

Практическая работа № 5

Тема: Структурированные типы данных. Одномерные массивы

Цель работы:

1. Ознакомиться со структурированными типами данных C++
2. Построить графический алгоритм и реализовать программу на языке C++ на тему «Одномерные массивы» .

Теоретические сведения

1. Понятие структурированные типы

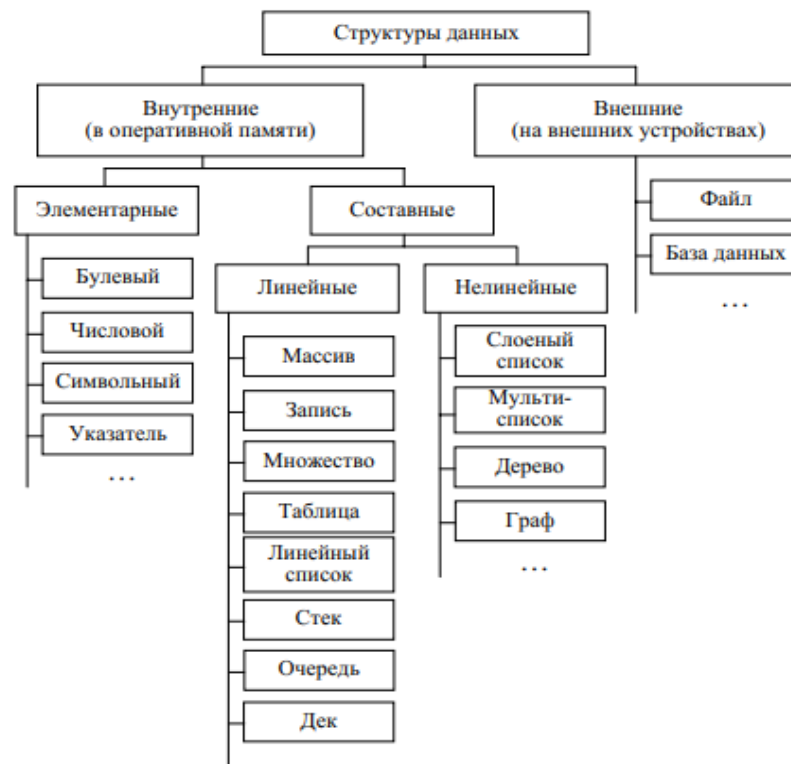
Различают простые примитивны структуры данных и составные интегрированные, сложные).

К простым типам относятся стандартные типы данных такие как:

- integer,
- float,
- boolean,
- char,
- указатели.

Составные структуры данных содержат части, которые также содержат другие структуры данных- простые или составные.

Классификацию структур данных можно переставить в виде такой блок-схемы:



Простейшими структурами данных Си++ являются массивы, строки, множества

и записи. Более ложную нелинейную структуру имеют графы и деревья. На сегодняшней лекции мы рассмотрим одномерные массивы

2. Одномерные массивы

Сравнительно часто надо обрабатывать большие объемы данных. Например, программа должна считать из файла возраст учащихся в группе из 14 человек, отсортировать их в порядке возрастания, а затем вывести на экран. Для этого необходимо, чтобы все данные одновременно находились в памяти, иначе отсортировать их будет проблематично. Использовать для этих целей 14 различных целочисленных переменных крайне неудобно хотя бы по одной, вполне очевидной причине – а что если учащихся станет больше или меньше? Для таких целей в языке Си существуют структурированный тип данных — массив.

Массивы хранят элементы одного и того же базового типа (например, массив целых чисел, или массив действительных чисел), доступ к которым осуществляется по их порядковым номерам.

Пример инициализации массива: <тип> <имя>[<размер>]

`int mas[14];` // массив из 14 целочисленных переменных

`double degrees[100];` // массив из 100 переменных типа `doubl`

Мы также можем смешивать в одном операторе объявление обычных переменных и массивов:

`int i, j = 1, buf[100], buf2[100], count = 0 ;`

Чтобы получить доступ к элементу массива, необходимо сначала записать имя массива, а затем в квадратных скобках – порядковый номер того элемента, к которому мы хотим обратиться.

Например:

`mas[2] = 4;`

`degrees[28] = 4.5`

В языке C++, в отличие от, например, языка Pascal, нумерация элементов в массиве начинается с нуля, поэтому в примере выше первый элемент массива `mas` – `mas[0]`, последний – `mas[13]`, а `mas[2]` обращается к третьему элементу по порядку. Как правило, номера элементов массива называются индексами, а процесс доступа к ним – индексированием.

В C++ различают статически и динамические массивы. При работе со статическими массивами область памяти под них выделяется на момент инициализации потому следует заранее задать количество элементов массиве через константу.

При работе с динамическими массивами место под массив выделяется в памяти по команде программиста.

В памяти компьютера массив представляет собой последовательность подряд идущих переменных заданного типа. Например, `char str[10]` – это 10 подряд идущих байт в которые можно записать символы, а `int mas[3]` это три подряд идущие целочисленные переменные, каждая из которых занимает по 4

байта, то есть всего этот массив занимает 12 байт.

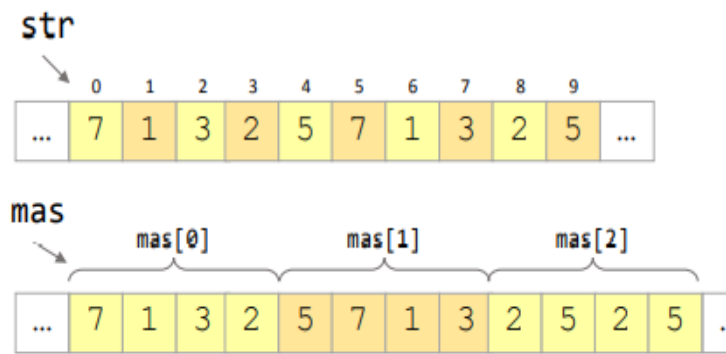


Рис. 7.2: Массивы `char str[10]` и `int mas[3]` в памяти

Рис. Примеры массивов

Значения элементов массива можно задать на момент объявления массива:

```
int mas[5] = { 1, 2, 3, 4, 5 };
```

Запись `int mas[5] = { 1, 2, 3, 4, 5, 6, 7, 8 };` вызовет ошибку компиляции так как значений в списке инициализации больше, чем элементов массива.

Второй пример: `int mas[5] = { 1, 2 };` Эта запись ошибки не вызовет. В первый элемент будет записана единица, во второй – двойка, а в три оставшихся – ноль.

Отсюда следует простой способ обнуления массива любого размера:

```
int mas[256] = { 0 }
```

Примеры ввода одномерного массива:

Ввод массива с клавиатуры:

```
#include <iostream>
#include <locale>
#include <cmath> // подключение библиотеки
using namespace std;

int main()
{
    setlocale(LC_ALL, "Russian");
    const int n = 10;
    int mas[n], i;
    cout << "Введите " << n << " элементов массива" << endl;
    for (i = 0; i < n; i++)
        cin >> mas[i];
    cout << "Вывод массива";
    for (i = 0; i < n; i++)
        cout << mas[i] << endl;
}
```

Заполнение массива случайным образом:

```
#include <iostream>
#include <locale>
#include <ctime> // подключение библиотеки

using namespace std;
```

```

int main()
{
    setlocale(LC_ALL, "Russian");
    const int N = 10;
    int mas[N], i;
    srand(time(NULL)); //обнуление времени для новой комбинации случайных чисел
    for (i = 0; i < N; i++) mas[i] = rand();
    cout << " Вывод массива :";
    for (i = 0; i < N; i++)
        cout << mas[i] << ' ';
    return 0;
}

```

Заполнение массива случайным образом из элементов в заданном диапазоне.

Для генерации случайных чисел из конкретного диапазона, например [1,100]. Сначала с помощью операции нахождения остатка от деления `rand() % 100` отображаем все получаемые случайные числа на отрезок [0.. 99], а затем, прибавляя 1, получаем то, что требовалось диапазон [1..100].

В общем случае, если нужно получить числа из диапазона [a, b], формула будет выглядеть следующим образом: `int x = a + (rand() % (b - a + 1))`.

Если, например, нужно получить случайное вещественное число на единичном отрезке [0,1], то можно воспользоваться следующей формулой: `double x = (double)rand() / RAND_MAX`;

Задание: Требуется получить массив в с элементами в диапазоне -500 до 500.

Решение:

```

// В общем случае для получения x в заданном диапазоне [a..b ]
// используют выражение int x = a + (rand() % (b - a + 1)).
cout << "Вывод массива в заданном диапазоне -500 до 500" << endl;
for (i = 0; i < N; i++)
    mas[i] = -500 + rand() % 1001;
    for (i = 0; i < N; i++)
        cout << mas[i] << endl;
return 0;

```

Простейшие алгоритмы на тему одномерные массивы

Задание Получить массив случайным образом в диапазоне 0..20 найти максимальный элемент и его номер.

```

cout << "Вывод массива в заданном диапазоне 0..20" << endl;
for (i = 0; i < N; i++) {
    mas[i] = rand() % 21;
    cout << mas[i] << endl;
}
//поиск максимума
max = mas[0]; ind_max = 0;
for (i = 1; i < N; i++)
    if (mas[i] >= max) {
        max = mas[i]; ind_max = i;
    }
cout << "max=" << max << ' ' << "ind_max=" << ind_max;
return 0;

```

В задачах часто встречаются задания с заменами

Пример поменять местами максимальный и первый элемент массива

```

int temp;
temp = mas[ind_max]; mas[ind_max] = mas[0]; mas[0] = temp;
//вывести новый массив

```

```

for (i = 0; i < N; i++)
    cout << mas[i] << endl;
return 0;

```

Задание Получить массив случайным образом в диапазоне -20 ..20 найти сумму и произведение элементов массива.

```

....
for (i = 0; i < N; i++) {
    mas[i] = -20 + rand() % 41; //-20..20
    cout << mas[i] << endl;
}
//поиск суммы и произведения
sum = 0; p = 1;
for (i = 0; i < N; i++) {
    sum += mas[i]; p *= mas[i];
}
cout << "sum=" << sum << ' ' << "p=" << p << endl;
return 0;
}

```

Пример

Ввод массива rand

```

#include <iostream>
#include <locale>
#include <ctime> // подключение библиотеки srand

```

```

using namespace std;

```

```

int main()
{
    setlocale(LC_ALL, "Russian");
    const int N = 10;
    int mas[N], i;
    srand(time(NULL));
    for (i = 0; i < N; i++) mas[i] = rand();
    cout << " Вывод массива  :";
    for (i = 0; i < N; i++)
        cout << mas[i] << ' ';
    // В общем случае для получения x в заданном диапазоне [a..b ]
    // используют выражение int x = a + (rand() % (b - a + 1)).
    cout << "Вывод массива в заданном диапазоне -500 до 500" << endl;
    for (i = 0; i < N; i++)
        mas[i] = -500 + rand() % 1001;
    for (i = 0; i < N; i++)
        cout << mas[i] << endl;
    return 0;
}

```

Задания на тему одномерные массивы

Задание: Разработать графический алгоритм и код программы на с++ по теме статические «Одномерны массивы». Число n задать как `const`.

Номер варианта выбирать в соответствии с номером в списке группы.

Вариант 1

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1. Сумму отрицательных элементов массива.
2. Произведение элементов массива, расположенных между максимальным и минимальным элементами.
3. Упорядочить элементы массива по возрастанию.

Вариант 2

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1. Сумму положительных элементов массива.
2. Произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами.
3. Упорядочить элементы массива по убыванию.

Вариант 3

В одномерном массиве, состоящем из n целочисленных элементов, вычислить:

1. Произведение элементов массива с четными номерами.
2. Сумму элементов массива, расположенных между первым и последним нулевыми элементами.
3. Преобразовать массив таким образом, чтобы сначала располагались все положительные элементы, а потом — все отрицательные (элементы, равные нулю, считать положительными).

Вариант 4

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1. Сумму элементов массива с нечетными номерами.
2. Сумму элементов массива, расположенных между первым и последним отрицательными элементами.
3. Сжать массив, удалив из него все элементы, модуль которых не превышает единицу. Освободившиеся в конце массива элементы заполнить нулями.

Вариант 5

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1. Максимальный элемент массива.
2. Сумму элементов массива, расположенных до последнего положительного элемента.
3. Сжать массив, удалив из него все элементы, модуль которых находится

в интервале $[a, b]$. Освободившиеся в конце массива элементы заполнить нулями.

Вариант 6

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1. Минимальный элемент массива.
2. Сумму элементов массива, расположенных между первым и последним положительными элементами.
3. Преобразовать массив таким образом, чтобы сначала располагались все элементы, равные нулю, а потом — все остальные.

Вариант 7

В одномерном массиве, состоящем из n целочисленных элементов, вычислить:

1. Номер максимального элемента массива.
2. Произведение элементов массива, расположенных между первым и вторым нулевыми элементами.
3. Преобразовать массив таким образом, чтобы в первой его половине располагались элементы, стоявшие в нечетных позициях, а во второй половине — элементы, стоявшие в четных позициях.

Вариант 8

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1. Номер минимального элемента массива.
2. Сумму элементов массива, расположенных между первым и вторым отрицательными элементами.
3. Преобразовать массив таким образом, чтобы сначала располагались все элементы, модуль которых не превышает единицу, а потом — все остальные.

Вариант 9

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1. Максимальный по модулю элемент массива.
2. Сумму элементов массива, расположенных между первым и вторым положительными элементами.
3. Преобразовать массив таким образом, чтобы элементы, равные нулю, располагались после всех остальных.

Вариант 10

В одномерном массиве, состоящем из n целочисленных элементов, вычислить:

1. Минимальный по модулю элемент массива.
2. Сумму модулей элементов массива, расположенных после первого элемента, равного нулю.
3. Преобразовать массив таким образом, чтобы в первой его половине рас-

полагались элементы, стоявшие в четных позициях, а во второй половине — элементы, стоявшие в нечетных позициях.

Вариант 11

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1. Номер минимального по модулю элемента массива.
2. Сумму модулей элементов массива, расположенных после первого отрицательного элемента.
3. Сжать массив, удалив из него все элементы, величина которых находится в интервале $[a, b]$. Освободившиеся в конце массива элементы заполнить нулями.

Вариант 12

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1. Номер максимального по модулю элемента массива.
2. Сумму элементов массива, расположенных после первого положительного элемента.
3. Преобразовать массив таким образом, чтобы сначала располагались все элементы, целая часть которых лежит в интервале $[a, b]$, а потом — все остальные.

Вариант 13

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1. Количество элементов массива, лежащих в диапазоне от A до B .
2. Сумму элементов массива, расположенных после максимального элемента.
3. Упорядочить элементы массива по убыванию модулей.

Вариант 14

В одномерном массиве, состоящем из n целочисленных элементов, вычислить:

1. Минимальный по модулю элемент массива.
2. Сумму модулей элементов массива, расположенных после первого элемента, равного нулю.
3. Преобразовать массив таким образом, чтобы элементы, равные нулю, располагались после всех остальных.

Вариант 15

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

1. Минимальный элемент массива.
2. Сумму элементов массива, расположенных между первым и последним положительными элементами.
3. Преобразовать массив таким образом, чтобы сначала располагались все элементы, меньше нуля, а потом — все остальные.