

# 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:

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

Рис. 1. Завдання для написання першої програми

### Лістинг програми:

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>
#include <windows.h>

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    srand(time(NULL));
    int gen, a = 0, b = 0;
    double c = 0;
    printf("Виберіть варіант генератору псевдовипадкових чисел:\n1) [-4; -1).      2) [100; 299].  3) [-35; -1] парні.\n4) [-128; 127].  5) [-7; 13) непарні.  ");
    printf("6) [-7.85; 28.3).\n7) [-100; 100].  8) [23; 71).  9) [0; 2).  10) [sqrt(17); sqrt(82)).\n");
    scanf_s("%d", &gen);
    switch (gen) {
        case 1: a = rand() % 3 - 4; break;
        case 2: a = 100 + rand() % (299 - 100 + 1); break;
        case 3: do b = -35 + rand() % (-1 - (-35) + 1); while (b % 2 != 0); break;
        case 4: a = -128 + rand() % (127 + 128 + 1); break;
        case 5: do b = -7 + rand() % (13 + 7); while (b % 2 == 0); break;
```

					ДУ «Житомирська політехніка».22.122.13.000 – Лр11			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Черниш М.			Звіт з лабораторної роботи		Лім.	Арк.
Перевір.		Терещук С.О.						1
Керівник								6
Н. контр.							ФІКТ Гр. КН-22-3[2]	
Зав. каф.								

```

case 6: c = -7.85 + (double)rand() * (28.3 - (-7.85)) / RAND_MAX; break;
case 7: a = rand() % (201 - 100); break;
case 8: a = 23 + rand() % (71 - 23); break;
case 9: a = rand() % 2; break;
case 10: c = sqrt(17) + (double)rand() * (sqrt(82) - sqrt(17)) / RAND_MAX; break;
default: printf("Помилка: введене число не в діапазоні від 1 до 10\n");
}
if (gen == 3 || gen == 5) printf("r = %d\n", b);
else if (gen == 1 || gen == 2 || gen == 4 || gen == 7 || gen == 8 || gen == 9)
printf("r = %d\n", a);
else if (gen == 6 || gen == 10) printf("r = %.2f\n", c);
return 0;
}

```

Результат виконання програми:

```

Виберіть варіант генератора псевдовипадкових чисел:
1) [-4; -1). 2) [100; 299]. 3) [-35; -1] парні.
4) [-128; 127]. 5) [-7; 13] непарні. 6) [-7.85; 28.3].
7) [-100; 100]. 8) [23; 71]. 9) [0; 2). 10) [sqrt(17); sqrt(82)].
6
r = -3.32
D:\КН-22-3[2]\Основи програмування\Лабораторні роботи з ОП\ОП_Lab_11\х64

```

Рис. 2. Результат виконання першої програми

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.

№	а	б
13	[13, 900]	[-1, 1] з точністю до сотих

Рис. 3. Завдання для написання другої програми

Лістинг програми:

```

#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>
#include <windows.h>

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    srand(time(NULL));
    int m, n, k, count = 0, a1;

```

```

double a2;
printf("Введіть кількість цілих випадкових чисел: ");
scanf_s("%d", &m);
printf("Введіть кількість дійсних випадкових чисел: ");
scanf_s("%d", &n);
printf("Введіть кількість випадкових чисел для виведення у рядок (3 <= k <= 10): ");
scanf_s("%d", &k);
for (; m >= 1; m--) {
    count++;
    a1 = 13 + rand() % (900 - 13 + 1);
    printf("%d ", a1);
    if (count == k) {
        printf("\n");
        count = 0;
    }
}
printf("\n_____ \n");
for (count = 0; n >= 1; n--) {
    count++;
    a2 = (double)rand() * (1 - (-1)) / RAND_MAX - 1;
    printf("%.2lf ", a2);
    if (count == k) {
        printf("\n");
        count = 0;
    }
}
return 0;
}

```

Результат виконання програми:

```

Введіть кількість цілих випадкових чисел: 16
Введіть кількість дійсних випадкових чисел: 16
Введіть кількість випадкових чисел для виведення у рядок (3 <= k <= 10): 4
25 149 380 771
499 159 690 404
803 258 359 64
288 693 64 237

-----
-1.00 0.03 -0.87 -0.41
-0.94 0.82 -0.08 -0.26
-0.94 -0.71 -0.61 -0.27
0.67 -0.84 -0.71 0.64
D:\КН-22-3[2]\Основи програмування\Лабораторні роботи з ОП\ОР_Lab_11\x64\Debug

```

Рис. 4. Результат виконання другої програми

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.

13	<p>В одновимірному масиві, що складається з <math>n</math> дійсних елементів, обчислити:</p> <ol style="list-style-type: none"> <li>1) кількість елементів, що лежать в діапазоні від <math>A</math> до <math>B</math> (<math>A</math> і <math>B</math> – з клавіатури);</li> <li>2) суму елементів масиву, розташованих після максимального елемента.</li> </ol>
----	---

Рис. 5. Завдання для написання третьої програми

Лістинг програми:

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>
#include <windows.h>

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    srand(time(NULL));
    int n, count = 0, maxi = 0;
    double array[1000], a, b, summ = 0, max = 0, A, B;
    printf("Введіть кількість елементів масиву: ");
    scanf_s("%d", &n);
    printf("Введіть перше число 'a' проміжку випадкових чисел [a, b], де a < b: ");
    scanf_s("%lf", &a);
    printf("Введіть останнє число 'b' проміжку випадкових чисел [a, b], де a < b: ");
    scanf_s("%lf", &b);
    if (a < b) {
        for (int i = 0; i < n; i++) {
            array[i] = a + (double)rand() * (b - (a)+1) / RAND_MAX;
            printf("arr[%d] = %.2f", i, array[i]);
            count++;
            if (count == 3) {
                printf("\n");
                count = 0;
            }
        }
    }
    else printf("\nПомилка: введене число 'a' не менше 'b'\n");
    printf("\n\nВведіть перше число 'A' проміжку випадкових чисел [A, B], де A < B: ");
    scanf_s("%lf", &A);
    printf("Введіть останнє число 'B' проміжку випадкових чисел [A, B], де A < B: ");
    scanf_s("%lf", &B);
    if (A < B) {
        count = 0;
        for (int i = 0; i < n; i++) {
```

```

        if ((array[i] >= A) && (array[i] <= B)) count++;
        if (array[i] > max) {
            max = array[i];
            maxi = i;
        }
    }
    for (int i = maxi + 1; i < n; i++)
        summ += array[i];
}
else printf("\nПомилка: введене число 'A' не менше 'B'\n");
printf("\nКількість елементів в діапазоні [%g, %g] = %d\n", A, B, count);
printf("Максимальний елемент = %.2f. Його індекс = %d\n", max, maxi);
printf("Сума елементів масиву, розташованих після максимального = %.2f\n", summ);
return 0;
}

```

Результат виконання програми:

```

Введіть кількість елементів масиву: 9
Введіть перше число 'a' проміжку випадкових чисел [a, b], де a < b: 10
Введіть останнє число 'b' проміжку випадкових чисел [a, b], де a < b: 100
arr[0] = 39.89    arr[1] = 68.02    arr[2] = 73.64
arr[3] = 48.48    arr[4] = 46.10    arr[5] = 94.73
arr[6] = 82.36    arr[7] = 82.29    arr[8] = 59.33

Введіть перше число 'A' проміжку випадкових чисел [A, B], де A < B: 20
Введіть останнє число 'B' проміжку випадкових чисел [A, B], де A < B: 50

Кількість елементів в діапазоні [20, 50] = 3
Максимальний елемент = 94.73. Його індекс = 5
Сума елементів масиву, розташованих після максимального = 223.97

D:\КН-22-3[2]\Основи програмування\Лабораторні роботи з ОП\ОР_Lab_11\x64\Debug

```

Рис. 6. Результат виконання третьої програми

Завдання 4. Написати програму:

13	У заданому масиві E[13] визначити різницю між середнім арифметичним перших п'яти елементів та сумою квадратів решти елементів масиву.
----	---

Рис. 7. Завдання для написання четвертої програми

Лістинг програми:

```

#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>
#include <windows.h>

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    srand(time(NULL));
}

```

		Черниш М.			ДУ «Житомирська політехніка».22.122.13.000 – Лр11	Арк.
		Терещук С.О.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

```

int arrE[13], count = 0, summ1 = 0, summ2 = 0;
double dif, average;
printf("Дано масив E[13]:\n");
for (int i = 0; i < 13; i++) {
    count++;
    arrE[i] = rand() % 21 - 10;
    printf("arrE[%d] = %d  ", i, arrE[i]);
    if (count == 3) {
        printf("\n");
        count = 0;
    }
    if (i < 5) summ1 += arrE[i];
    else if (i >= 5) summ2 += pow(arrE[i], 2);
}
average = summ1 / 5.0;
dif = average - summ2;
printf("\nСереднє арифметичне перших п'яти елементів = %.2f\n", average);
printf("Сума квадратів решти елементів масиву = %d\n", summ2);
printf("Їхня різниця = %.2f\n", dif);
return 0;
}

```

Результат виконання програми:

```

Дано масив E[13]:
arrE[0] = 6   arrE[1] = -4   arrE[2] = -4
arrE[3] = 6   arrE[4] = -10  arrE[5] = -7
arrE[6] = 6   arrE[7] = -6   arrE[8] = -9
arrE[9] = -4  arrE[10] = -7  arrE[11] = 1
arrE[12] = -1
Середнє арифметичне перших п'яти елементів = -1.20
Сума квадратів решти елементів масиву = 269
Їхня різниця = -270.20

D:\КН-22-3[2]\Основи програмування\Лабораторні роботи

```

Рис. 8. Результат виконання четвертої програми

		Черниш М.			ДУ «Житомирська політехніка».22.122.13.000 – Лр11	Арк.
		Терещук С.О.				6
Змн.	Арк.	№ докум.	Підпис	Дата		