

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



Звіт

про виконання лабораторних та практичних робіт блоку № 4

На тему: «Одновимірні масиви. Двовимірні Масиви. Вказівники та Посилання. Динамічні масиви. Структури даних. Вкладені структури.

Алгоритми обробки та робота з масивами та структурами.»

з дисципліни: «Основи програмування»

до:

ВНС Лабораторної Роботи №4

ВНС Лабораторної Роботи №5

Алготестер Лабораторної Роботи №2

Алготестер Лабораторної Роботи №3

Практичних Робіт до блоку №4

Виконав:

Студент групи ШІ-11

Кравченко Артем Миколайович

Тема:

Одновимірні масиви. Двовимірні Масиви. Вказівники та Посилання. Динамічні масиви. Структури даних. Вкладені структури. Алгоритми обробки та робота з масивами та структурами.

Мета:

Навчитися працювати з одновимірними і двовимірними масивами, розібратися з вказівниками і посиланнями, щоб зрозуміти, як управляти динамічною пам'яттю.

Теоретичні відомості:

1. Класи пам'яті у C++
2. Вступ до Масивів і Вказівників
3. Одновимірні Масиви
4. Вказівники та Посилання
5. Двовимірні Масиви
6. Динамічні Масиви
7. Структури Даних
8. Вкладені Структури
9. Використання структур
10. Алгоритми обробки та робота з масивами та структурами

Індивідуальний план опрацювання теорії:

1. <http://cpp.dp.ua/klasy-pam-yati-u-c-builder/>
2. <https://acode.com.ua/urok-86-vkazivnyky-i-masyvy/>
3. <https://acode.com.ua/urok-77-masyvy/>
4. <https://acode.com.ua/urok-92-posylannya/>
5. <https://acode.com.ua/urok-81-bagatovymirni-masyvy/>
6. <https://acode.com.ua/urok-90-dynamichni-masyvy/>
7. <https://acode.com.ua/urok-64-struktury/>
8. https://www.youtube.com/watch?v=999IE-6b7_s
9. https://www.youtube.com/watch?v=999IE-6b7_s
10. <https://dou.ua/forums/topic/40645/>

Виконання роботи:

Завдання 1: VNS Lab 4 - Task 1. Варіант - 24

- 1) Реалізувати з використанням масиву однонаправлене кільце (перегляд можливий зліва направо, від останнього елемента можна перейти до першого).
- 2) Роздрукувати отриманий масив, починаючи з K -ого елемента й до $K-1$.
- 3) Впорядкувати елементи за зростанням
- 4) Знищити з кільця парні елементи.
- 5) Роздрукувати отриманий масив, починаючи з K -ого елемента і до $K-1$.

Завдання 2: VNS Lab 5 - Task 1. Варіант - 24

Визначити скільки елементів двовимірного масиву більші від будь-якого елемента на головній діагоналі.

Завдання 3: Algotester lab 2 Варіант 2

У вас є масив r розміром N . Також вам дано 3 цілих числа. Спочатку ви маєте видалити з масиву ці 3 числа, які вам дані. Після цього перетворити цей масив у масив сум, розміром $N_{\text{new}} - 1$ (розмір нового масиву після видалення елементів), який буде відображати суми сусідніх елементів нового масиву. Далі необхідно вивести масив сум на екран.

Завдання 4: Algotester lab 3 Варіант 3

Вам дана стрічка s . Ваше завдання зробити компресію стрічки, тобто якщо якась буква йде більше одного разу підряд у стрічці замінити її на букву + кількість входжень підряд.

Завдання 5: Class Practice Work

Реалізувати програму, яка перевіряє, чи дане слово чи число є паліндромом за допомогою рекурсії.

Паліндром — це послідовність символів, яка однаково читається вперед і назад (наприклад, «radar», «level», «12321»).

Вимоги:

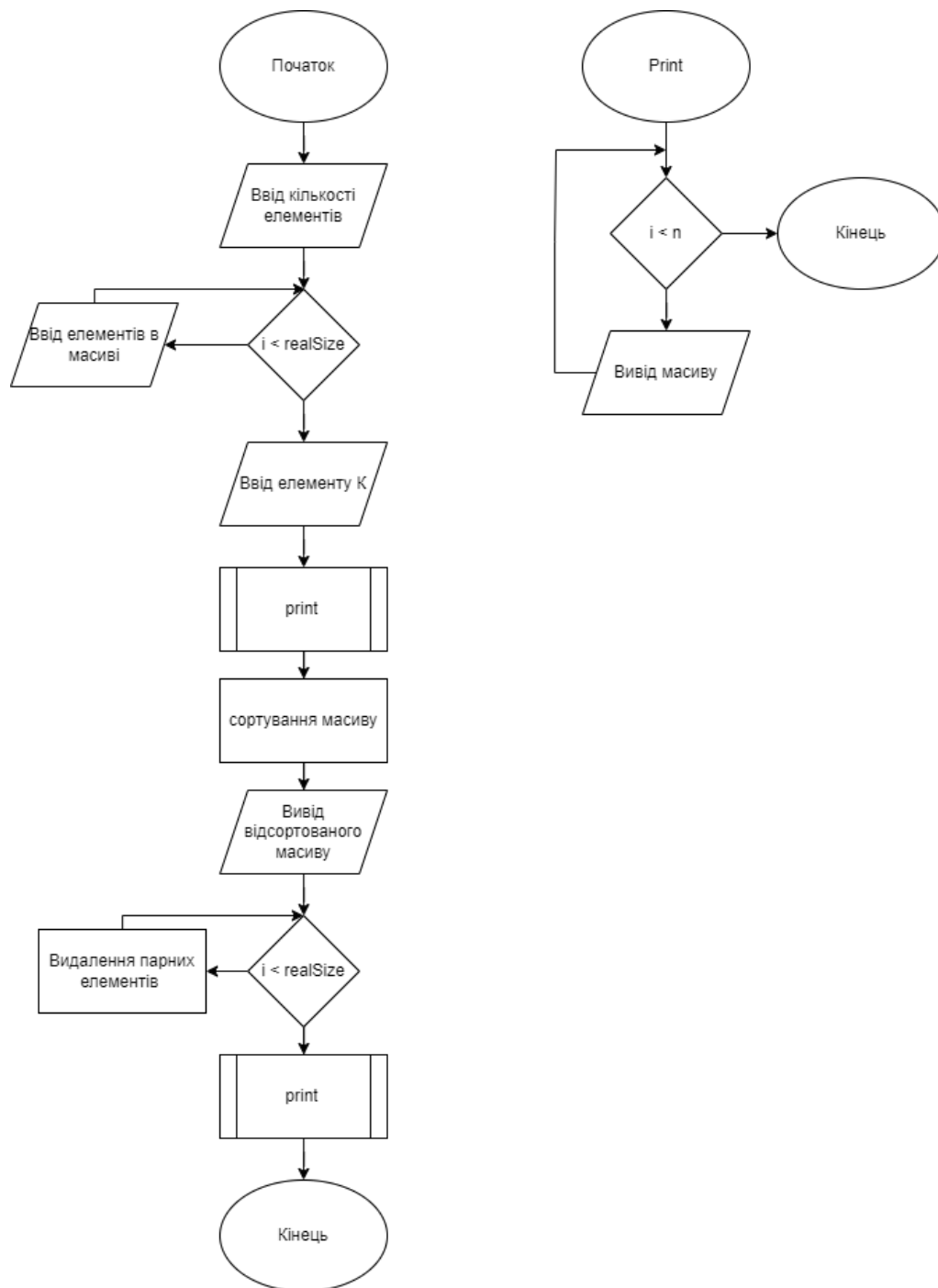
1. Визначення функції:
 - a. Реалізуйте рекурсивну функцію *isPalindrome*, яка перевіряє, чи заданий рядок є паліндромом.
2. Приклад визначення функції:
 - a. *bool isPalindrome(const string& str, int start, int end);*
3. Перевантаження функцій:
 - a. Перевантажте функцію *isPalindrome* для роботи з цілими значеннями.
 - b. *bool isPalindrome(ціле число);*
4. Рекурсія:
 - a. Рекурсивна функція для рядків перевірить символи в поточній початковій і кінцевій позиціях. Якщо вони збігаються, він буде рекурсивно перевіряти наступні позиції, поки початок не перевищить кінець, після чого рядок буде визначено як паліндром.

Завдання 6: Self Practice Work

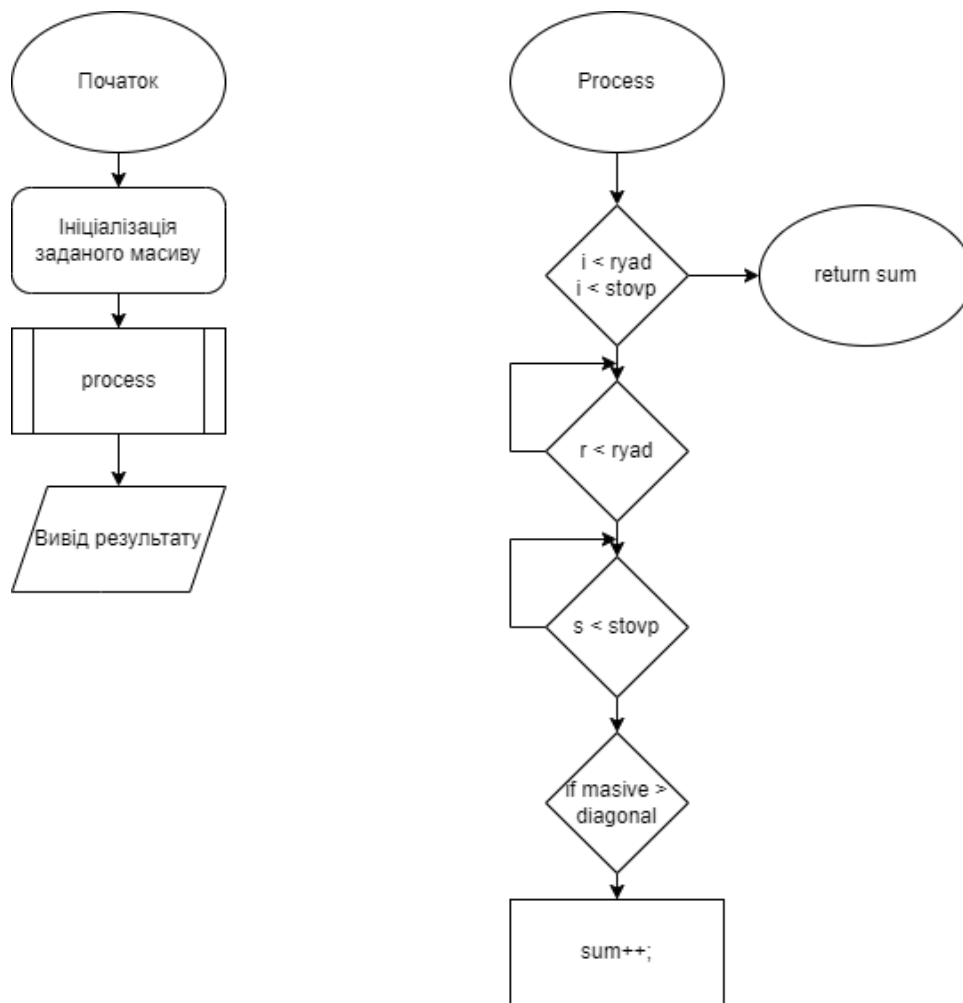
Одного разу до Ужгорода на літню школу з алгоритмічного програмування приїхали n студентів, що сформували k команд. Відомо, що кожна команда складається з одного, двох або трьох студентів.

Вам необхідно визначити, скільки студентів було в кожній із команд

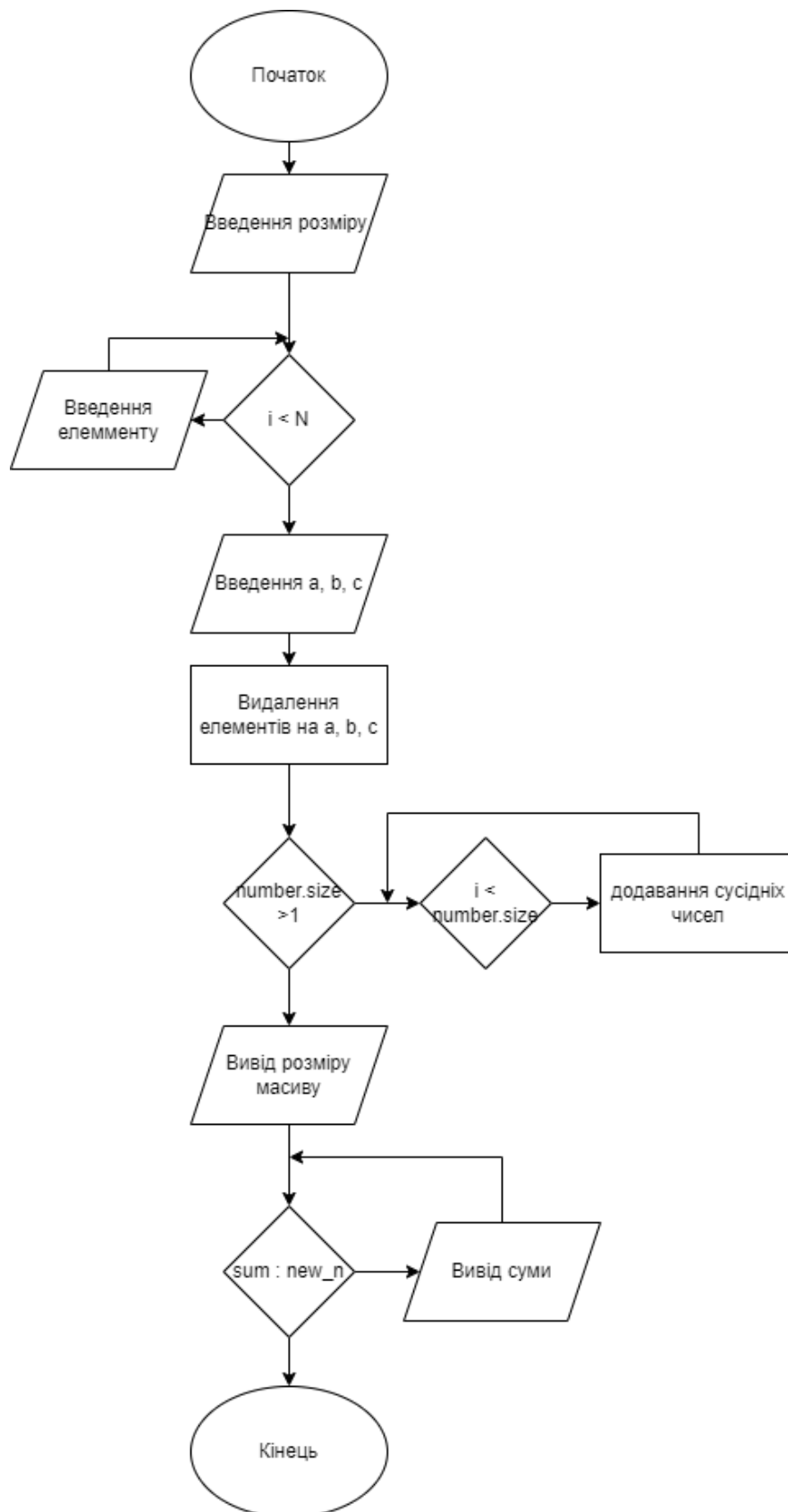
Дизайн та планувальна оцінка часу виконання завдань:
Завдання 1: Запланований час виконання 20 хвилин.



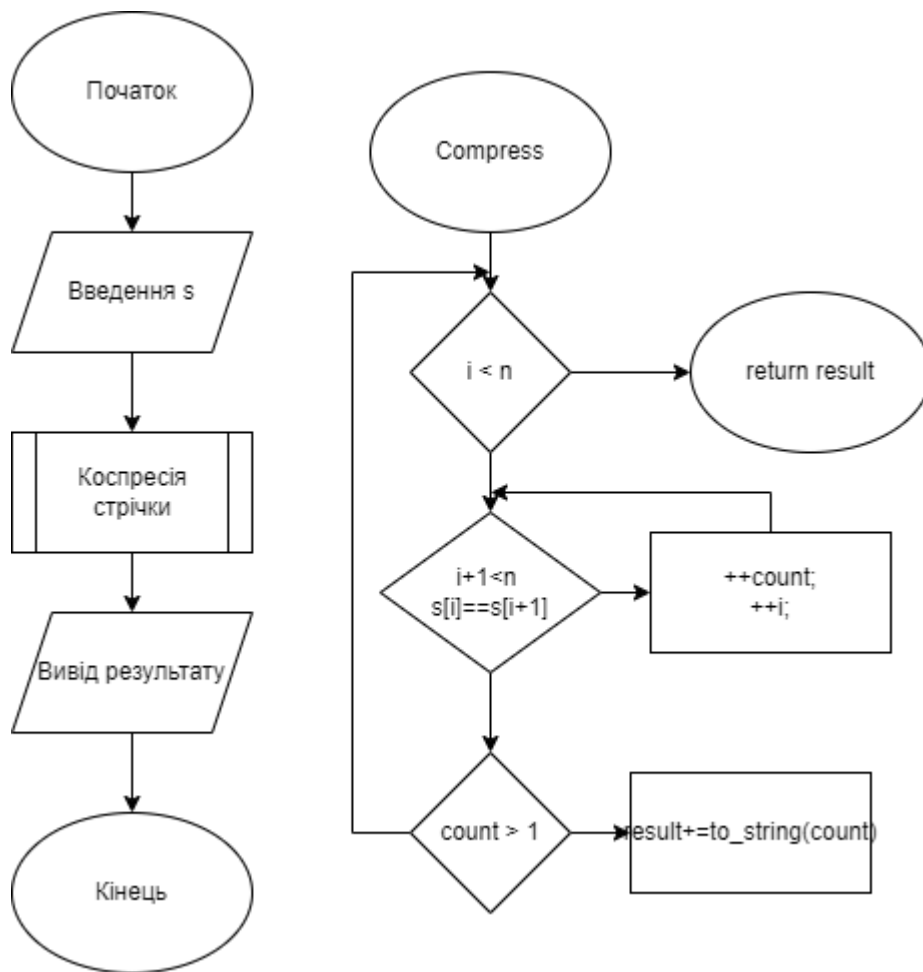
Завдання 2: Запланований час виконання 1-1.5 години.



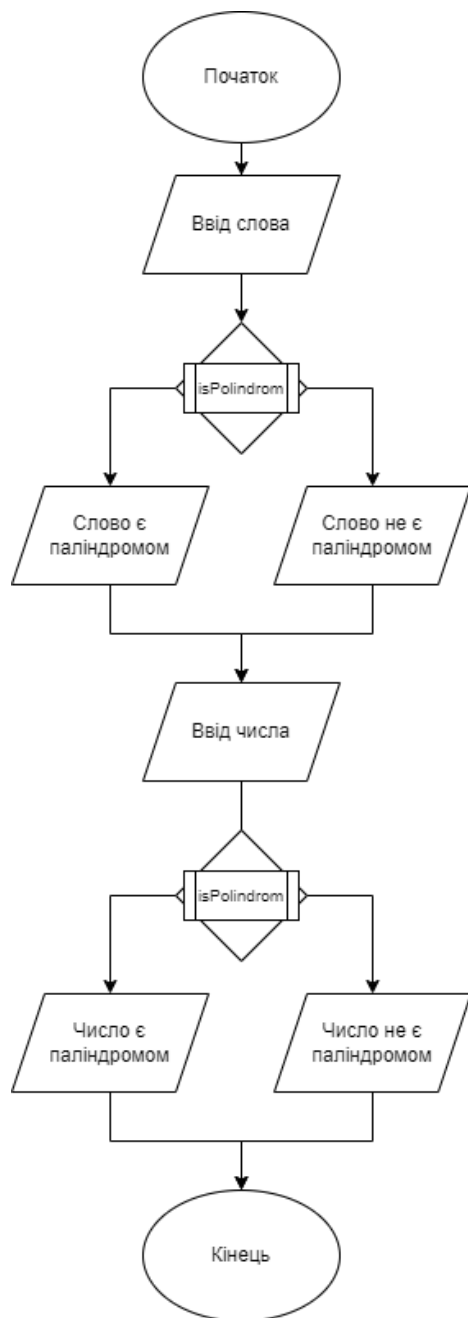
Завдання 3: Запланований час виконання 1 година.



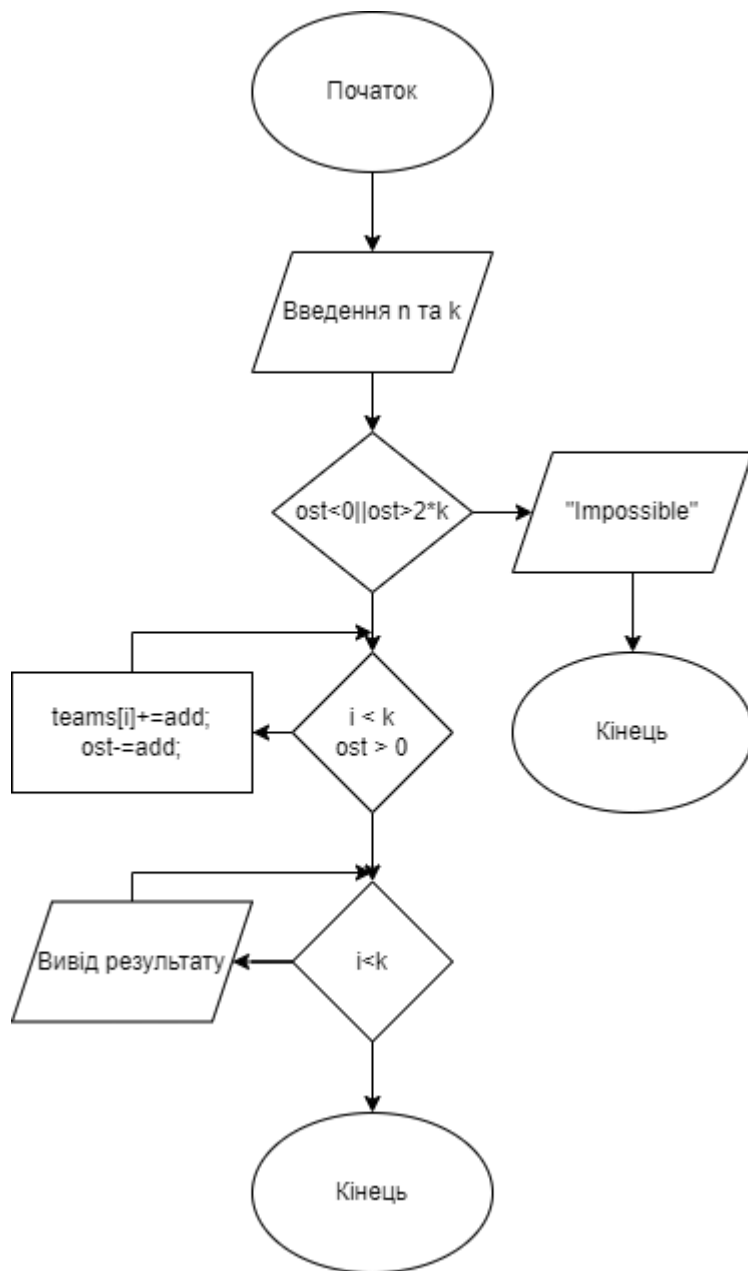
Завдання 4: Запланований час виконання 40-50 хвилин.



Завдання 5: Запланований час виконання 1-2 години.



Завдання 6: Запланований час виконання 20-25 хвилин.



Код програм з посиланням на зовнішні ресурси:

Завдання 1:

```
1  #include <iostream>
2  #include <vector>
3  #include <algorithm>
4
5  using namespace std;
6
7  void print(const int massive[], int n, int startIndex){
8      for (int i = 0; i < n; i++)
9      {
10         cout << massive[(startIndex + i) % n] << " ";
11     }
12     cout << endl;
13 }
14
15 int main(){
16     const int N = 100;
17     int massive[N];
18     int realSize;
19     cout << "Введіть кількість елементів масиву: ";
20     cin >> realSize;
21     cout << "Введіть елементи в масив: ";
22     for (int i = 0; i < realSize; i++)
23     {
24         cin >> massive[i];
25     }
26
27     int K;
28     cout << "Введіть елемент K: ";
29     cin >> K;
30
31     print(massive, realSize, K);
32
33     sort(massive, massive + realSize);
34     cout << "Відсортований масив: ";
35     for (int i = 0; i < realSize; i++)
36     {
37         cout << massive[i] << " ";
38     }
39     cout << endl;
```

```
40
41     int new_massive[N];
42     int new_size = 0;
43     for (int i = 0; i < realSize; i++)
44     {
45         if (massive[i] % 2 != 0)
46         {
47             new_massive[new_size++] = massive[i];
48         }
49     }
50
51     cout << "Масив після видалення парних чисел: ";
52     print(new_massive, new_size, K);
53
54     return 0;
55 }
```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/195/files#diff-5f9d4fa398251c5ac07ca38e955291eaf89d42418bb3198bd8794821d534da95

Завдання 2:

```
1  #include <iostream>
2
3  using namespace std;
4
5  int process(const int masive[][3], int ryad, int stovp){
6      int sum = 0;
7      for (int i = 0; i < ryad && i < stovp; i++)
8      {
9          int diagonal = masive[i][i];
10         for (int r = 0; r < ryad; r++)
11         {
12             for (int s = 0; s < stovp; s++)
13             {
14                 if (masive[r][s] > diagonal)
15                 {
16                     sum++;
17                 }
18             }
19         }
20     }
21 }
22
23 return sum;
24 }
25
26 int main(){
27     const int ryad = 3;
28     const int stovp = 3;
29     int masive[ryad][stovp] = {
30         {1, 2, 3},
31         {4, 5, 6},
32         {7, 8, 9}
33     };
34     int result = process(masive, ryad, stovp);
35     cout << "Результат: " << result << endl;
36 }
```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/195/files#diff-00b81d87c6f83e4d2ddb5788ca73e8a18102602a926d5340213324a9a9229a76

Завдання 3:

```
1  #include <iostream>
2  #include <vector>
3  #include <algorithm>
4
5  using namespace std;
6
7  int main(){
8      int N;
9      cin >> N;
10     vector<int> number(N);
11     for (int i = 0; i < N; i++)
12     {
13         cin >> number[i];
14     }
15     int a, b, c;
16     cin >> a >> b >> c;
17
18     number.erase(remove(number.begin(), number.end(), a), number.end());
19     number.erase(remove(number.begin(), number.end(), b), number.end());
20     number.erase(remove(number.begin(), number.end(), c), number.end());
21
22     vector<int> new_number;
23     if (number.size() > 1)
24     {
25         for (size_t i = 0; i < number.size() - 1; i++)
26         {
27             new_number.push_back(number[i] + number[i+1]);
28         }
29     }
30
31     cout << new_number.size() << endl;
32     for (int sum : new_number)
33     {
34         cout << sum << " ";
35     }
36     cout << endl;
37
38     return 0;
39 }
```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/195/files#diff-ac5c6e82e4760fa748dd6bc3335dcb1c439a319415a183e0b600438f4b5f4beb

Завдання 4:

```
1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  string compress (const string& s){
7      string result;
8      int n = s.length();
9      for (int i = 0; i < n; i++)
10     {
11         int count = 1;
12         while (i + 1 < n && s[i] == s[i + 1])
13         {
14             ++count;
15             ++i;
16         }
17         result += s[i];
18
19         if (count > 1)
20         {
21             result += to_string(count);
22         }
23     }
24     return result;
25 }
26
27
28 int main(){
29     string s;
30     cin >> s;
31     string result_c = compress(s);
32     cout << result_c << endl;
33     return 0;
34 }
```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/195/files#diff-283b5d6b7f1e78d4120b1abcf5b72b630caa0f1f1df11f4aaa87d7bfc6a9c261

Завдання 5:

```
1  #include <iostream>
2  #include <string>
3  #include <algorithm>
4
5  using namespace std;
6
7  bool isPolindrom (const string& word){
8      string reversedWord = word;
9      reverse(reversedWord.begin(), reversedWord.end());
10     return word == reversedWord;
11 }
12
13 bool isPolindrom( int number){
14     string numberStr = to_string(number);
15     string reversedStr = numberStr;
16     reverse(reversedStr.begin(), reversedStr.end());
17     return numberStr == reversedStr;
18 }
19
20 int main(){
21     string word;
22     cout << "Введіть слово для перевірки: ";
23     cin >> word;
24     if (isPolindrom(word))
25     {
26         cout << "Слово є паліндромом." << endl;
27     }else{
28         cout << "Слово не є паліндромом." << endl;
29     }
30
31     int number;
32     cout << "Введіть число: ";
33     cin >> number;
34
35     if (isPolindrom(number))
36     {
37         cout << "Число є паліндромом." << endl;
38     }else{
39         cout << "Число не є паліндромом." << endl;
40     }
41     return 0;
42 }
```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/195/files#diff-349bfe4d070380ce5f7a56a422075df196e2d73f973717c1a248ee69ec44c0bc

Завдання 6:

```
1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5  int main() {
6      int n, k;
7      cin >> n >> k;
8
9      vector<int> teams(k, 1);
10     int ost = n - k;
11
12     if (ost < 0 || ost > 2 * k) {
13         cout << "Impossible" << endl;
14         return 0;
15     }
16
17     for (int i = 0; i < k && ost > 0; ++i) {
18         int add = min(2, ost);
19         teams[i] += add;
20         ost -= add;
21     }
22
23     for (int i = 0; i < k; ++i) {
24         cout << teams[i] << (i == k - 1 ? "\n" : " ");
25     }
26
27     return 0;
28 }
```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/195/files#diff-6559ac963651f4cee54e7dd2c0f5a9c769888588a79f19983505f8fdf2eb914e

Результат виконання завдань, тестування та фактично витрачений час:

Завдання 1:

```
Введіть кількість елементів у масиві: 5
Введіть елементи в масиві: 2 3 5 6 7
Введіть елемент K: 2
5 6 7 2 3
Відсортований масив: 2 3 5 6 7
Масив після видалення парних чисел: 7 3 5
PS D:\VS Code project\C ++>
```

Фактично витрачений час: 25-30 хвилин.

Завдання 2:

```
PS D:\VS Code project\C ++> & 'c:\Users\akrav\.vscode\extensions\ms-vscode.cpptools-1.23.0
xe' '--stdin=Microsoft-MIEngine-In-ftxfvpfl.qed' '--stdout=Microsoft-MIEngine-Out-0ioupz0y.
1' '--pid=Microsoft-MIEngine-Pid-3sajn2e1.b1h' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--i
Результат: 12
PS D:\VS Code project\C ++>
```

Фактично витрачений час: 20-25 хвилин

Завдання 3:

```
6
1 2 3 4 5 7
4 5 6
3
3 5 10
PS D:\VS Code project\C ++>
```

Фактично витрачений час: 55-60 хвилин.

Завдання 4:

```
T --pid=Microsoft-MIEngine-Pid-m1gJ2041.T4C --dbgExe=C:\msy
AAAAQQQCCDDDD
A4Q3C3D3
PS D:\VS Code project\C ++>
```

Фактично витрачений час: 1 година.

Завдання 5:

```
Введіть слово для перевірки: level  
Слово є паліндромом.  
Введіть число: 1234  
Число не є паліндромом.  
PS D:\VS Code project\C ++> █
```

Фактично витрачений час: 1 година.

Завдання 6:

```
7 4  
3 2 1 1  
PS D:\VS Code project\C ++> █
```

Фактично витрачений час: 40-45 хвилин.

Висновок: Я навчився працювати з одновимірними та двовимірними масивами, зрозумів принципи використання вказівників і посилань для керування динамічною пам'яттю. Також ознайомився з основами роботи зі структурами даних і алгоритмами обробки масивів, що допоможе в подальшому ефективно працювати з даними.