

Лабораторна робота №7. Одновимірні масиви

Міністерство освіти і науки України
Національний технічний університет України «Київський
політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації
і управління

Звіт

з лабораторної роботи № 7 з дисципліни
«Основи програмування»

«Одновимірні масиви»

Варіант __4__

Виконав студент _____ Берлінський Ярослав Владленович _____
(шифр, прізвище, ім'я, по батькові)

Перевірив _____
(прізвище, ім'я, по батькові)

Київ 2020

Назва роботи: робота з одновимірними масивами.

Варіант: 4

Умова задачі:

4. Заданий масив дійсних чисел $M(n)$, серед яких є і від'ємні. Знайти суму його елементів, розміщених між максимальним та мінімальним елементами масива (в суму включити і самі ці елементи). Всі від'ємні елементи масиву збільшити на величину отриманої суми.

Постановка задачі. Генерується масив дійсних чисел з наперед заданою кількістю елементів a , інтервалом значень ***begin – end*** та числом розрядів після коми ***eps***.

Послідовно відбувається пошук максимального та мінімального елемента масиву, обчислюється сума елементів між максимальним(мінімальним) та мінімальним(максимальним). До від'ємних елементів масиву додається значення отриманої суми.

Під кінець виводиться змінений масив.

Вигідним способом буде організація вирішення поставленої задачі за допомогою підпрограм: генерація масиву, пошук мін. і макс. елементів, обчислення суми, змінення від'ємних елементів масиву і наступний вивід.

Розв'язок:

1. C++:

```

1
2 //
3 // main.cpp
4 // LAB3
5 //
6 // Created by Берлинский Ярослав Владленович on 11.11.2020.
7
8 #include <iostream>
9 #include <iomanip>
10 #include <cmath>
11 using namespace std;
12
13 void input_m(float*, int, int, int, int);
14 void max_n_min(float[], int);
15 float array_sum(float[], float);
16 void array_change(float*, int, int, float);
17 void array_output(float*, int, int);
18 int num1=0, num2=0;
19 int main(){
20     float sum=0;
21     int begin, end;
22     int eps;
23     int a;
24     do{
25         cout<<"Розмір: ";
26         cin>>a;
27     }
28     while (a<=0);
29     float m[a];
30     do{
31         cout<<"Інтервал цілих значень чисел масиву: ";
32         cin>>begin;
33         cin>>end;
34     }
35     while (begin>end);
36     do{
37         cout<<"Розмір дробної частини дійсних чисел масиву: ";
38         cin>>eps;
39     }
40     while (eps<0);
41     cout<<endl;
42     input_m(m, a, begin, end, eps);
43     array_output(m, a, eps);
44     max_n_min(m, a);
45     array_sum(m, sum);
46     sum=array_sum(m, sum);
47     cout<<"Значення суми між ["<<num1<<"]\nта ["<<num2<<"] елементами "<<"дорівнює:

```

```

LAB5 / LAB5 / main.cpp / max_n_min(array, a)
47     cout<<"Значення суми між ["<<num1<<"]\nta ["<<num2<<"] елементами "<<"дорівнює:
      "<<fixed<<setprecision(eps)<<sum<<endl<<endl;
48     array_change(m, a, eps, sum);
49     array_output(m, a, eps);
50 }
51 void max_n_min(float array[], int a){
52     float max=array[0];
53     float min=array[0];
54     int numb_max=0;
55     int numb_min=0;
56
57     for(int j=0; j<a; j++){
58         if(array[j]>max){
59             max=array[j];
60             numb_max=j;
61         }
62     }
63
64     for(int j=0; j<a; j++){
65         if(array[j]<min){
66             min=array[j];
67             numb_min=j;
68         }
69     }
70
71     numb_max>numb_min? (num1=numb_min):(num1=numb_max);
72     numb_max>numb_min? (num2=numb_max):(num2=numb_min);
73     cout<<"\nМаксимум = "<<max<<". Порядковий номер: "<<numb_max<<endl;
74     cout<<"Мінімум = "<<min<<". Порядковий номер: "<<numb_min<<endl<<endl;
75 }
76 void input_m(float *p, int a, int begin, int end, int eps){
77     srand(time(NULL));
78     for (int i=0; i<a; i++){
79         *p=begin + rand()%(end-begin) + (float)( rand()%((int)(pow(10,eps)))/(pow(10,eps)));
80         p++;
81     }
82 }
83 float array_sum(float array[], float sum){
84     for(int i=num1; i<=num2; i++){
85         sum+=array[i];
86     }
87
88     return sum;
89 }
90 void array_change(float *p, int a, int eps, float sum){
91     for(int i=0; i<a; i++){
92         if(*p<0){

```

```

62     }
63
64     for(int j=0; j<a; j++){
65         if(array[j]<min){
66             min=array[j];
67             numb_min=j;
68         }
69     }
70
71     numb_max>numb_min? (num1=numb_min):(num1=numb_max);
72     numb_max>numb_min? (num2=numb_max):(num2=numb_min);
73     cout<<"\nМаксимум = "<<max<<". Порядковый номер: "<<numb_max<<endl;
74     cout<<"Минимум = "<<min<<". Порядковый номер: "<<numb_min<<endl<<endl;
75 }
76 void input_m(float *p, int a, int begin, int end, int eps){
77     srand(time(NULL));
78     for (int i=0; i<a; i++){
79         *p=begin + rand()%(end-begin) + (float)( rand()%((int)(pow(10,eps)))/(pow(10,eps)));
80         p++;
81     }
82 }
83 float array_sum(float array[], float sum){
84     for(int i=num1; i<=num2; i++){
85         sum+=array[i];
86     }
87
88     return sum;
89 }
90 void array_change(float *p, int a, int eps, float sum){
91     for(int i=0; i<a; i++){
92         if(*p<0){
93             *p=*p+sum;
94         }
95         p++;
96     }
97 }
98
99 void array_output(float *p, int a, int eps){
100     cout<<"Отриманий масив: \n";
101     for(int i=0; i<a; i++){
102         cout<<i<<"-й елемент масиву: "<<fixed<<setprecision(eps)<<*p<<endl;
103         p++;
104     }
105 }
106
107
108

```

Тестування вхідних даних(C++):

1) a=10, begin=-50, end=50, eps=2:

```
Інтервал цілих значень чисел масиву: -50
50
Розмір дробної частини дійсних чисел масиву: 2

0-й) елемент масиву 24.71
1-й) елемент масиву -4.22
2-й) елемент масиву 34.84
3-й) елемент масиву -29.54
4-й) елемент масиву -11.36
5-й) елемент масиву -37.36
6-й) елемент масиву 37.09
7-й) елемент масиву -6.33
8-й) елемент масиву -35.71
9-й) елемент масиву -27.11

Максимум = 37.09. Порядковий номер: 6
Мінімум = -37.36. Порядковий номер: 5

[5] -37.36
[6] 37.09

Сума = -0.27

Змінений масив:
0)-й елемент: 24.71
1)-й елемент: -4.49
2)-й елемент: 34.84
3)-й елемент: -29.81
4)-й елемент: -11.63
5)-й елемент: -37.63
6)-й елемент: 37.09
7)-й елемент: -6.60
8)-й елемент: -35.98
9)-й елемент: -27.38
Program ended with exit code: 0
```

2) a=10, begin=-1000, end=1000, eps=2:

```
Інтервал цілих значень чисел масиву: -1000
1000
Розмір дробної частини дійсних чисел масиву: 2
```

```
0-й) елемент масиву -85.43
1-й) елемент масиву -599.85
2-й) елемент масиву 667.36
3-й) елемент масиву 53.95
4-й) елемент масиву 187.51
5-й) елемент масиву -316.06
6-й) елемент масиву 828.02
7-й) елемент масиву 336.10
8-й) елемент масиву 243.09
9-й) елемент масиву 546.82
```

```
Максимум = 828.02. Порядковий номер: 6
Мінімум = -599.85. Порядковий номер: 1
```

```
[1] -599.85
[2] 667.36
[3] 53.95
[4] 187.51
[5] -316.06
[6] 828.02
```

```
Сума = 820.93
```

```
Змінений масив:
0)-й елемент: 735.50
1)-й елемент: 221.08
2)-й елемент: 667.36
3)-й елемент: 53.95
4)-й елемент: 187.51
5)-й елемент: 504.87
6)-й елемент: 828.02
7)-й елемент: 336.10
8)-й елемент: 243.09
9)-й елемент: 546.82
```

All Output ↕

Filter



3) a=10, begin=0, end=1, eps=6:

```
Інтервал цілих значень чисел масиву: 0
1
Розмір дробної частини дійсних чисел масиву: 6

0-й) елемент масиву 0.764245
1-й) елемент масиву 0.513463
2-й) елемент масиву 0.805713
3-й) елемент масиву 0.672528
4-й) елемент масиву 0.059547
5-й) елемент масиву 0.378442
6-й) елемент масиву 0.625337
7-й) елемент масиву 0.411556
8-й) елемент масиву 0.472718
9-й) елемент масиву 0.425387

Максимум = 0.805713. Порядковий номер: 2
Мінімум = 0.059547. Порядковий номер: 4

[2] 0.805713
[3] 0.672528
[4] 0.059547

Сума = 1.537788

Змінений масив:
0)-й елемент: 0.764245
1)-й елемент: 0.513463
2)-й елемент: 0.805713
3)-й елемент: 0.672528
4)-й елемент: 0.059547
5)-й елемент: 0.378442
6)-й елемент: 0.625337
7)-й елемент: 0.411556
8)-й елемент: 0.472718
9)-й елемент: 0.425387
Program ended with exit code: 0
```

All Output ↕

Filter

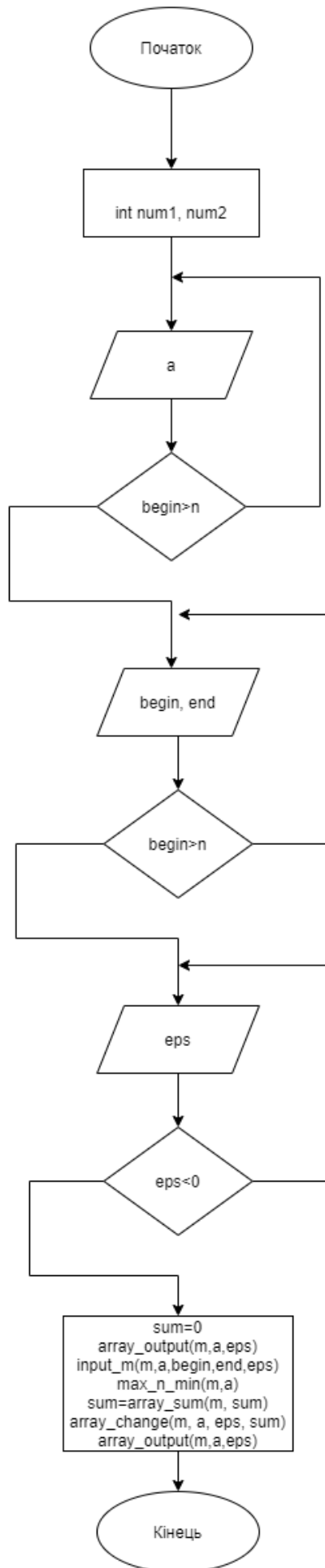


Нескладно перевірити вірність результатів. Інші дані можна перевірити самостійно, перейшовши безпосередньо до коду програми:

[GitHub](#)

Блок-схема роботи алгоритму:

Програма



Підпрограма генерації масиву

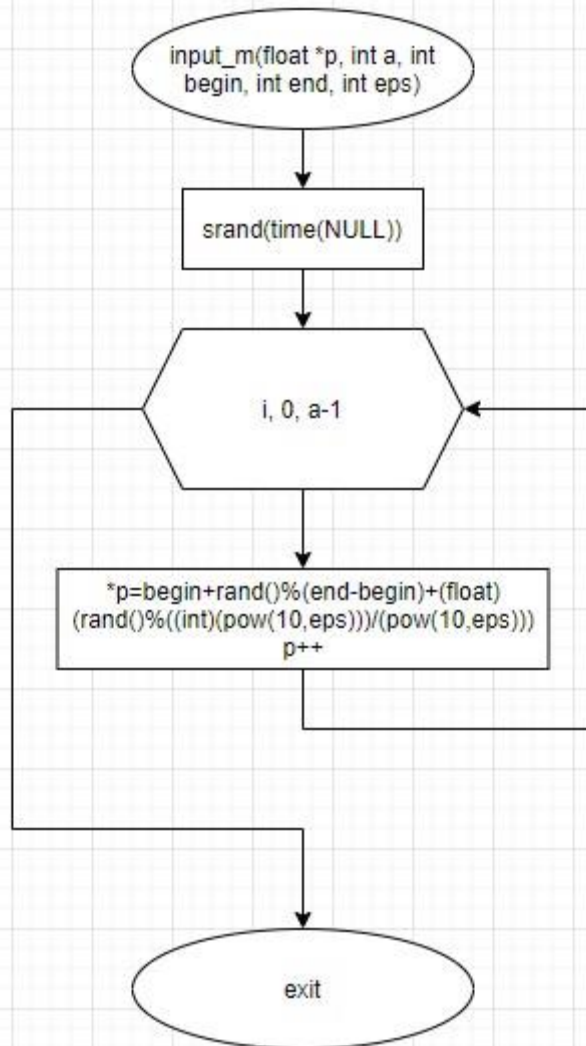
input_m(float *p, int a, int
begin, int end, int eps)

srand(time(NULL))

i, 0, a-1

*p=begin+rand()%(end-begin)+(float)
(rand()%((int)(pow(10,eps)))/(pow(10,eps)))
p++

exit



Підпрограма знаходження
макс. і мін. та їх номерів

max_n_min(float array[], int a)

max=array[0]
min=array[0]

j, 0, a-1

array[j]>max

max=array[j]
numb_max=j

j, 0, a-1

array[j]<min

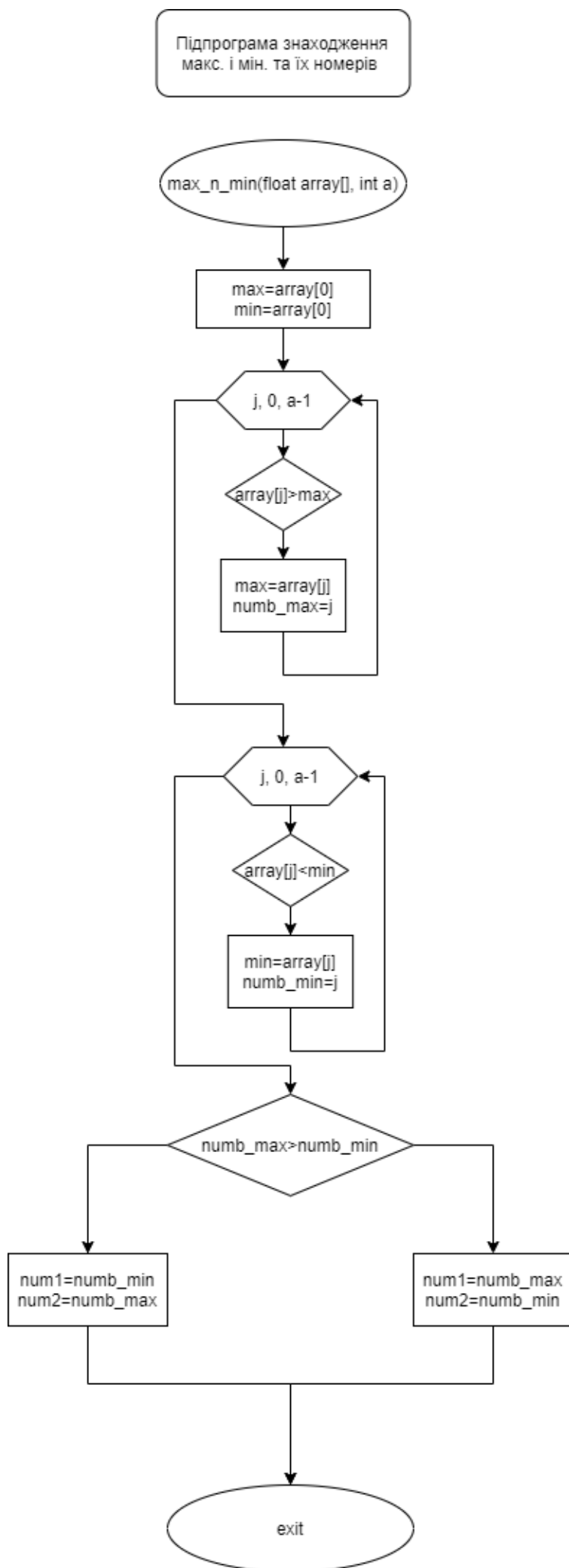
min=array[j]
numb_min=j

numb_max>numb_min

num1=numb_min
num2=numb_max

num1=numb_max
num2=numb_min

exit



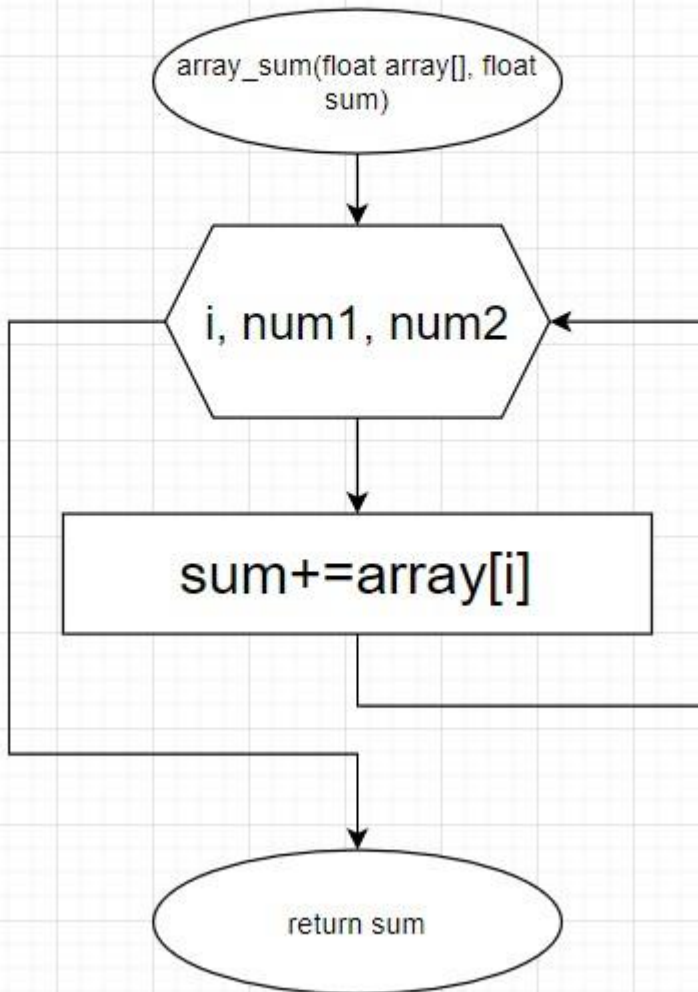
Підпрограма суми елементів
масиву на інтервалі

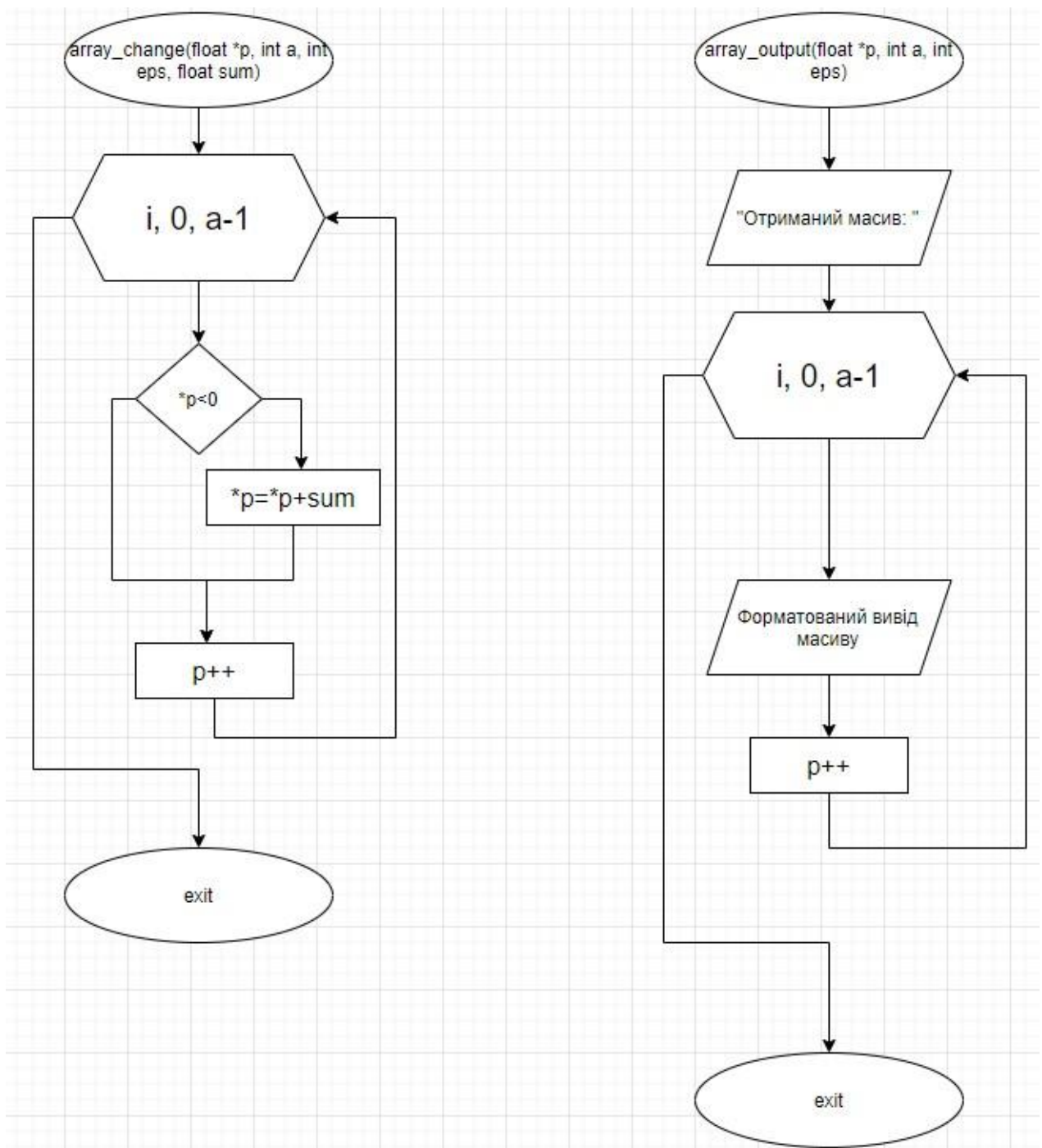
array_sum(float array[], float
sum)

i, num1, num2

sum+=array[i]

return sum





Висновок: отже, програма роботи з одновимірним масивом(генерації/пошук максимумів і мінімумів/заміни елементів/вибіркові суми/вивід) була реалізована на компільованій мові С++ з урахуванням можливого неправильного з боку користувача вводу, а також можливістю змінювати розмір, діапазон значень та розрядність елементів.

Сам алгоритм був реалізований за допомогою використання підпрограм(перелічені вище) основних етапів розв'язання задачі, щоб спростити сприйняття коду та зробити процес подальшої модифікації програми менш кропітким.