

## **Лабораторная работа №4**

### **Тема: «Массивы»**

**Цель работы:** сформировать навыки и умения обработки структурированных типов данных, организованных в виде некоторой линейной последовательности, а также организованных в виде матрицы.

**Время выполнения:** 4 часа.

### **Теоретические сведения**

Пример инициализации массива

```
string students[10] = {  
    "Иванов", "Петров", "Сидоров",  
    "Ахмедов", "Ерошкин", "Выхин",  
    "Андеев", "Вин Дизель", "Картошкин", "Чубайс"  
};
```

#### **Описание синтаксиса**

Массив создается почти так же, как и обычная переменная. Для хранения десяти фамилий нам нужен массив, состоящий из 10 элементов. Количество элементов массива задается при его объявлении и заключается в квадратные скобки.

Чтобы описать элементы массива сразу при его создании, можно использовать фигурные скобки. В фигурных скобках значения элементов массива перечисляются через запятую. В конце закрывающей фигурной скобки ставится точка с запятой.

Попробуем вывести наш массив на экран с помощью оператора cout.

```
#include <iostream>
```

```
#include <string>
```

```
int main()
```

```
{
```

```
    std::string students[10] = {
```

```
        "Иванов", "Петров", "Сидоров",
```

```
        "Ахмедов", "Ерошкин", "Выхин",
```

```
        "Андеев", "Вин Дизель", "Картошкин", "Чубайс"
```

```
    };
```

```
    std::cout << students << std::endl; // Пытаемся вывести весь массив
```

непосредственно

```
    return 0;
```

```
}
```

Скомпилируйте этот код и посмотрите, на результат работы программы. Готово? А теперь запустите программу еще раз и сравните с предыдущим результатом. В моей операционной системе вывод был следующим:

Первый вывод: 0x7ffff8b85820

Второй вывод: 0x7fff7a335f90

Третий вывод: 0x7ffff847eb40

Мы видим, что выводится адрес этого массива в оперативной памяти, а никакие не «Иванов» и «Петров».

Дело в том, что при создании переменной, ей выделяется определенное место в памяти. Если мы объявляем переменную типа `int`, то на машинном уровне она описывается двумя параметрами — ее адресом и размером хранимых данных.

Массивы в памяти хранятся таким же образом. Массив типа `int` из 10 элементов описывается с помощью адреса его первого элемента и количества байт, которое может вместить этот массив. Если для хранения одного целого числа выделяется 4 байта, то для массива из десяти целых чисел будет выделено 40 байт.

Так почему же, при повторном запуске программы, адреса различаются? Это сделано для защиты от атак переполнения буфера. Такая технология называется рандомизацией адресного пространства и реализована в большинстве популярных ОС.

Попробуем вывести первый элемент массива — фамилию студента Иванова.

```
#include <iostream>
```

```
#include <string>
```

```
int main()
```

```
{
```

```
    std::string students[10] = {
```

```
        "Иванов", "Петров", "Сидоров",
```

```
        "Ахмедов", "Ерошкин", "Выхин",
```

```
        "Андеев", "Вин Дизель", "Картошкин", "Чубайс"
```

```
    };
```

```

    std::cout << students[0] << std::endl;
    return 0;
}

```

Смотрим, компилируем, запускаем. Убедились, что вывелся именно «Иванов». Заметьте, что нумерация элементов массива в C++ начинается с нуля. Следовательно, фамилия первого студента находится в `students[0]`, а фамилия последнего — в `students[9]`.

В большинстве языков программирования нумерация элементов массива также начинается с нуля.

Попробуем вывести список всех студентов. Но сначала подумаем, а что если бы вместо группы из десяти студентов, была бы кафедра их ста, факультет из тысячи, или даже весь университет? Ну не будем же мы писать десятки тысяч строк с `cout`?

Вывод элементов массива через цикл

```

#include <iostream>
#include <string>

int main()
{
    std::string students[10] = {
        "Иванов", "Петров", "Сидоров",
        "Ахмедов", "Ерошкин", "Выхин",
        "Андеев", "Вин Дизель", "Картошкин", "Чубайс"
    };
    for (int i = 0; i < 10; i++) {
        std::cout << students[i] << std::endl;
    }

    return 0;
}

```

Если бы нам пришлось выводить массив из нескольких тысяч фамилий, то мы бы просто увеличили конечное значение счетчика цикла — строку `for (...; i < 10; ...)` заменили на `for (...; i < 10000; ...)`.

Заметьте, что счетчик нашего цикла начинается с нуля, а заканчивается девяткой. Если вместо оператора строгого неравенства — `i < 10` использовать оператор «меньше, либо равно» — `i <= 10`, то на последней итерации

программа обратится к несуществующему элементу массива — students[10]. Это может привести к ошибкам сегментации и аварийному завершению программы. Будьте внимательны — подобные ошибки бывает сложно отловить.

Массив, как и любую переменную можно не заполнять значениями при объявлении.

### ***Объявление массива без инициализации***

```
string students[10];
```

```
// или
```

```
string teachers[5];
```

Элементы такого массива обычно содержат в себе «мусор» из выделенной, но еще не инициализированной, памяти. Некоторые компиляторы, такие как GCC, заполняют все элементы массива нулями при его создании.

При создании статического массива, для указания его размера может использоваться только константа. Размер выделяемой памяти определяется на этапе компиляции и не может изменяться в процессе выполнения.

```
int n;
```

```
cin >> n;
```

```
string students[n]; /* Неверно */
```

Выделение памяти в процессе выполнения возможно при работе с динамическими массивами. Но о них немного позже.

Заполним с клавиатуры пустой массив из 10 элементов.

### ***Заполнение массива с клавиатуры***

```
#include <iostream>
```

```
#include <string>
```

```
using std::cout;
```

```
using std::cin;
```

```
using std::endl;
```

```
int main()
```

```
{
```

```
    int arr[10];
```

```

// Заполняем массив с клавиатуры
for (int i = 0; i < 10; i++) {
    cout << "[" << i + 1 << "]" << ": ";
    cin >> arr[i];
}

// И выводим заполненный массив.
cout << "\nВаш массив: ";

for (int i = 0; i < 10; ++i) {
    cout << arr[i] << " ";
}

cout << endl;

return 0;
}

```

### ***Индивидуальные задания к лабораторной работе №4***

#### ***Задание 1. Одномерные массивы***

1. В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить сумму отрицательных элементов массива и сравнить с введенным значением  $C$ .
2. В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить произведение элементов с четными номерами.
3. В одномерном массиве, состоящем из  $n$  элементов, вычислить максимальный элемент и определить его номер (индекс).
4. Преобразовать массив  $A$  размером  $n$  в два таким образом, чтобы в одном хранились положительные элементы, а во втором – отрицательные.
5. В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить произведение минимального и  $K$ -ого элементов ( $K$  вводится с клавиатуры).
6. Все элементы массива  $K$  размером  $n$ , модуль которых не превышает 1, заменить 0.
7. Преобразовать массив  $K$  размером  $n$  таким образом, чтобы в первой его половине располагались элементы, стоявшие в нечетных позициях, а во второй половине – элементы, стоявшие в четных позициях.
8. В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить максимальный по модулю элемент.

9. Преобразовать массив  $K$  размером  $n$  таким образом, чтобы элементы, равные нулю, располагались после всех остальных.
10. Вычислить сумму модулей элементов массива  $K$  размером  $n$ , расположенных после минимального элемента.
11. В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить количество элементов, равных 0.
12. В одномерном массиве, состоящем из  $n$  элементов, вычислить количество элементов, больших  $C$ .
13. В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить произведение элементов массива, расположенных перед максимальным по модулю элементом.
14. В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить сумму модулей элементов массива, расположенных перед минимальным по модулю элементом.
15. В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить количество элементов, меньших  $C$ .
16. В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить сумму положительных элементов массива, расположенных до максимального элемента.
17. В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить максимальный элемент и поменять местами с первым.
18. В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить минимальный элемент и поменять его с последним.
19. В одномерном массиве, состоящем из  $n$  вещественных элементов, разместить элементы таким образом, чтобы сначала шли элементы, большие шести.
20. В одномерном массиве, состоящем из  $n$  целых чисел, заменить элементы с четными номерами нулем, а элементы с нечетными номерами - 1.
21. В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить количество элементов, входящих в промежуток  $M$  и  $P$  ( $M$  и  $P$  вводятся с клавиатуры).
22. В одномерном массиве, состоящем из  $n$  вещественных элементов, заменить каждый элемент  $> 0$ , значением среднего арифметического элементов массива.
23. Дан массив из  $n$  целых чисел. Посчитать, сколько раз в нем встречается минимальный по модулю элемент.
24. Дан массив из  $k$  целых чисел. Переменной  $t$  присвоить значение true, если в массиве нет нулевых элементов, и false – в противном случае.

25. Дан массив размерностью 10 из целых чисел. Сформировать два массива размерностью 5, включив в первый массив элементы с четными индексами, а во второй – с нечетными.

26. Задан массив  $A$  размером  $n$ . Найдите в нем первую пару последовательных элементов, таких, что  $a_i \cdot a_{i+1} < 0$ .

27. Задан массив  $A$  размером  $n$ . Найдите суммы четных и нечетных элементов этого массива.

28. Задан массив  $A$  размером  $n$  из неравных нулю чисел. Определите знак произведения элементов массива не перемножая их.

29. Задан массив  $A$  размером  $n$ . Поменяйте местами элементы массива, равноудаленные от концов массива, например:  $a_1$  и  $a_n$ ,  $a_2$  и  $a_{n-1}$ .

Задан массив  $A$  размером  $n$  из положительных и отрицательных чисел. Определите число перемен знака при переходе от  $a_i$  к  $a_{i+1}$ .

### ***Задание 2. Двумерные массивы.***

1. Дан двумерный массив  $5 \times 5$ . Заменить нулем элементы, расположенные под главной диагональю.

2. Дан двумерный массив целых чисел. Вычислить сумму элементов, расположенных на главной и побочной диагоналях.

3. Дан двумерный массив целых чисел. Найти произведение максимального и минимального элементов.

4. Написать программу, которая вычисляет сумму элементов двумерного массива по столбцам.

5. Найти среднее арифметическое элементов двумерного массива.

6. Написать программу, которая определяет номер строки квадратной матрицы, сумма элементов которой максимальна (минимальна).

7. Дан массив размерностью  $n \times m$ . Поделить все элементы каждой строки на наибольший по модулю элемент этой строки.

8. Дан двумерный массив  $5 \times 5$ . Заменить нулем элементы, расположенные над главной диагональю.

9. Написать программу, которая вычисляет сумму элементов двумерного массива по строкам.

10. Написать программу, которая определяет номер столбца квадратной матрицы, сумма элементов которого максимальна.

11. Дан двумерный массив  $5 \times 5$ . Заменить нулем все его отрицательные элементы.

12. Дан массив размерностью  $n \times m$ . Поделить все элементы каждого столбца на наименьший по модулю элемент этого столбца.

13. Дан двумерный массив  $5 \times 5$ . Заменить нулем элементы, расположенные над побочной диагональю.

14. Дан двумерный массив. Получить новый массив путем деления всех элементов массива на наименьший по модулю элемент.

15. Дан двумерный массив. Определить количество строк, в которых присутствует хотя бы один отрицательный элемент.

16. Дан двумерный массив. Получить новый массив путем деления всех элементов строки на наименьший по модулю элемент строки. Если наименьший элемент некоторой строки равен 0, то элементы этой строки оставить без изменений.

17. Дан двумерный массив размерностью  $5 \times 5$ . Вычислить сумму элементов, номера строк у которых четные.

18. Присвоить переменной  $t$  значение `true`, если среднее арифметическое элементов двумерного массива больше произведения элементов, находящихся на главной диагонали. В обратном случае присвоить значение `false`.

19. Дан двумерный массив размерностью  $n \times m$ , в котором не все элементы равны нулю. Получить новый массив путем деления всех элементов массива на его наименьший элемент. Если наименьший элемент равен нулю, то вывести соответствующее сообщение, а элементы массива оставить без изменения.

20. Дан квадратный массив размерностью  $5 \times 5$ . Найти произведение элементов массива, расположенных в строках, которые начинаются с отрицательного элемента.

21. Дан двумерный массив размерностью  $5 \times 5$ . Вычислить сумму элементов, номер столбца которых нечетный.

22. Дан двумерный массив вещественных чисел размерностью  $n \times m$ . В каждой строке найти минимальный элемент, а среди них – максимальный.

23. Найти среднее арифметическое максимального и минимального по модулю элементов массива.

24. Дан двумерный массив размерностью  $5 \times 5$ . Получить новый массив путем деления всех элементов массива на наибольший по модулю элемент.

25. Дан двумерный массив размерностью  $5 \times 5$ . Заменить нулевые элементы, расположенные под побочной диагональю.

26. Дан двумерный массив размерностью  $5 \times 5$ . Заменить все элементы массива =  $C$  нулями. В случае если таких элементов нет, вывести соответствующее сообщение.

27. Дан двумерный массив вещественных чисел размерностью  $n \times m$ . В каждом столбце найти максимальный элемент, а среди них – минимальный.

28. Дан двумерный массив  $5 \times 5$ . Заменить нулевые элементы, расположенные над побочной диагональю, на -1.



29. Дан двумерный массив размером  $n \times m$ , заполненный случайным образом. Определить, есть ли в данном массиве строки, в которых есть отрицательные элементы.

30. Дан двумерный массив размерностью  $3 \times 4$ , заполненный целыми числами с клавиатуры. Сформировать одномерный массив, каждый элемент которого равен произведению четных положительных элементов.

### **ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ**

1. Изучить теоретическую часть лабораторной работы.
2. Реализовать индивидуальное задание по вариантам, представленные в теоретических сведениях, сделать скриншоты работающих программ. Написать комментарии.
3. Написать отчет, содержащий:
  1. Титульный лист, на котором указывается:
    - а) полное наименование министерства образования и название учебного заведения;
    - б) название дисциплины;
    - в) номер практического занятия;
    - г) фамилия преподавателя, ведущего занятие;
    - д) фамилия, имя и номер группы студента;
    - е) год выполнения лабораторной работы.
  2. Индивидуальное задание из раздела «Теоретические сведения» с кодом, комментариями и скриншотами работающих программ.
  3. Построение блок-схем.