

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



Звіт

про виконання лабораторних та практичних робіт блоку № 3
На тему: «Цикли. Вкладені Цикли. Завершення виконання циклів.
Функції. Простір імен. Перевантаження функцій. Функції з змінною
кількістю параметрів (еліпсис). Рекурсія. Вбудовані функції.»
з дисципліни: «Основи програмування»

до:

ВНС Лабораторної Роботи № 2
ВНС Лабораторної Роботи № 3
ВНС Лабораторної Роботи № 7
Практичних Робіт до блоку № 3

Виконав):
Студент групи ШІ-11
Кравченко Артем Миколайович

Тема роботи:

Цикли. Вкладені Цикли. Завершення виконання циклів. Функції. Простір імен. Перевантаження функцій. Функції з змінною кількістю параметрів (еліпсис). Рекурсія. Вбудовані функції.

Мета роботи:

Ознайомитися з основами роботи з циклами, вкладеними циклами, завершенням циклів, функціями, простором імен, перевантаженням функцій, рекурсією та вбудованими функціями. Розвинути навички написання ефективного коду, вивчити принципи роботи з параметрами різної кількості, організування коду за допомогою простору імен, а також застосування рекурсії і вбудованих функцій для розв'язання практичних задач.

Теоретичні відомості:

- Тема №1: Введення в Цикли та їх Види в C++
- Тема №2: Управління Виконанням Циклів.
- Тема №3: Вкладені Цикли.
- Тема №4: Основи Функцій у C++.
- Тема №5: Перевантаження Функцій та Простір Імен.
- Тема №6: Розширені Можливості Функцій.
- Тема №7: Вбудовані Функції в C++.

Індивідуальний план опрацювання теорії:

Тема №1: Введення в Цикли та їх Види в C++ та Управління Виконанням Циклів.

[https://www.youtube.com/watch?v=LYIPTmN37SU\](https://www.youtube.com/watch?v=LYIPTmN37SU)

Тема №3: Вкладені Цикли.

<https://www.youtube.com/watch?v=70CkVC2vVCQ>

Тема №4: Основи Функцій у C++.

<https://www.youtube.com/watch?v=G8P6SvdqU9s>

Тема №5: Перевантаження Функцій та Простір Імен
<https://acode.com.ua/urok-108-perevantazhennya-funktsij/>

Тема №6: Розширені Можливості Функцій та Вбудовані Функції.
<https://m.youtube.com/watch?v=q3GPKxRsgsM>

Виконання роботи:

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

Використовуючи оператор циклу, знайти суму елементів, зазначених у конкретному варіанті. Результат надрукувати, надавши відповідний заголовок.

24) Знайти суму 9 членів ряду, у якому

$$a_n = e^{-\sqrt{n}}$$

Завдання 2: VNS Lab 3 - Task 1. Варіант - 24

Для x , що змінюється від a до b з кроком $(b-a)/k$, де $(k=10)$, обчислити функцію $f(x)$, використовуючи її розклад в степеневий ряд у двох випадках:

а) для заданого n ;

б) для заданої точності ϵ ($\epsilon=0.0001$).

Для порівняння знайти точне значення функції.

24	$y = \ln\left(\frac{1}{2+2x+x^2}\right)$	$-2 \leq x \leq -0,1$	40	$S = -(1+x)^2 + \frac{(1+x)^4}{2} + \dots + (-1)^n \frac{(1+x)^{2n}}{n}$
----	--	-----------------------	----	--

Завдання 3: VNS Lab 7 - Task 1. Варіант - 24

Розв'язати зазначене у варіанті завдання, використовуючи функції зі змінною кількістю параметрів.

24. Написати функцію (або макровизначення), що знаходить довжину сторони за координатами його точок.. Написати функцію `square`, що обчислює площу трикутника, заданого координатами вершин. Написати функцію `squaren` с змінною кількістю параметрів, що визначає площу опуклого багатокутника, заданого координатами своїх вершин.

Завдання 4: VNS Lab 7 - Task 2. Варіант - 24

Написати перевантажені функції й основну програму, що їх викликає.
24.

- а) для масиву цілих чисел знищує всі парні елементи з масиву;
- б) для рядка знищує всі парні слова.

Завдання 5: Class Practice Work

Ви створюєте просту програму керування бібліотекою. Книги в бібліотеці є, користувачі можуть їх взяти або повернути.

Програма повинна вміти

- Перерахувати всі книги.
- Дозволити взяти книгу (за наявності).
- Дозволити повернення книги.

Вимоги:

1. `while`: продовжувати працювати, доки користувач не вирішить вийти.
2. `do while`: Після кожної операції (позичити, повернути, перерахувати) запитуйте користувача, чи хоче він виконати іншу операцію. Якщо так, поверніться назад.
3. `for`: список усіх книг за допомогою циклу.
4. `for each`: перевірити наявність кожної книги.
5. `goto`: якщо користувач вводить неправильний вибір, використовуйте `goto`, щоб перенаправити його до головного меню.

Завдання 5: Self Practice Work

Коли Коля та Вася прийшли робити ремонт на «Екстралогіку» — першим, що вони побачили в офісі, був стіл для настільного тенісу. Поки всі інші працювали, Коля та Вася вирішили пограти. Через декілька годин прийшов директор і накричав на заробітчан через те, що вони нічим не займаються. Тож Вася і Коля мушили йти працювати.

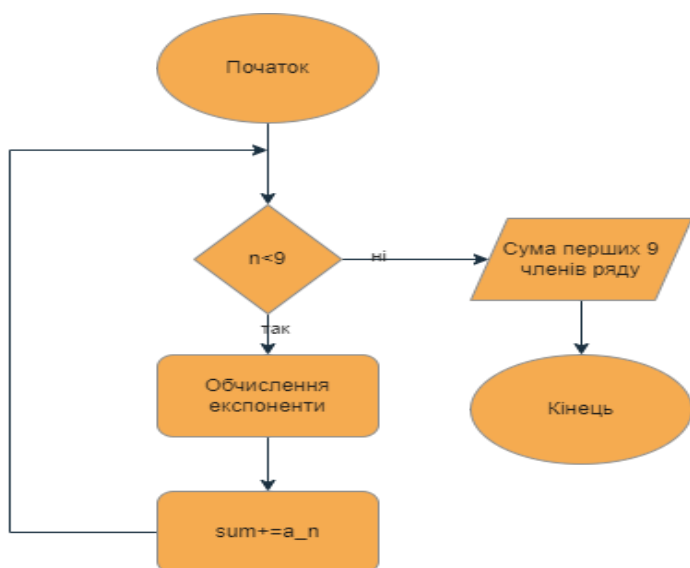
По дорозі вони сперечалися, хто ж виграв і з яким рахунком. Оскільки вони записували результати кожної подачі, то це можна порахувати. Але оскільки гра тривала дуже довго — порахувати це вручну дуже тяжко.

Всього відбулося n подач. Про кожну з них ми знаємо, хто переміг. За виграну подачу гравець отримує одне очко. Партія вважається виграною, коли один з гравців набере не менше одинадцяти очок з перевагою щонайменше у два очки. Наприклад, за рахунків 11:9, 4:11, 15:13 партія закінчується, а за рахунків 11:10 та 99:98 — ні. Як тільки Коля і Вася закінчили одну партію — вони починають іншу.

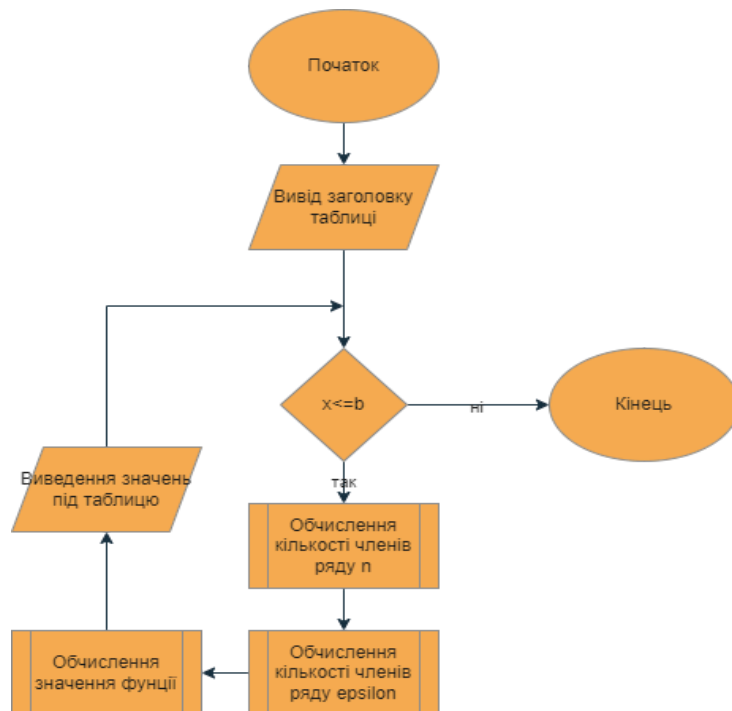
Знаючи, хто переміг кожної подачі — виведіть загальний рахунок по партіях в грі Коля-Вася. А якщо вони не дограли останню партію, то і її рахунок теж.

Дизайн та планувальна оцінка часу виконання завдань:

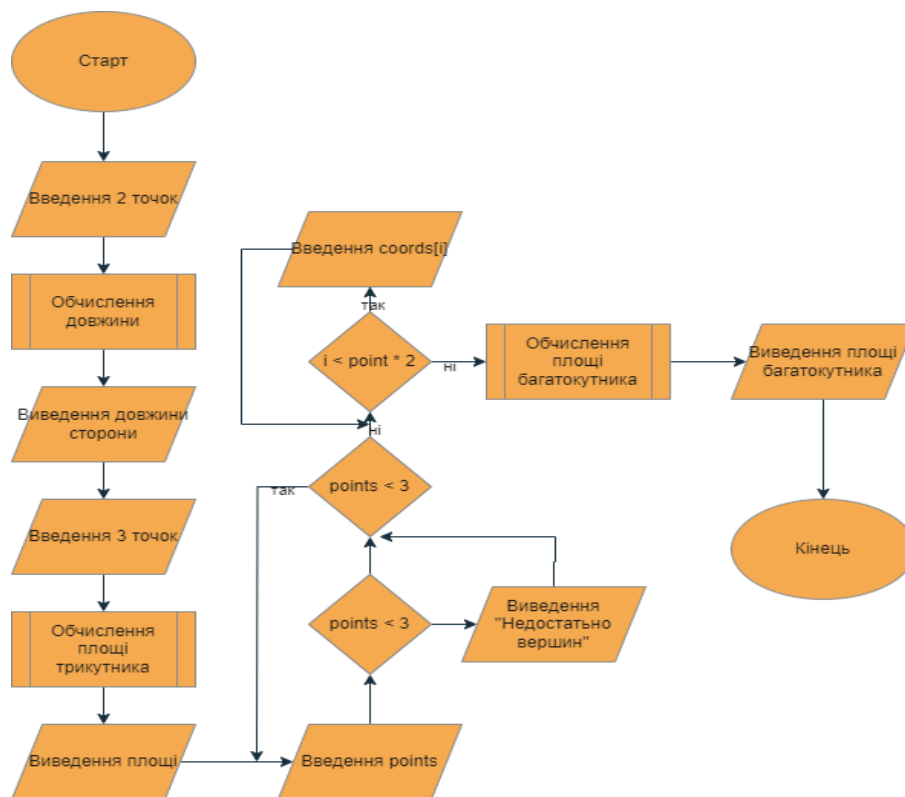
Завдання 1: Запланований час виконання 20 хвилин.



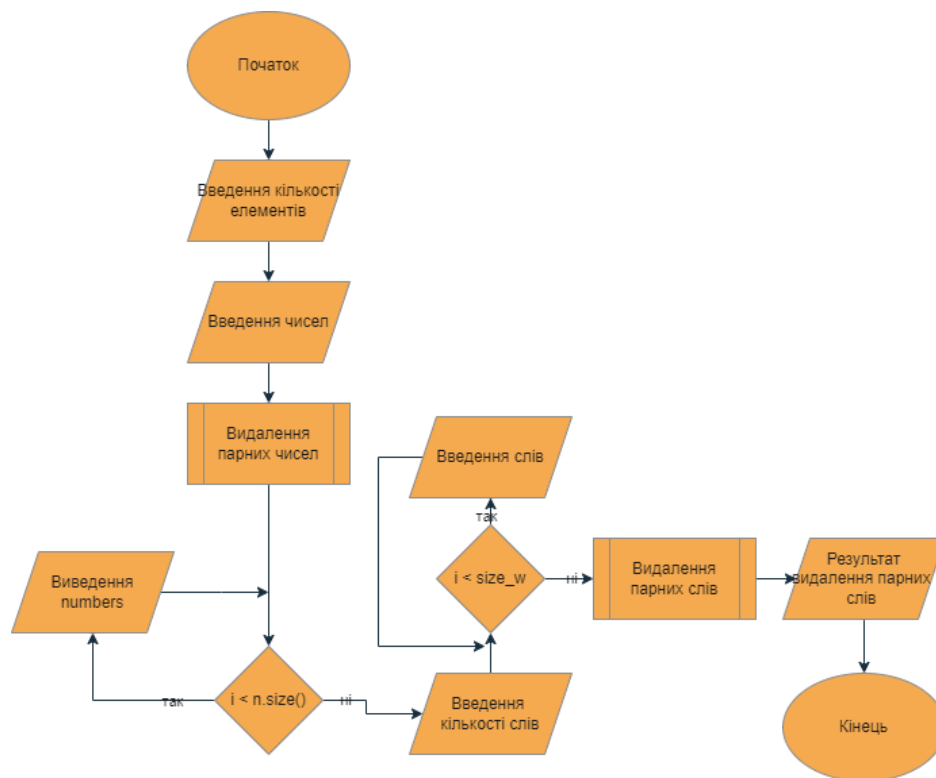
Завдання 2: Запланований час виконання 1-1.5 години.



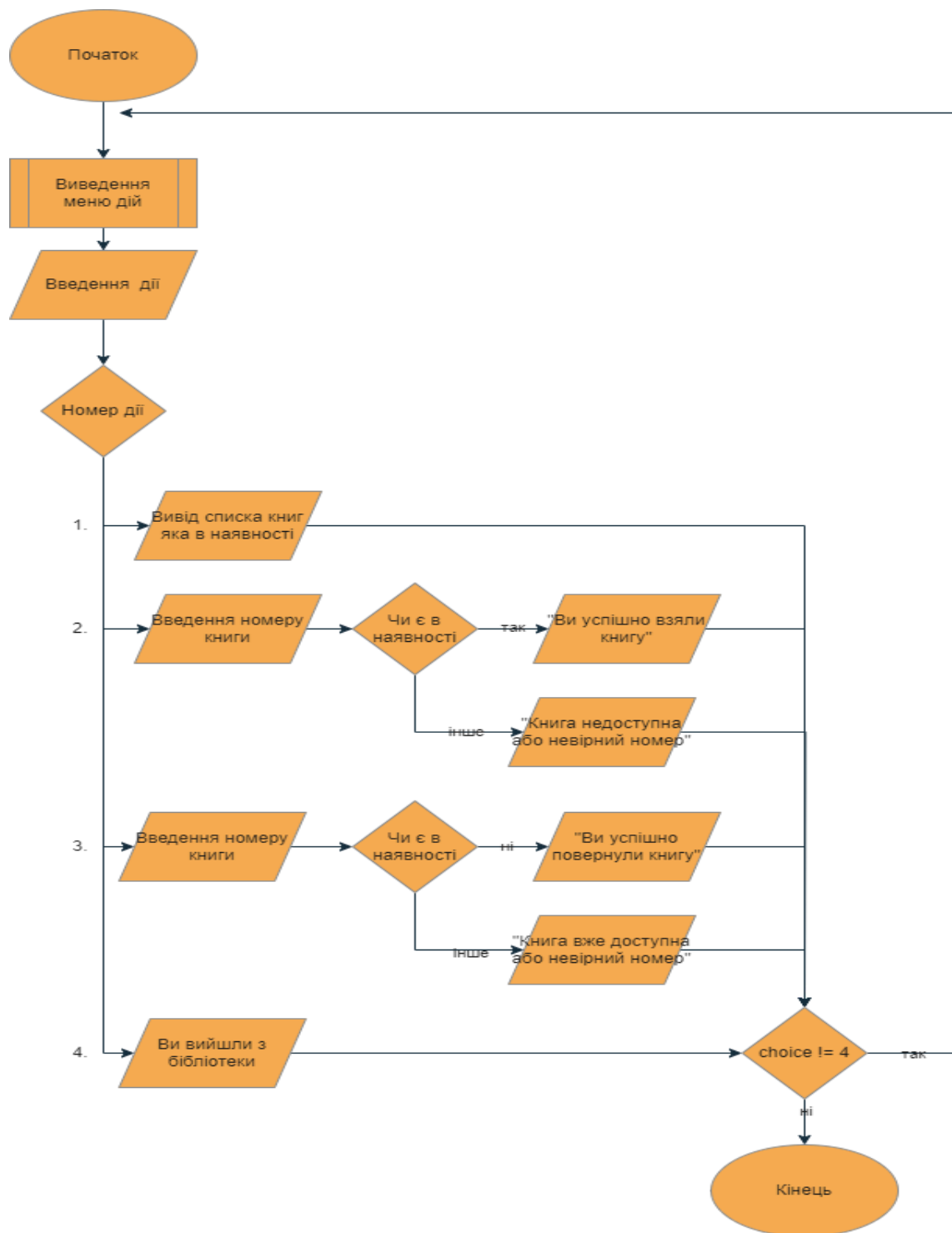
Завдання 3: Запланований час виконання 1 година.



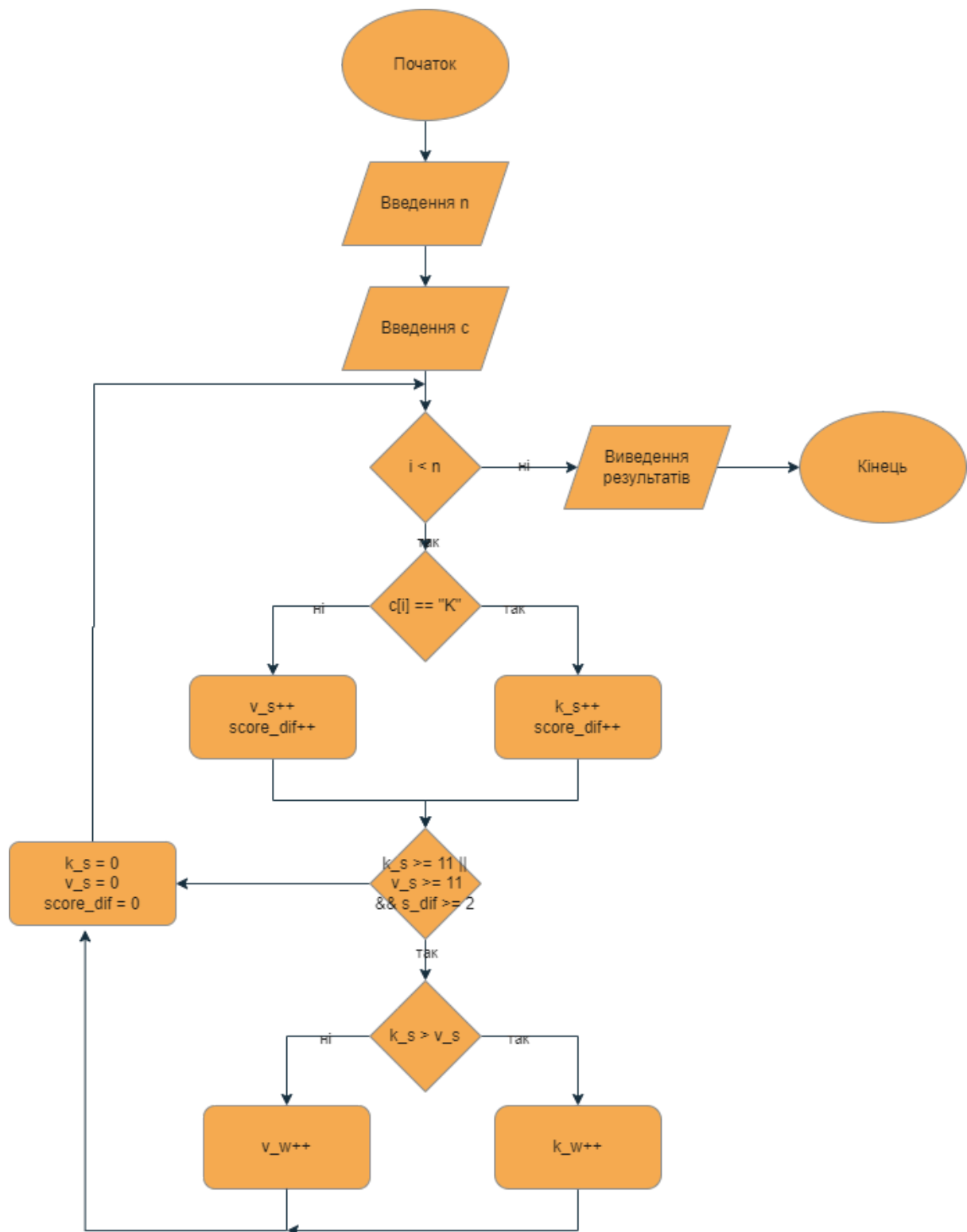
Завдання 4: Запланований час виконання 40-50 хвилин.



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 <cmath>
3
4  using namespace std;
5
6  int main(){
7      double sum = 0;
8
9      for (int n = 0; n < 9; n++)
10     {
11         double a_n = exp(-sqrt(n));
12         sum += a_n;
13     }
14
15     cout << "Сума перших 9 членів ряду: " << sum << endl;
16     return 0;
17 }
```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/196/files#diff-d9121435168188b3e3ab944faa9dbd585d970ce4b37d5e5f23c689150027706e

Завдання 2:

```
1  #include <iostream>
2  #include <cmath>
3  #include <iomanip>
4
5  using namespace std;
6
7  double ryad_n(double x, int n){
8      double sum = 0;
9      for (int i = 1; i < n; i++)
10     {
11         sum += pow(-1, i)*(pow((1+x),2*i)/i);
12     }
13
14     return sum;
15 }
16
17 double ryad_epsilon (double x, double epsilon){
18     double sum = 0;
19     double term;
20     int i = 1;
21     do
22     {
23         term = pow(-1, i)*(pow((1+x),2*i)/i);
24         sum += term;
25         i++;
26     } while (fabs(term) >= epsilon);
27     return sum;
28 }
29
30 double function (double x){
31     double znamenik = 2 + 2*x + pow(x,2);
32     if (znamenik <= 0)
33     {
34         return NAN;
35     }
36     return log( 1.0 / znamenik);
37 }
38
39 int main (){
```

```

39  int main (){
40      double a = -2;
41      double b = -0.1;
42      int k = 10;
43      double step = (b-a)/k;
44      int n = 40;
45      double epsilon = 0.0001;
46
47      cout << fixed << setprecision(6);
48      cout << "X\t\tSN\t\tSE\t\tV" << endl;
49
50      for (double x = a; x <= b; x += step)
51      {
52          double sn = ryad_n (x,n);
53          double se = ryad_epsilon (x,epsilon);
54          double y = function (x);
55
56          cout << x << "\t" << sn << "\t" << se << "\t" << y << endl;
57      }
58
59
60      return 0;
61  }

```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/196/files#diff-36795714ad4d128cdeaa5632ede221facfbacafd4eb9b6ef93a6b218a78b0ad1

Завдання 3:

```

1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6  double size(double x1, double x2, double y1, double y2){
7      return sqrt( pow((x2 - x1), 2) + pow((y2 - y1), 2));
8  }
9
10 double square(double x1, double x2, double x3, double y1, double y2, double y3){
11     return abs((x1 * (y2 - y3) + x2 * (y3 - y1) + x3 * (y1 - y2)) / 2.0);
12 }
13
14 double squren(int points, double* coords){
15
16     double x0 = coords[0];
17     double y0 = coords[1];
18     double area = 0;
19     double x_prev = x0;
20     double y_prev = y0;
21
22     for (int i = 1; i < points; ++i)
23     {
24         double x = coords[i * 2];
25         double y = coords[i * 2 + 1];
26         area += (x_prev * y - y_prev * x);
27         x_prev = x;
28         y_prev = y;
29     }
30
31     area += (x_prev * y0 - y_prev * x0);
32
33     return abs(area) / 2.0;
34 }
35
36 int main(){
37     double x1, x2, x3, y1, y2, y3;
38     int points;
39

```

```

40     cout << "введіть координати 2 точок (x1, y1, x2, y2):" << endl;
41     cin >> x1 >> y1 >> x2 >> y2;
42     cout << "довжина сторони: " << size(x1, x2, y1, y2) << endl;
43
44     cout << "введіть координати 3 точок трикутника (x1, y1, x2, y2, x3, y3):" << endl;
45     cin >> x1 >> y1 >> x2 >> y2 >> x3 >> y3;
46     cout << "Площа трикутника:" << square(x1, x2, x3, y1, y2, y3) << endl;
47
48
49     do
50     {
51         cout << "Введіть кількість вершин багатокутника:" << endl;
52         cin >> points;
53         if (points < 3 )
54         {
55             cerr << "Недостатньо вершин" << endl;
56         }
57     } while (points < 3);
58
59     double* coords = new double[points * 2];
60     cout << "Введіть координати вершин по черзі (x, y): \n";
61     for (int i = 0; i < points * 2; i++)
62     {
63         cin >> coords[i];
64     }
65
66     cout << "Площа багатокутника: " << squren(points, coords) << endl;
67
68     return 0;
69 }

```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/196/files#diff-a7f197135c67d7f8c1bb2b28020157d30bbd244b5686c1bd22f227b0df27eb42

Завдання 4:

```
1  #include <iostream>
2  #include <vector>
3  #include <string>
4  #include <algorithm>
5
6  using namespace std;
7
8  void numbers_rem(vector<int>& vec){
9      vec.erase(remove_if(vec.begin(), vec.end(), [](int x) { return (x % 2) == 0 ; }), vec.end());
10 }
11
12 void words_rem(vector<string>& vec){
13     vec.erase(remove_if(vec.begin(), vec.end(), [&](const string& word) {
14         auto index = &word - &vec[0];
15         return index % 2 != 0;
16     })), vec.end());
17 }
18
19 int main(){
20     int size_n = 0;
21     cout << "Введіть кількість елементів: "<< endl;
22     cin >> size_n;
23
24     vector<int> numbers(size_n);
25     cout << "Введіть " << size_n << " чисел: " << endl;
26     for (int i = 0; i < size_n; i++)
27     {
28         cin >> numbers[i];
29     }
30     numbers_rem(numbers);
31     cout << "Результат видалення парних чисел: " << endl;
32     for (int i = 0; i < numbers.size(); i++)
33     {
34         cout << numbers[i] << " " << endl;
35     }
36
37     int size_w = 0;
38     cout << "Введіть кількість слів: " << endl;
39     cin >> size_w;
```

```
40     cin.ignore();
41
42     vector<string> words(size_w);
43     cout << "Введіть " << size_w << " слів:" << endl;
44     for (int i = 0; i < size_w; i++)
45     {
46         getline(cin, words[i]);
47     }
48
49     words_rem(words);
50     cout << "Результат видалення парних слів: " << endl;
51     for (const string& word : words)
52     {
53         cout << word << " ";
54     }
55
56
57     return 0;
58
59 }
```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/196/files#diff-d34ff7df7795f8946ad6e938cea06b4d552535b5f6bbf9553db2d2f523fb474c

Завдання 5:

```
1  #include <iostream>
2  #include <vector>
3  #include <string>
4
5  using namespace std;
6
7  void display_menu(){
8      cout << "\nМеню бібліотеки:\n";
9      cout << "1. Переглянути всі книги\n";
10     cout << "2. Взяти книгу\n";
11     cout << "3. Повернути книгу\n";
12     cout << "4. Вийти\n";
13     cout << "Оберіть опцію: ";
14 }
15
16 int main(){
17
18     vector<string> book = {"Бушкрафт", "Місний дикий глушині", "Місний розрядженому повітрі", "Приборкати страх"};
19     vector<bool> avail(book.size(), true);
20
21     int choice;
22     do{
23         display_menu();
24         cin >> choice;
25         int number;
26         switch (choice)
27         {
28             case 1:
29                 cout << "\nСписок книг:\n";
30                 for (size_t i = 0; i < book.size(); i++)
31                 {
32                     cout << i+1 << ". " << book[i] << " - " << (avail[i] ? "Доступна" : "Зайнята") << "\n";
33                 }
34
35                 break;
36             case 2:
37                 cout << "\nВведіть номер книги, яку хочете взяти: ";
38                 cin >> number;
39                 if (number >= 1 && number <= book.size() && avail[number - 1])
```

```
40     {
41         avail[number - 1] = false;
42         cout << "Ви успішно взяли книгу: " << book[number - 1] << "\n";
43     }else{
44         cout << "Книга недоступна або невірний номер\n";
45     }
46     break;
47     case 3:
48         cout << "\nВведіть номер книги, яку хочете повернути: ";
49         cin >> number;
50         if (number >= 1 && number <= book.size() && !avail[number - 1])
51         {
52             avail[number - 1] = true;
53             cout << "Ви успішно повернули книгу: " << book[number - 1] << "\n";
54         }else{
55             cout << "Книга вже доступна або невірний номер\n";
56         }
57         break;
58     case 4:
59         cout << "\nВи вийшли з бібліотеки\n";
60         break;
61     default:
62         cout << "\nНеправильний вибір. Будь ласка, спробуйте ще раз\n";
63         break;
64     } while (choice != 4);
65
66     return 0;
67 }
68 }
```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/196/files#diff-4d5adc5f7848d7976d18f5e1fcfbb977a636bbe7e4bedf6e25292f5dd9ea7af8

Завдання 6:

```
1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  int main() {
7      int n;
8      string c;
9      cin >> n;
10     cin >> c;
11
12     int k_s = 0;
13     int v_s = 0;
14     int k_w = 0;
15     int v_w = 0;
16     int score_dif = 0;
17
18     for (int i = 0; i < n; i++)
19     {
20         if (c[i] == 'K')
21         {
22             k_s++;
23             score_dif++;
24         }else{
25             v_s++;
26             score_dif--;
27         }
28         if ((k_s >= 11 || v_s >= 11) && abs(score_dif) >= 2)
29         {if (k_s > v_s)
30             {
31                 k_w++;
32             }else{
33                 v_w++;
34             }
35             k_s = 0;
36             v_s = 0;
37             score_dif = 0;
38         }
39     }
40
41     cout << k_w << ":" << v_w << endl;
42
43     if (k_s != 0 || v_s != 0)
44     {
45         cout << k_s << ":" << v_s;
46     }else if (k_w == 0 && v_w == 0) {
47         cout << "0:0\n";
48     }
49
50     return 0;
```

https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/196/files#diff-34b52c4721e95737ef9d8a51a72fdffb2a83184f5f568ee9e1df6ed6cd8173d2

Результат виконання завдань, тестування та фактично витрачений час:

Завдання 1:

```
PS D:\VS Code project\C ++\epicy> & 'c:\Users\akrav\.vscode\extensions\
MIEngine-In-gkmkuuak.asc' '--stdout=Microsoft-MIEngine-Out-uksd1eck.kfu'
gExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Сума перших 9 членів ряду: 2.24653
PS D:\VS Code project\C ++\epicy>
```

Фактично витрачений час: 25-30 хвилин.

Завдання 2:

```
PS D:\VS Code project\C ++\epicy> & 'c:\Users\akrav\.vscode\extensions\
MIEngine-In-1h04yfco.3ux' '--stdout=Microsoft-MIEngine-Out-2lsdtkof.
gExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
X                SN                SE                Y
-2.000000        -0.705803        -0.693197        -0.693147
-1.810000        -0.504465        -0.504437        -0.504465
-1.620000        -0.325267        -0.325252        -0.325267
-1.430000        -0.169658        -0.169664        -0.169658
-1.240000        -0.056002        -0.056005        -0.056002
-1.050000        -0.002497        -0.002497        -0.002497
-0.860000        -0.019410        -0.019410        -0.019410
-0.670000        -0.103369        -0.103366        -0.103369
-0.480000        -0.239332        -0.239320        -0.239332
-0.290000        -0.408195        -0.408210        -0.408195
-0.100000        -0.593330        -0.593284        -0.593327
PS D:\VS Code project\C ++\epicy> █
```

Фактично витрачений час: 20-25 хвилин.

Завдання 3:

```
MIEngine-In-dbkfzns.el1' '--stdout=Microsoft-MIEngine-Out-w3ibfgia.  
gExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'  
введіть координати 2 точок (x1, y1, x2, y2):  
1 1 4 4  
довжина сторони: 4.24264  
введіть координати 3 точок трикутника (x1, y1, x2, y2, x3, y3):  
5 1 7 5 8 1  
Площа трикутника:6  
Введіть кількість вершин багатокутника:  
5  
Введіть координати вершин по черзі (x, y):  
1 5 2 7 4 7 5 5 3 3  
Площа багатокутника: 10  
PS D:\VS Code project\C ++\epicy>
```

Фактично витрачений час: 55-60 хвилин.

Завдання 4:

```
введіть кількість елементів:  
5  
Введіть 5 чисел:  
2 3 4 5 6  
Результат видалення парних чисел:  
3  
5  
Введіть кількість слів:  
4  
Введіть 4 слів:  
hello  
world  
space  
home  
Результат видалення парних слів:  
hello space  
PS D:\VS Code project\C ++\epicy>
```

Фактично витрачений час: 1 година.

Завдання 5:

Меню бібліотеки:

1. Переглянути всі книги
2. Взяти книгу
3. Повернути книгу
4. Вийти

Оберіть опцію: 1

Список книг:

1. Бушкрафт - Доступна
2. У дикій глушині - Доступна
3. У розрядженому повітрі - Доступна
4. Приборкати страх - Доступна

Меню бібліотеки:

1. Переглянути всі книги
2. Взяти книгу
3. Повернути книгу
4. Вийти

Оберіть опцію: 2

Введіть номер книги, яку хочете взяти: 2

Ви успішно взяли книгу: У дикій глушині

Меню бібліотеки:

1. Переглянути всі книги
2. Взяти книгу
3. Повернути книгу
4. Вийти

Оберіть опцію: 3

Введіть номер книги, яку хочете повернути: 2

Ви успішно повернули книгу: У дикій глушині

Меню бібліотеки:

1. Переглянути всі книги
2. Взяти книгу
3. Повернути книгу
4. Вийти

Оберіть опцію: 4

Ви вийшли з бібліотеки

PS D:\VS Code project\C ++\epicy>

Фактично витрачений час: 2 години.

Завдання 6:

```
PS D:\VS Code project\C ++\epicy> g++ -std=c++11 -I C:\Users\ukr\av  
MIEngine-In-v1hqgvzj.x11' '--stdout=Microsoft-MIEngi  
gExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi  
30  
VVKVKKVVVVKVKKKKKVVKVKKKKVVVV  
1:0  
2:4  
PS D:\VS Code project\C ++\epicy>
```

Фактично витрачений час: 40-45 хвилин.

Висновок: Під час роботи було освоєно цикли, функції, простір імен, рекурсію та інші важливі концепції, що дозволяють писати ефективний і структурований код. Це допомагає оптимізувати програму, покращити її читабельність і створює основу для подальшого вивчення програмування.