

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 ІІІ-12

Сирватка Олександр Ігорович

Львів 2024

Тема роботи:

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

Мета роботи:

Ознайомлення з основами використання циклічних конструкцій, включаючи вкладені цикли та способи завершення їх виконання. Вивчення функцій, простору імен, а також перевантаження функцій. Окрему увагу приділено функціям із змінною кількістю параметрів (еліпсис) та рекурсії. Додатково досліджуються можливості вбудованих функцій і їх роль в оптимізації коду.

Теоретичні відомості:

У даній роботі розглядаються цикли та вкладені цикли, які використовуються для повторюваних дій. Також вивчаються способи завершення виконання циклів за допомогою `break` та `continue`. Досліджується використання функцій, їх перевантаження, функції з змінною кількістю параметрів (еліпсис), а також рекурсія. Окрім цього, аналізуються вбудовані функції та поняття простору імен для організації коду.

Джерела:

- Декілька відео на YouTube:
 - Цикл `while` - <https://www.youtube.com/watch?v=zPi163Xo4m8&t=12s>
 - Цикл `do while` - <https://www.youtube.com/watch?v=AFTfqXdrYBQ>
 - Цикл `for` – 1) <https://www.youtube.com/watch?v=CJm6o82Vu2g>
2) https://www.youtube.com/watch?v=v_ZdslCeRfQ
- Певну інформацію брав на сайтах:
 - <http://cpp.dp.ua/operator-y-tsyklu/>
 - <https://acode.com.ua/urok-72-tsykl-for/>
- Також вивчав як працюють цикли за допомогою ChatGPT.

Виконання роботи:

- **Завдання №1:** Theory Education Activities

Очікувано часу: **3 дні.**

Витрачено часу: **2 дні.**

- **Завдання №2:** Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-7)

Очікувано часу: **2 години.**

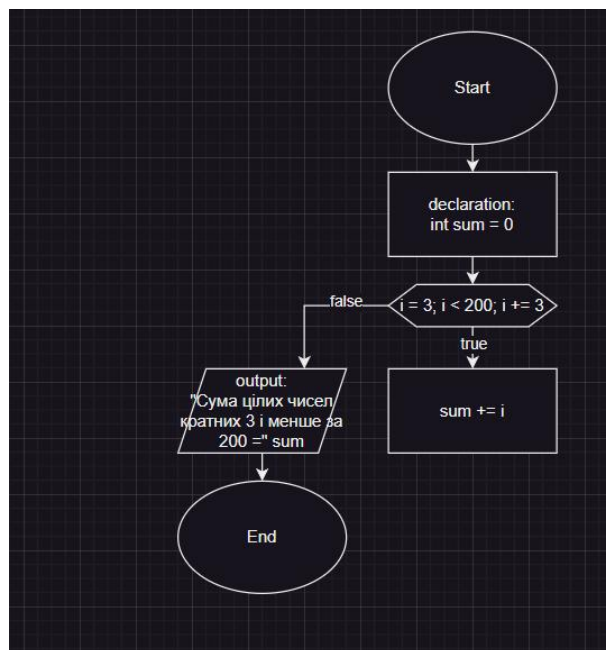
Витрачено часу: **4 години.**

- **Завдання №3:** Lab# programming: VNS Lab 2

Код:

```
Click to add a breakpoint team>
3  using namespace std;
4
5  int main()
6  {
7      int sum = 0;
8      for(int i = 3 ; i < 200; i+= 3){
9          sum += i;
10     }
11     cout << "Сума цілих чисел кратних 3 і менших за 200 = " << sum ;
12
13     return 0;
14 }
15 }
```

Блок-схема:



Очікувано часу: **30 хвилин.**

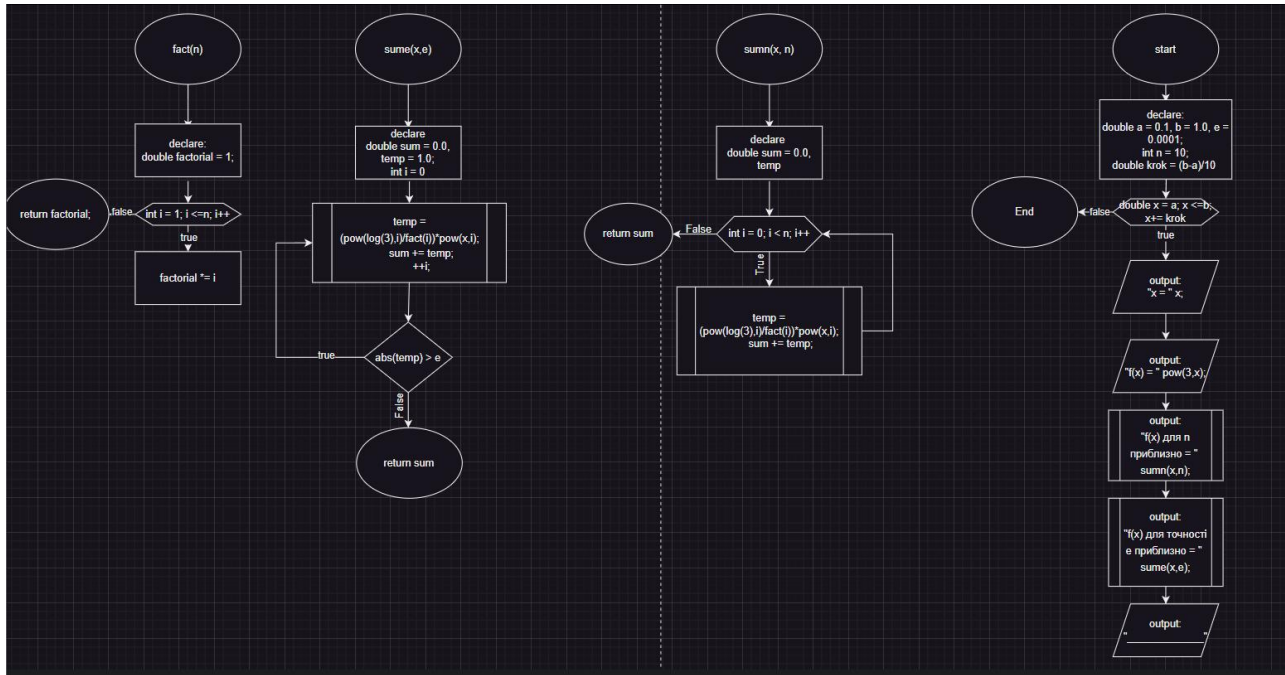
Витрачено часу: **30 хвилин.**

- **Завдання №4:** Lab# programming: VNS Lab 3

Код:

```
1  #include <iostream>
2  #include <math.h>
3
4  using namespace std;
5
6  double fact(int &n){
7      double factorial= 1;
8      for(int i = 1; i<=n; i++){
9          factorial *= i;
10     }
11     return factorial;
12 }
13 double sumn(double &x, int &n){
14     double sum = 0.0;
15     double temp;
16     for(int i = 0; i<n; i++){
17         temp = (pow(log(3),i)/fact(i))*pow(x,i);
18         sum += temp;
19     }
20     return sum;
21 }
22 double sume(double &x, double &e){
23     double temp = 1.0;
24     double sum = 0.0;
25     int i = 0;
26     do {
27         temp = (pow(log(3),i)/fact(i))*pow(x,i);
28         sum += temp;
29         i++;
30     }
31     while (abs(temp) > e );
32
33     return sum;
34 }
35 int main() {
36     double a = 0.1, b = 1.0, e = 0.0001;
37     int n = 10;
38     double krok = (b - a)/10;
39     for (double x = a; x <= b; x+=krok){
40         cout << "x = " << x << "\n";
41         cout << "f(x) = " << pow(3,x) << "\n";
42         cout << "f(x) для n приблизительно = " << sumn(x,n) << "\n";
43         cout << "f(x) для точності e приблизительно = " << sume(x,e) << "\n";
44         cout << "_____ " << "\n";
45     }
46     return 0;
47 }
```

Блок-схема:



Очікувано часу: **30 хвилин.**

Витрачено часу: **30 хвилин.**

- Завдання №5: Lab# programming: VNS Lab 7

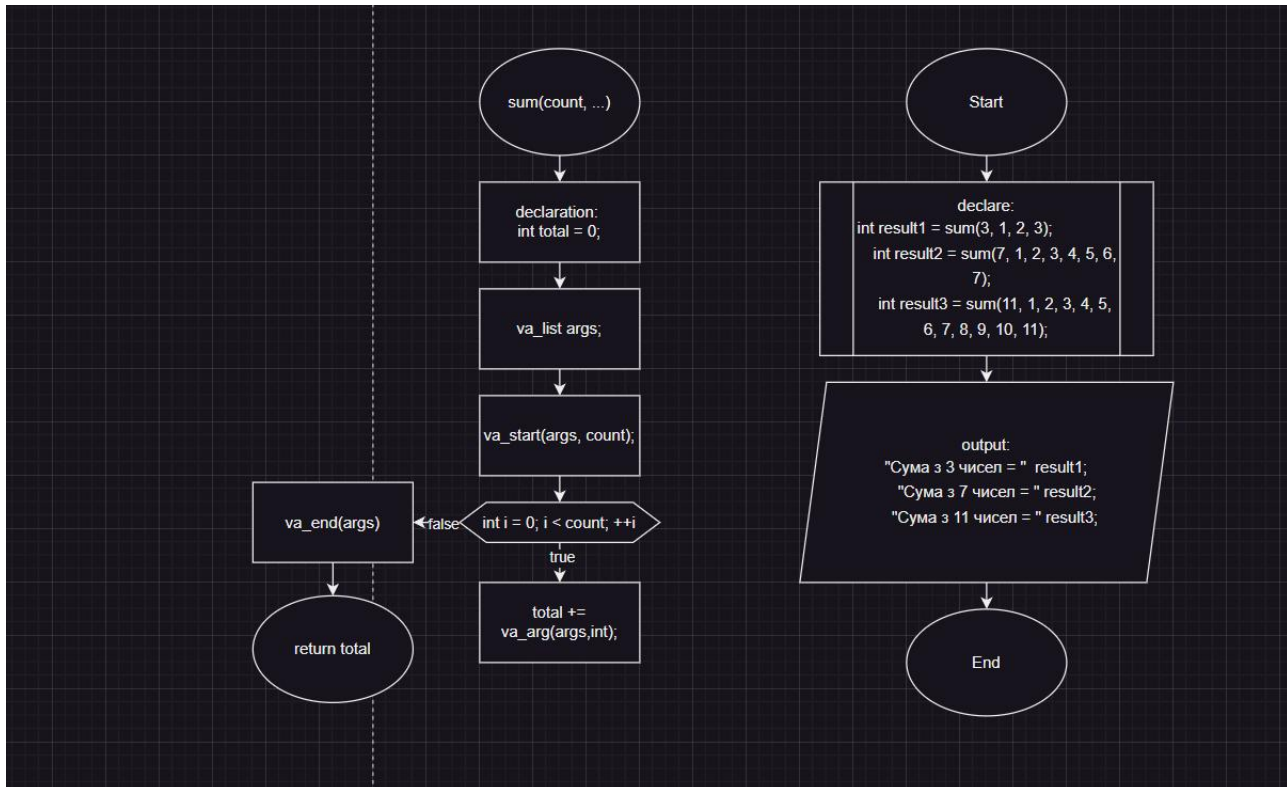
Task 1:

Код:

```

1  #include <iostream>
2  #include <cstdint>
3
4  using namespace std;
5
6
7  int sum(int count, ...) {
8      int total = 0;
9
10     va_list args;
11     va_start(args, count);
12
13
14     for (int i = 0; i < count; ++i) {
15         total += va_arg(args, int);
16     }
17
18     va_end(args);
19
20     return total;
21 }
22
23 int main() {
24
25     int result1 = sum(3, 1, 2, 3);
26     int result2 = sum(7, 1, 2, 3, 4, 5, 6, 7);
27     int result3 = sum(11, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11);
28
29     cout << "Сума з 3 чисел = " << result1 << endl;
30     cout << "Сума з 7 чисел = " << result2 << endl;
31     cout << "Сума з 11 чисел = " << result3 << endl;
32
33     return 0;
34 }
35
  
```

Блок-схема:

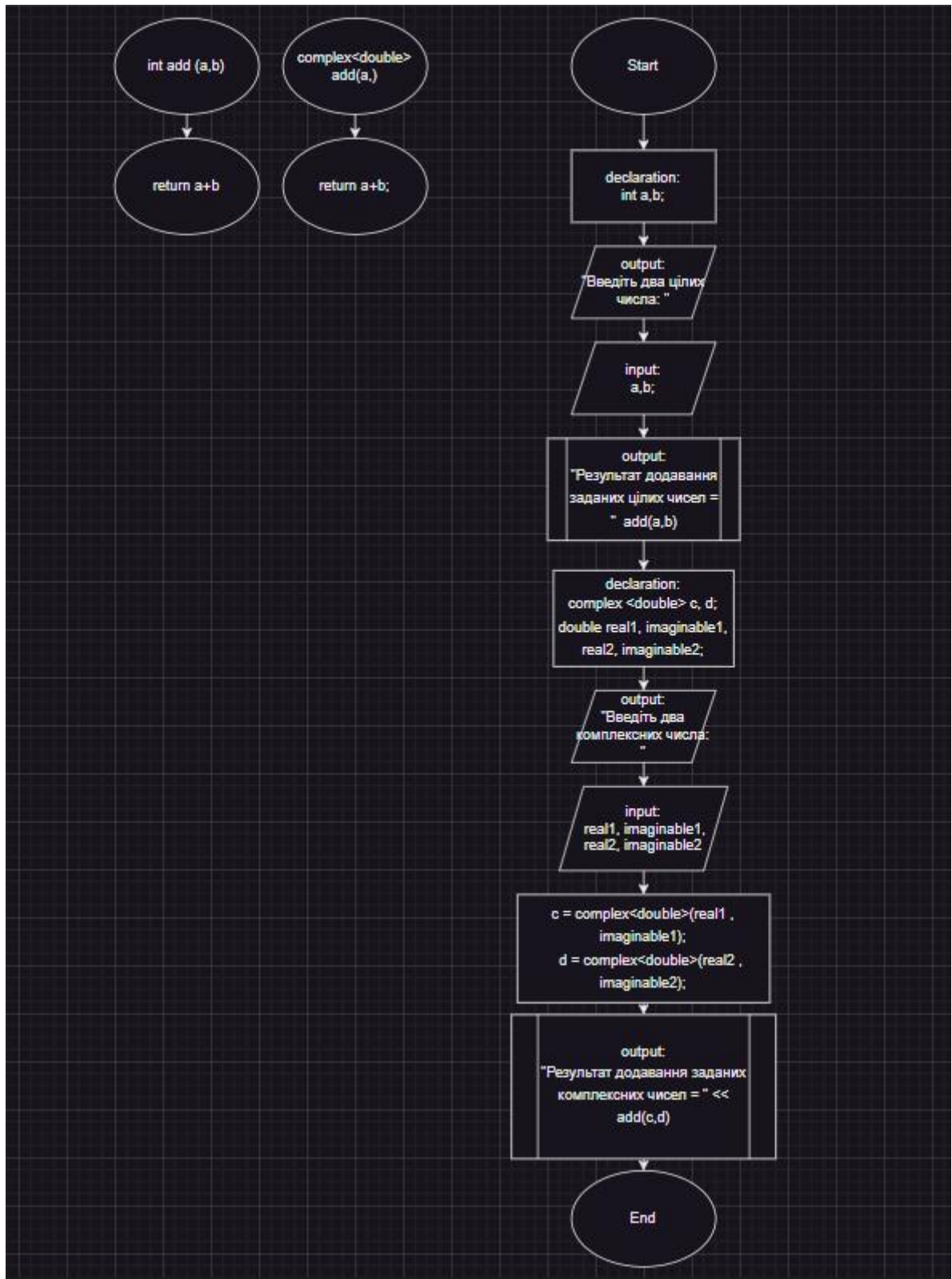


Task 2:

Код:

```
1  #include <iostream>
2  #include <complex>
3
4  using namespace std;
5
6  int add (int a, int b){
7      return a+b;
8  }
9
10 complex <double> add (complex <double> a, complex <double> b){
11     return a+b;
12 }
13
14
15 int main() {
16     int a,b;
17     cout << "Введіть два цілих числа: ";
18     cin >> a >> b;
19     cout << "Результат додавання заданих цілих чисел = " << add(a,b) << "\n";
20
21     complex <double> c, d;
22     double real1, imaginable1, real2, imaginable2;
23     cout << "Введіть два комплексних числа: ";
24     cin >> real1 >> imaginable1 >> real2 >> imaginable2;
25     c = complex<double>(real1 , imaginable1);
26     d = complex<double>(real2 , imaginable2);
27     cout << "Результат додавання заданих комплексних чисел = " << add(c,d);
28
29     return 0;
30 }
```


Блок-схема:



Очікувано часу: 1 година.

Витрачено часу: 2 години.

- Завдання №6: Practice# programming: Class Practice Task
Код:

```

1  #include <iostream>
2  #include <vector>
3  #include <string>
4
5  using namespace std;
6
7  void list(vector<string> &Booklist){
8      cout << "Список книг: ";
9      for (int i = 0; i < Booklist.size(); i++){
10         cout << Booklist[i] << " , ";
11     }
12     cout << "\b\b .\n";
13 }
14
15 void CheckList(vector<string> &Booklist,vector <bool> &availability){
16     int i = 0;
17     for(auto book:Booklist){
18         cout << Booklist[i];
19         if (availability[i]){
20             cout << " є наявності\n";
21         }
22         else{
23             cout << " відсутня\n";
24         }
25         i++;
26     }
27 }
28
29 void Borrow(vector<string> &Booklist,vector <bool> &availability){
30     int booknumber = 0;
31     cout << "Введіть номер книги яку хочете позичити: ";
32     cin >> booknumber;
33     if (availability[(booknumber-1)]){
34         cout << "Ви взяли книгу " << booknumber << "\n";
35         availability[(booknumber-1)] = false;
36     }
37     else{
38         cout << "Книга відсутня\n";
39     }
40 }
41
42 void Return (vector<string> &Booklist,vector <bool> &availability){
43     int booknumber = 0;
44     cout << "Введіть номер книги яку хочете повернути: ";
45     cin >> booknumber;
46     if (!availability[(booknumber-1)]){
47         cout << "Ви повернули книгу " << booknumber << "\n";
48         availability[(booknumber-1)] = true;
49     }
50     else{
51         cout << "Книга уже є бібліотеці\n";
52     }
53 }
54
55 int main(){
56     vector<string> Booklist{"Книга 1", "Книга 2", "Книга 3", "Книга 4", "Книга 5"};
57     vector <bool> availability{true,true,true,true,true};
58     char answer, choice;
59     do {
60         menu:
61         answer == 'n';
62         cout << "Меню:\n1)Список книг \n2)Перевірити наявність книг \n3)Позичити книгу \n4)Повернути книгу\n5)Покинути бібліотеку\n ";
63         cin >> choice;
64         switch(choice){
65             case '1':
66                 list(Booklist);
67                 break;
68             case '2':
69                 CheckList(Booklist,availability);
70                 break;
71             case '3':
72                 Borrow(Booklist,availability);

```

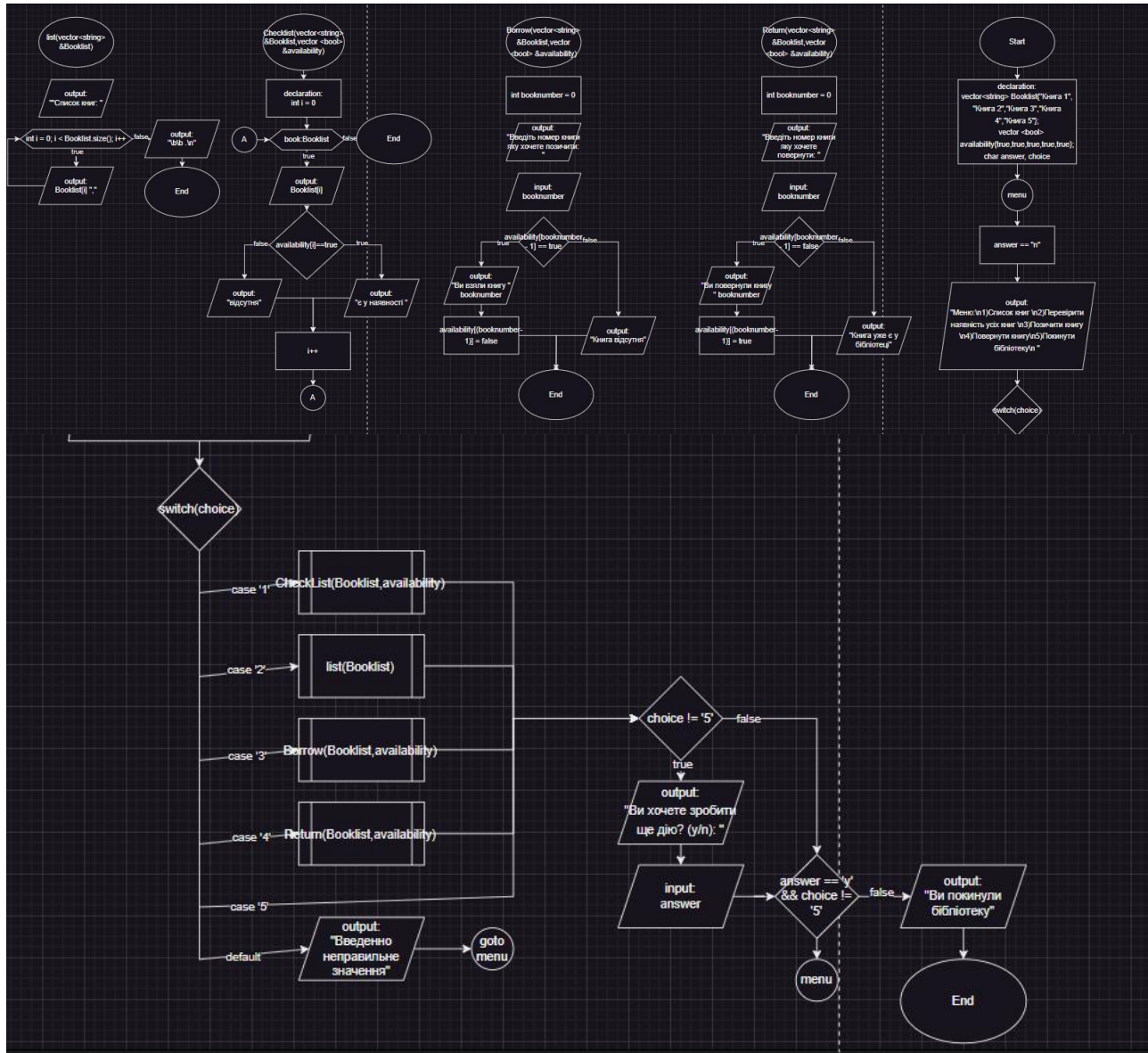


```

67         break;
68     case '2':
69         CheckList(Booklist,availability);
70         break;
71     case '3':
72         Borrow(Booklist,availability);
73         break;
74     case '4':
75         Return(Booklist,availability);
76         break;
77     case '5':
78         break;
79     default :
80         cout << "Введено невірні значення \n";
81         goto menu;
82         break;
83     }
84     if (choice != '5') {
85         cout << "Ви хочете зробити ще дію? (y/n): ";
86         cin >> answer;
87     }
88
89     }while (answer == 'y' && choice !='5');
90     cout << "Ви покинули бібліотеку";
91
92     return 0;
93 }

```

Блок-схема:



Очікувано часу: **2 години.**

Витрачено часу: **3 години.**

- **Завдання №7:** Practice# programming: Self Practice Task

Код:

```
#include <iostream>
using namespace std;

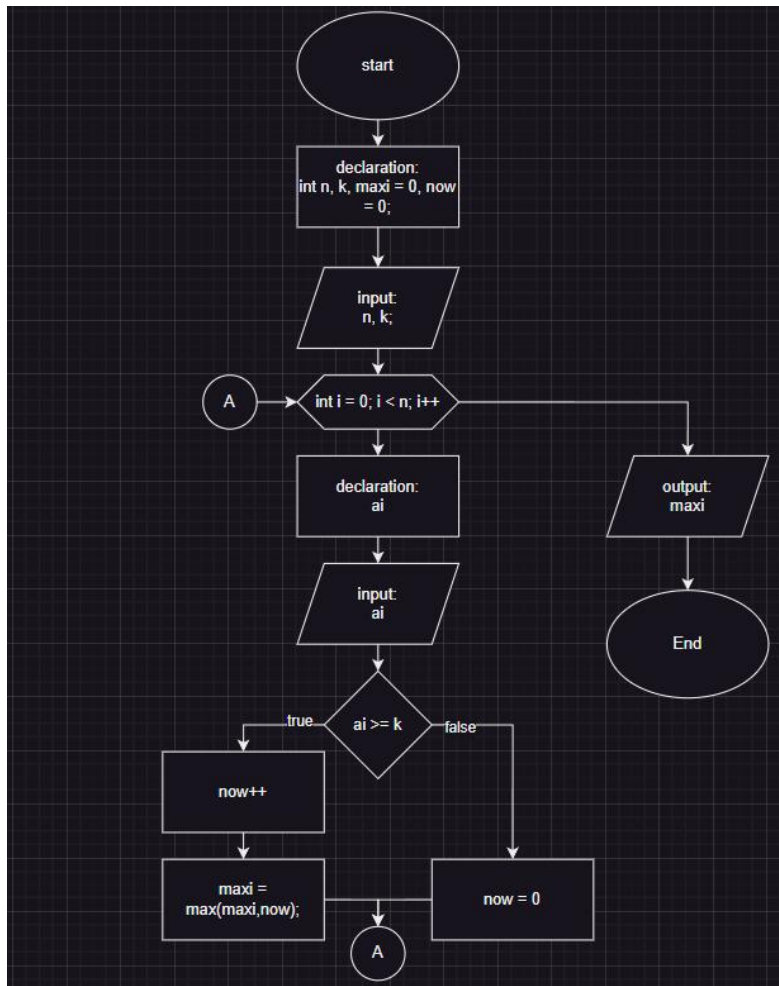
int main() {
    int n, k, maxi = 0, now = 0;
    cin >> n >> k;

    for (int i = 0; i < n; i++) {
        int ai;
        cin >> ai;
        if (ai >= k) {
            now++;
            maxi = max(maxi, now);
        } else {
            now = 0;
        }
    }

    cout << maxi << endl;

    return 0;
}
```

Блок - схема:



Задача на Алготестері:

<https://algotester.com/uk/ArchiveProblem/DisplayWithEditor/20075>

Очікувано часу: **20 хвилин.**

Витрачено часу: **20 хвилин.**

Pull-Request: https://github.com/artificial-intelligence-department/ai_programming_playground_2024/pull/263

Висновок: Під час роботи я ознайомився з основами циклів, таких як `for()`, `while()`, `do { } while()`, та оператором переходу `goto`. Я розібрався, як працюють ці конструкції, включаючи вкладені цикли та способи їх завершення. Також вивчив функції, простір імен, та перевантаження функцій для різних завдань. Окремо зосередився на функціях зі змінною кількістю параметрів (еліпсис) та рекурсії. Дослідив, як вбудовані функції можуть покращити ефективність коду.