

## Лабораторна робота № 11

### Одновимірні масиви

**Мета:** дослідити функції генерування послідовностей випадкових чисел, ознайомитись з можливостями створення одновимірних масивів і навчитись обробляти елементи масивів:

**Хід роботи:** Рішення всіх завдань супроводжується створенням блок-схеми.

**Завдання 1 .** Написати програму, що реалізує десять генераторів псевдовипадкових чисел. Кожний генератор викликати за допомогою меню, яке реагує на введення цілого числа: 1, ... , 10:

1	[-4; -1)	6	[-7,85; $28\sqrt{3}$ ]
2	[100; 299]	7	[-100; 100]
3	[-35; -1] парні	8	[23; 71]
4	[-128; 127]	9	[0; 2)
5	[-7; 13) непарні	10	$[\sqrt{17}; \sqrt{82})$

Листинг програми:

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>
int main() {
    srand(time(0));
    printf("Enter a number from 1 to 10: ");int number;scanf_s("%d", &number);
    double randomnumber2;
    int randomnumber;
    switch (number) {
        case 1:
            randomnumber = rand() % 3 - 4;
            break;
        case 2:
            randomnumber = rand() % 199 + 100;
            break;
        case 3:do {randomnumber = rand() % 37 - 35;
        }while (randomnumber % 2 != 0);
            break;
        case 4:
            randomnumber = rand() % 256 - 128;
            break;
```

					ДУ«Житомирська політехніка».23.121.10.000 – Лр11			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.	Семенчук О.А.				Звіт з лабораторної роботи		Літ.	Арк.
Перевір.	Чижмотря О. В							1
Керівник								9
Н. контр.							ФІКТ Гр. ІПЗ-121-1[2]	
Зав. каф.								

```

case 5:do {
    randomnumber = rand() % 20 - 7 + 1;
} while (randomnumber % 2 == 0);
break;
case 6:
    randomnumber2 = ((double)rand())/((double)RAND_MAX/(7.85 + 28 * sqrt(3)));
    break;
case 7:
    randomnumber = rand() % 200 - 100;
    break;
case 8:
    randomnumber = rand() % (71 - 23) + 23;
    break;

case 9:
    randomnumber = rand() % 2;
    break;
case 10:
    randomnumber2 = ((double)rand())/((double)RAND_MAX/(sqrt(82) - sqrt(17) +
sqrt(17)));
    break;
default:
    printf("Invalid user input!");
    return 0;}
if(number != 6 && number != 10)printf("The random number is : %d", randomnumber);
else printf("The random number is : %lf",randomnumber2));
return 0;}

```

Результат виконання програми:

```

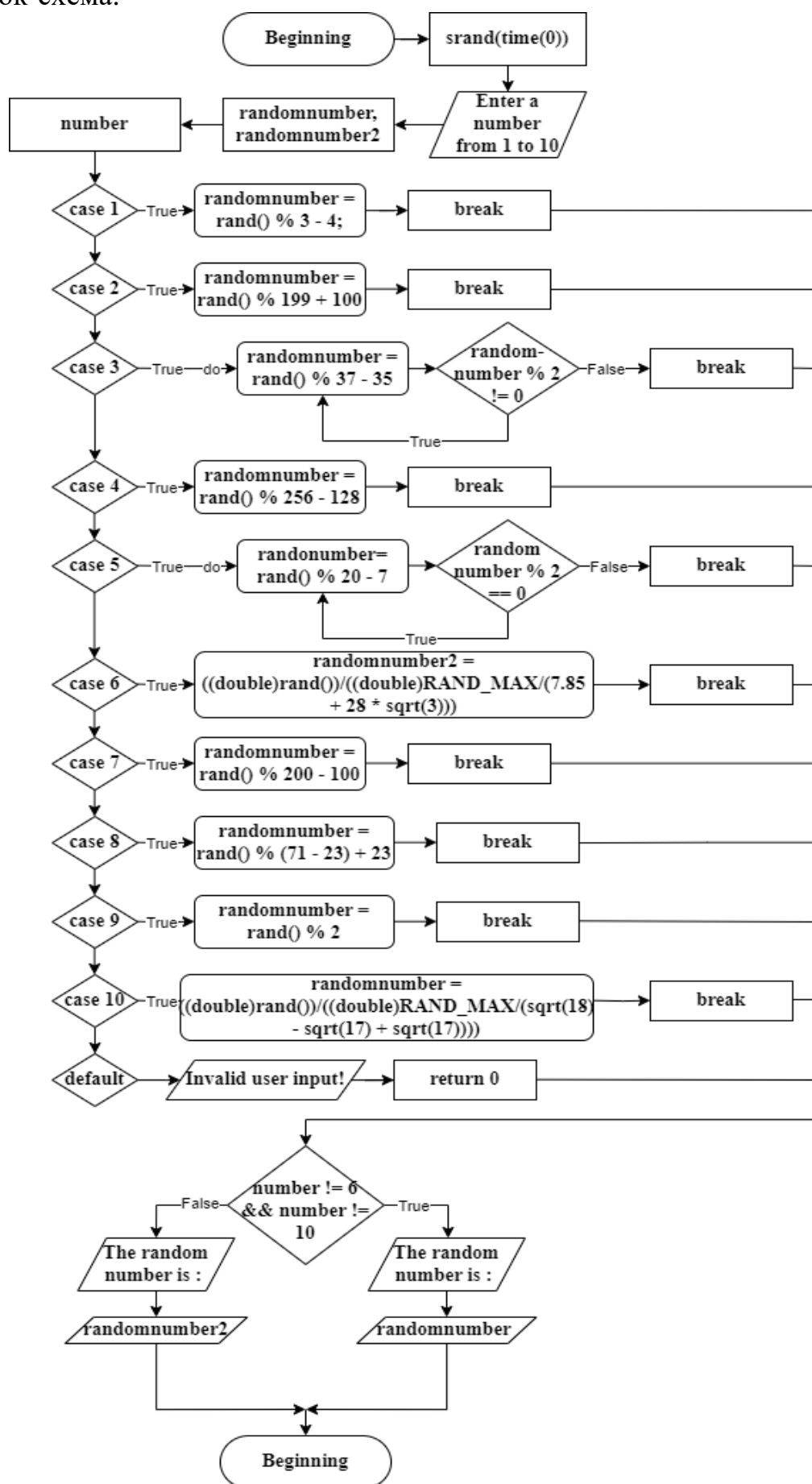
C:\Users\FearlessAtom\Desktop>task1.exe
Enter a number from 1 to 10: 1
The random number is : -2
C:\Users\FearlessAtom\Desktop>task1.exe
Enter a number from 1 to 10: 2
The random number is : 312
C:\Users\FearlessAtom\Desktop>task1.exe
Enter a number from 1 to 10: 3
The random number is : -14
C:\Users\FearlessAtom\Desktop>task1.exe
Enter a number from 1 to 10: 4
The random number is : -118
C:\Users\FearlessAtom\Desktop>task1.exe
Enter a number from 1 to 10: 5
The random number is : 11
C:\Users\FearlessAtom\Desktop>task1.exe
Enter a number from 1 to 10: 6
The random number is : 52.859994
C:\Users\FearlessAtom\Desktop>task1.exe
Enter a number from 1 to 10: 7
The random number is : 66
C:\Users\FearlessAtom\Desktop>task1.exe
Enter a number from 1 to 10: 8
The random number is : 30
C:\Users\FearlessAtom\Desktop>task1.exe
Enter a number from 1 to 10: 9
The random number is : 1
C:\Users\FearlessAtom\Desktop>task1.exe
Enter a number from 1 to 10: 10
The random number is : 8.514831

```

Рис. 1

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр11	Арк.
		Чижмоторя О. В.				2
Змн.	Арк.	№ докум.	Підпис	Дата		

# Блок-схема:



**Завдання 2.** Числа  $m$ ,  $n$  і  $k$  ( $3 \leq k \leq 10$ ) вводяться з клавіатури. Згенерувати і вивести на екран  $m$  цілих випадкових чисел з проміжку, вказаному у пункті а, та  $n$  дійсних чисел (виводити на екран з вказаною точністю) з проміжку, вказаному у пункті б. Виведення на екран здійснювати по  $k$  чисел у рядку:

10	[-10, 10]	[-1, 0] з точністю до сотих
----	-----------	-----------------------------

Листинг програми:

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int main(){
    char something[100] = {"-----\n"};
    srand(time(0));
    printf("What is the value of m : ");int m;scanf_s("%d",&m);
    printf("What is the value of n : ");int n;scanf_s("%d",&n);
    printf("What is the value of k [3;10] : ");int k;scanf_s("%d",&k);
    if(k < 3 || k > 10){printf("k can be only in the range of 3 and 10");return 1;}
    int randomnumber;
    printf("%s",something);
    for(int index = 0;index < m;index++){
        randomnumber = rand() % 20 - 10;
        printf("%3d",randomnumber);
        if((index + 1) % k == 0){printf("\n");}}
    double doublerandomnumber;
    printf("%s",something);
    for(int index = 0;index < n;index++){
        doublerandomnumber = ((double)rand())/(RAND_MAX);
        printf("%.2lf",doublerandomnumber - 1);
        if((index + 1) % k == 0){printf("\n");}}
    printf("%s",something);
    return 0;}
```

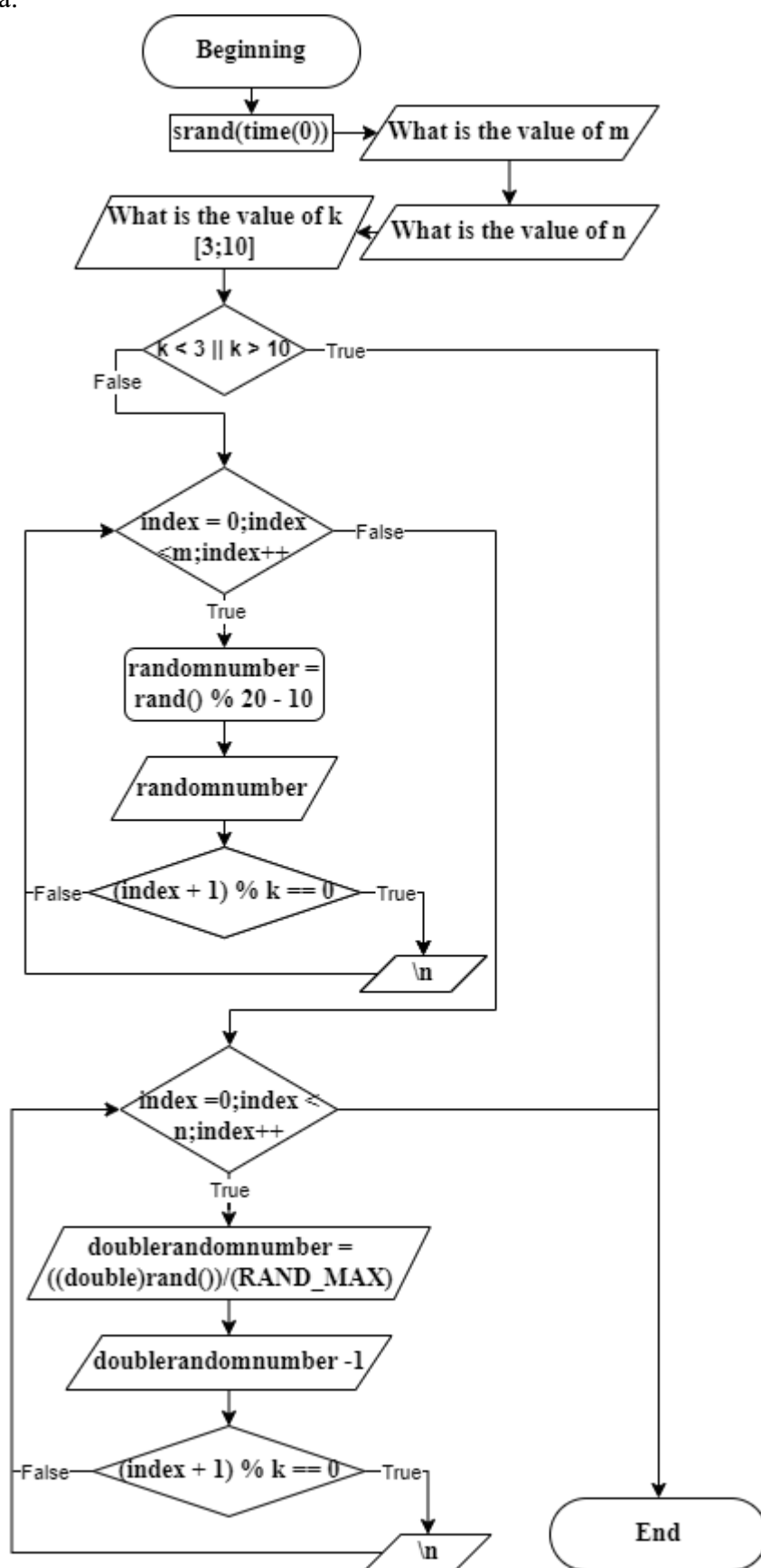
Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>task2.exe
What is the value of m : 9
What is the value of n : 9
What is the value of k [3;10] : 3

-----
-5 -5 3
 5 -5 4
 5 -8 3
-----
-0.12-0.15-0.31
-0.23-0.30-0.96
-0.82-0.97-0.12
-----
```

Рис. 2

Блок-схема:



		Семенчук О. А.		
		Чижмотря О. В.		
Змн.	Арк.	№ докум.	Підпис	Дата

### Завдання 3. Розробити програму, дотримуючись таких вимог:

- використовувати статичні масиви;
- число  $n$  (кількість елементів масиву);
- елементи масиву – випадкові числа, згенеровані на інтервалі  $[a, b]$ , де  $a$  і  $b$  вводяться з клавіатури ( $a < b$ );
- усі вхідні дані а також елементи масиву виводяться на екран:

10	В одновимірному масиві, що складається з $n$ цілих елементів, обчислити: 1) мінімальний за модулем елемент масиву; 2) суму модулів елементів, розташованих після першого елемента, рівного нулю.
----	--

#### Листинг програми:

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include <time.h>
int main() {
    srand(time(0));
    printf("Enter the value for a: ");int a;scanf_s("%d", &a);
    printf("Enter the value for b: ");int b;scanf_s("%d", &b);
    if (a >= b) {printf("a must be less than b!\n");return 1;}
    int array[10];
    int min = 9999999, sum = 0, foundzero = 0;
    for (int index = 0; index < 10; index++) {
        array[index] = rand() % (b - a + 1) + a;
        if (fabs(array[index]) < min) {
            min = abs(array[index]);}
        if (array[index] == 0) {
            foundzero = 1;}
        if (foundzero) {
            sum = sum + fabs(array[index]);}
        printf("%d ", array[index]);}
    printf("\nMin is: %d\n", min);
    if (sum) {printf("Sum is: %d\n", sum);}
    else {printf("No zeroes found!\n");}
    return 0;}
```

#### Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>task3.exe
Enter the value for a: -10
Enter the value for b: 10
4 2 -2 -10 -7 -5 1 -10 -7 4
Min is: 1
No zeroes found!
```

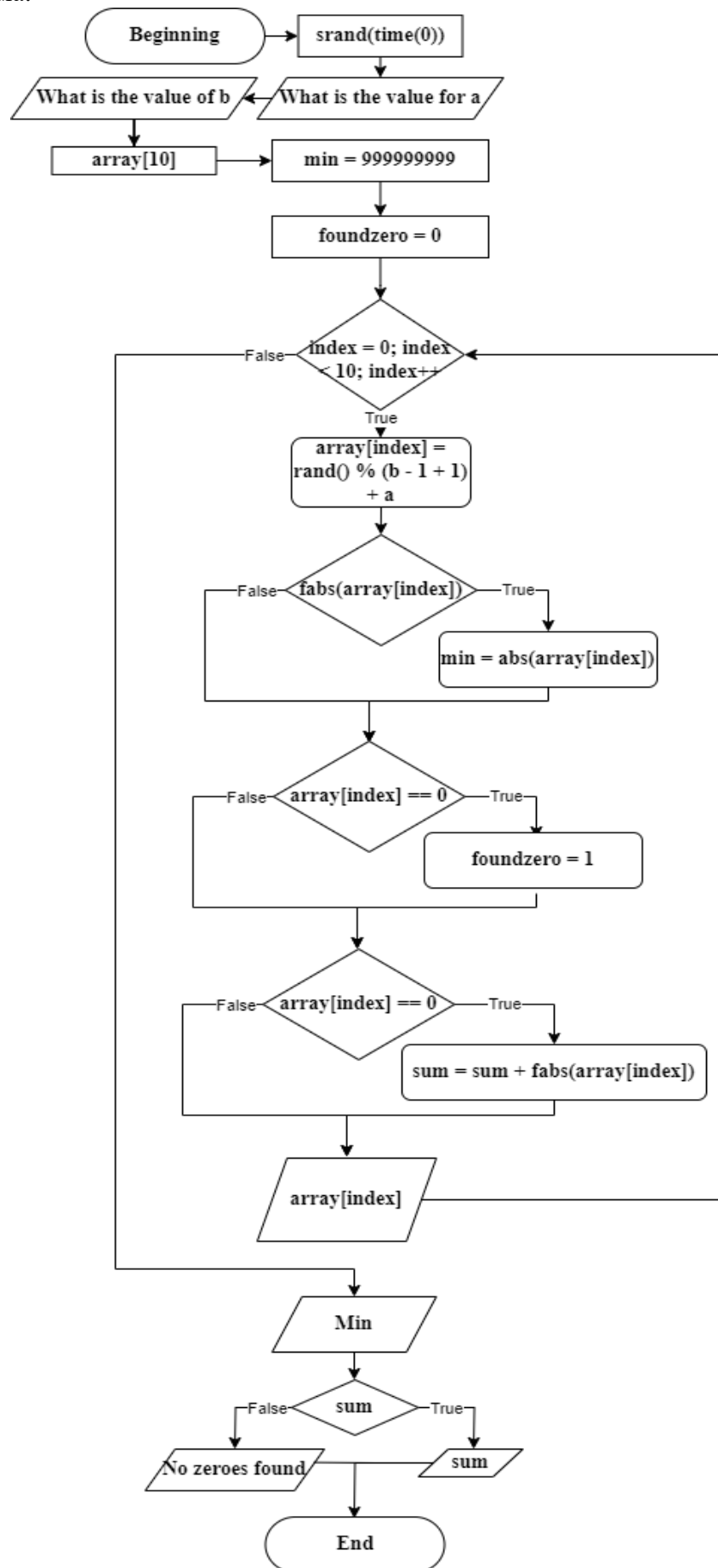
Рис. 3

```
C:\Users\FearlessAtom\Desktop>task3.exe
Enter the value for a: -10
Enter the value for b: 10
5 9 0 3 1 -7 10 -5 -9 2
Min is: 0
Sum is: 37
```

Рис. 4

		Семенчук О. А.			ДУ «Житомирська політехніка».23.121.10.000 – Лр11	Арк.
		Чижмоторя О. В.				6
Змн.	Арк.	№ докум.	Підпис	Дата		

Блок-схема:



#### Завдання 4. Написати програму:

**10**

Задано масив L[11]. Обчислити та вивести добуток індексів парних елементів масиву та поміняти мінімальний елемент масиву на добуток індексів. Вивести результуючий масив на екран.

Листинг програми:

```
#include <stdio.h>
int main() {
    int array[11] = {9, 6, 4, 5, 2, 1, 4, 5, 3, 7, 2};
    int sum = 1, atom, min = 999999;
    for (int index = 0; index < 11; index++){
        if (array[index] % 2 == 0) {sum = sum * index;}
        if (array[index] < min){min = array[index];
            atom = index;}}
    array[atom] = sum;
    for (int index = 0; index < 11; index++) {printf("%d ", array[index]);}
    printf("\nSum is: %d\n", sum);
    return 0;}
```

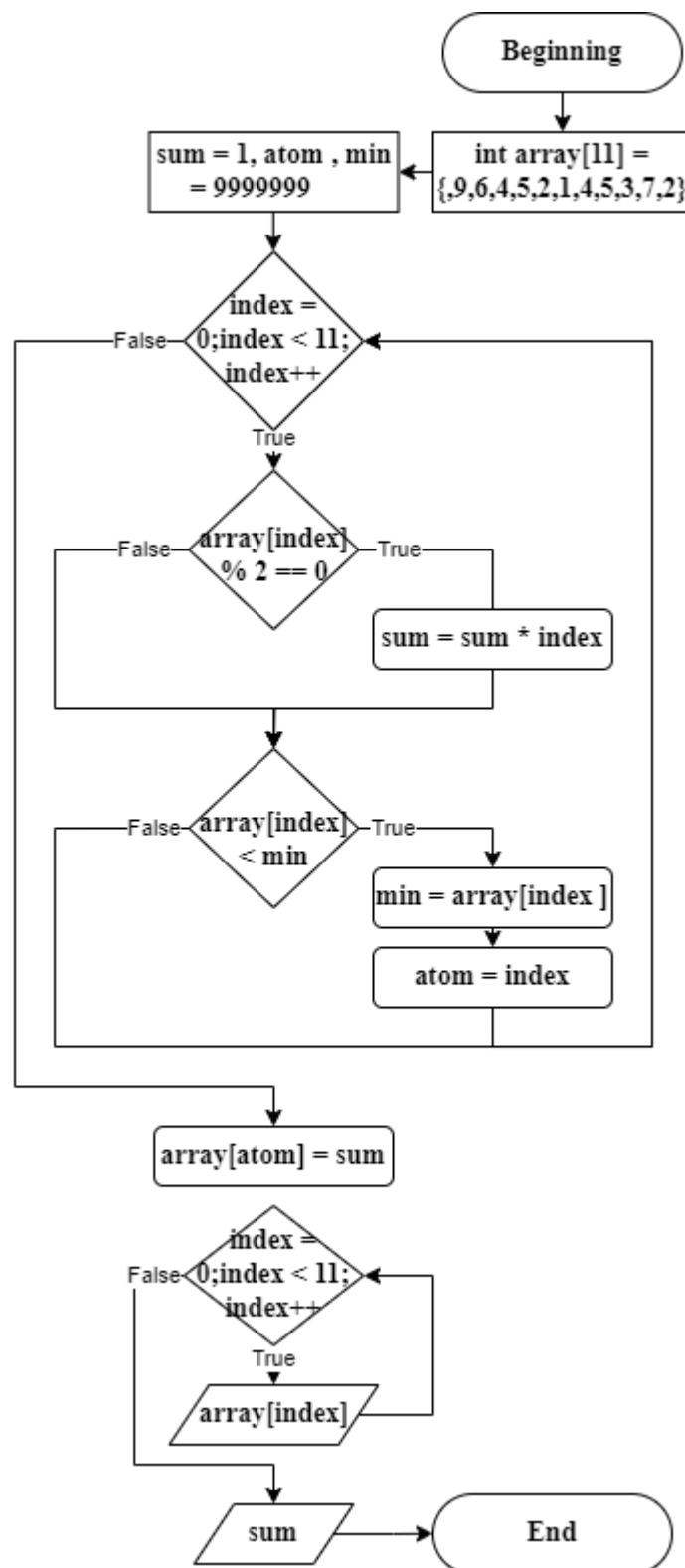
Результат виконання програми:

```
C:\Users\FearlessAtom\Desktop>task4.exe
9 6 4 5 2 480 4 5 3 7 2
Sum is: 480
```

Рис. 5



Блок-схема:



Github link : [FearlessAtom/Lab11 \(github.com\)](https://github.com/FearlessAtom/Lab11)

**Висновок:** Під час виконання лабораторної роботи №11, було успішно досліджено і застосовано концепції роботи з одновимірними масивами в мові програмування C. У результаті виконання завдань, було розроблено програми, що включали генерацію випадкових чисел, обробку цих чисел та виведення результатів на екран.