

ОДНОМЕРНЫЕ МАССИВЫ

Сортировка массива

Пример 1

Выполнить сортировку элементов массива по убыванию значений.

Алгоритм

1. Начальное значение **max** – первый элемент.
2. Начальное значение **imax** – индекс элемента (см. п.1).
3. Поиск **max** в массиве и определение его индекса.
4. Перестановка первого элемента и **max**.
5. Повтор п. 1-5 в укороченном на один элемент массиве.

1	10	3		150	200				100
---	----	---	--	-----	-----	--	--	--	-----

200	10	3		150	1				100
-----	----	---	--	-----	---	--	--	--	-----

200	150	3		10	1				100
-----	-----	---	--	----	---	--	--	--	-----

. . .

```
int imax, max;
```

```
int x[10];
```

```
for (int i = 0; i <= 9; i ++)
```

```
    cin>>x[i]; // Ввод элементов массива
```

```
for (int i = 0; i <=9; i++) // Цикл по первым
{
    // индексам массивов
    max=x[i]; imax=i;
    for (int j = i+1; j<=9; j++)
    {
        // Цикл поиска максимума
        if (x[j]>=max)
        {
            max=x[j];
            imax=j;
        }
    }
    x[imax]=x[i]; // Перестановка первого
    x[i]=max; // и максимального элементов
}
```

Окончание кода

```
cout<<endl; // Вывод пустой строки перед  
           // выводом результирующего массива
```

```
for (int i = 0; i <=9; i++)  
    cout<<x[i]<<'\\t'; // Вывод массива в строку
```

Пример 2

При помощи генератора случайных чисел заполнить массив. Размерность массива ввести с клавиатуры.

Отсортировать массив по возрастанию и вывести результат на экран.

В программе использовать сортировку методом пузырька.

Перестановка двух соседних элементов массива выполняется с использованием дополнительной переменной по следующей схеме:

$$\begin{aligned}R &= A(i) \\A(i) &= A(i+1) \\A(i+1) &= R\end{aligned}$$

Здесь R – дополнительная переменная.


```
#include <stdlib.h>
```

```
...
```

```
int N, i, K, R, c;
```

```
bool sort; // Объявление логического типа данных
```

```
cin>>N; // Ввод размерности массива
```

```
int A[N]; // Объявление массива
```

```
cout<<endl;
```

```
for (i = 0; i < N; i++)
```

```
{
```

```
    A[i] = rand(); // Заполнение массива
```

```
    // случайными числами от 0 до 32767
```

```
    cout<<A[i]<<"\t"; // Вывод элементов массива
```

```
}
```

```
cout<<endl;
```

```

c = 0; // Количество перестановок – начальное значение
do      // Цикл по сортировке массива
{ sort = false; // Сброс флага перестановки
  for (i = 0; i < N - 1; i++)
  {
      if (A[i] > A[i + 1])
      { R = A[i];           // Перестановка элементов
        A[i] = A[i + 1];
        A[i + 1] = R;
        sort = true;    // Включение флага перестановки
        c++; }
  }
}
while (sort);
  for (i = 1; i < N; i++) // Вывод отсортированного массива
    cout<<A[i]<<“\t”;
cout<<endl;
cout<<“Kol-vo perestанovok=”<<c<<endl;

```