0;
20.}

```

Деталі:

- $n$  – кількість упаковок з печивом;
- `cookies_in_pack` – кількість печива в одній упаковці;
- `total_cookies` – кількість печива, яку може з'їсти Марічка;
- `total_cookies += cookies_in_pack - 1` – віднімаємо 1, тому що, якщо залишається одне печиво в упаковці, то Зеник не помітить її дій.

3) Робота з двійковою системою числення:

$1. y_1 = 41$   
 $2. x_1 = 42$   
 $3. 41_2 = 1000111$   
 $4. 42_2 = 101010$   
 $5. +1000111$   
 $6. -1000111$   
 $7. 1000111$   
 $8. 1000111$   
 $9. K = 34$   
 $10. 34_{16} = 22$

## 5. Результати виконання завдань, тестування та фактично затрачених час:

```

C:\Users\User\Desktop\lpnu> epic1> C:\practice_work_task_1_anastasiia_humeniuk.cpp.cpp >
1 #include <iostream>
2 #include <cmath>
3 #include <cstdio>
4 #include <locale.h>
5
6 int main() {
7     setlocale(LC_ALL, "ukr");
8
9     double P, r, A;
10     extern "C" inline int _cdecl printf(const char *_Format, ...)
11
12     printf("Введіть початкову суму інвестицій (P): ");
13     scanf("%lf", &P);
14
15     printf("Введіть річну процентну ставку : ");
16     scanf("%lf", &r);
17
18     printf("Введіть кількість нарахувань відсотків на рік (n): ");
19     scanf("%d", &n);
20
21     printf("Введіть тривалість інвестицій у роках (t): ");

```

PROBLEMS OUTPUT PORTS DEBUG CONSOLE TERMINAL  
 Введіть початкову суму інвестицій (P): 500  
 Введіть річну процентну ставку : 3  
 Введіть кількість нарахувань відсотків на рік (n): 12  
 Введіть тривалість інвестицій у роках (t): 3  
 Початкова сума: 500,00  
 Загальна сума інвестицій після 3 років: 547,03  
 Зароблена сума: 47,03  
 PS C:\Users\User>

Фактично затрачений час: 30 хв.

[https://github.com/artificial-intelligence-department/ai\\_programming\\_playground\\_2024/blob/epic\\_1\\_practice\\_and\\_labs\\_anastasiia\\_humeniuk/ai\\_11/anastasiia\\_humeniuk/epic1/practice\\_work\\_task\\_1\\_anastasiia\\_humeniuk.cpp](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/blob/epic_1_practice_and_labs_anastasiia_humeniuk/ai_11/anastasiia_humeniuk/epic1/practice_work_task_1_anastasiia_humeniuk.cpp)

```
C: > Users > User > Desktop > lnu > epic1 > self_practice_work_algo

1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int n;
6      cin >> n;
7
8      long long total_cookies = 0;
9      int cookies_in_pack;
10
11      while (n-- > 0) {
12          cin >> cookies_in_pack;
13          total_cookies += cookies_in_pack - 1;
14      }
15
16      cout << total_cookies << endl;
17      return 0;
18  }
19
```

PROBLEMS OUTPUT PORTS DEBUG CONSOLE TERMINAL

```
PS C:\Users\User> & 'c:\Users\User\.vscode\extensions\ms-vscode.cmake-tools\bin\cmake.exe' '--stdin=Microsoft-MIEngine-In-hdsuifob.etm' '--s
ine-Error-0ted11sh.s2p' '--pid=Microsoft-MIEngine-Pid-p3jp

4
4 7 47 74
128
PS C:\Users\User>
```

Фактично затрачений час: 40 хв.

[https://github.com/artificial-intelligence-department/ai\\_programming\\_playground\\_2024/blob/epic\\_1\\_practice\\_and\\_labs\\_anastasiia\\_humeniuk/ai\\_11/anastasiia\\_humeniuk/epic1/self\\_practice\\_work\\_algotester\\_task\\_1\\_anastasiia\\_humeniuk.cpp](https://github.com/artificial-intelligence-department/ai_programming_playground_2024/blob/epic_1_practice_and_labs_anastasiia_humeniuk/ai_11/anastasiia_humeniuk/epic1/self_practice_work_algotester_task_1_anastasiia_humeniuk.cpp)

Результат на Algotester:

Створено	Компілятор	Результат	Час (сек.)	Пам'ять (МіБ)	Дії
8 днів тому	C++ 23	Зарховано	0.036	1.199	<a href="#">Перегляд</a>
8 днів тому	C++ 23	Неправильна відповідь 21	0.036	1.203	<a href="#">Перегляд</a>

## 6. Кооперація з командою:





**Висновок:** виконуючи завдання першого епіку, я встановила Visual Studio Code та налаштувала у ньому всі потрібні інструменти. Я отримала практичний досвід у роботі з FlowCharts і в організації командної роботи за допомогою Trello. Опанувала тему систем числення, попрактикувавшись у діях із числами у двійковій системі. Ознайомила з Linux-подібним терміналом, що заклало основу для подальшої роботи в ньому. Також зробила висновки щодо планування часу на виконання поставлених задач (фактичний час перевищив запланований). Встановила Git, вивчила основні команди для роботи з цим інструментом та успішно створила гілку на GitHub, що стало завершальним етапом моєї роботи.