

**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ**

**федеральное государственное автономное образовательное  
учреждение высшего образования  
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»  
(РУТ(МИИТ))**

Кафедра «Вычислительные системы, сети и информационная безопасность»

**ОТЧЕТ ПО ДИСЦИПЛИНЕ  
«АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ»  
ЛАБОРАТОРНАЯ РАБОТА №4**

*Направление: 10.03.01 Информационная безопасность*

*Профиль: Безопасность компьютерных систем*

Выполнил:  
студент группы УИБ-115  
Клепиков Степан Даниилович

Проверил:

\_\_\_\_\_  
(должность, ФИО)

\_\_\_\_\_  
(должность, ФИО)

Москва 2021 г.

## Задание

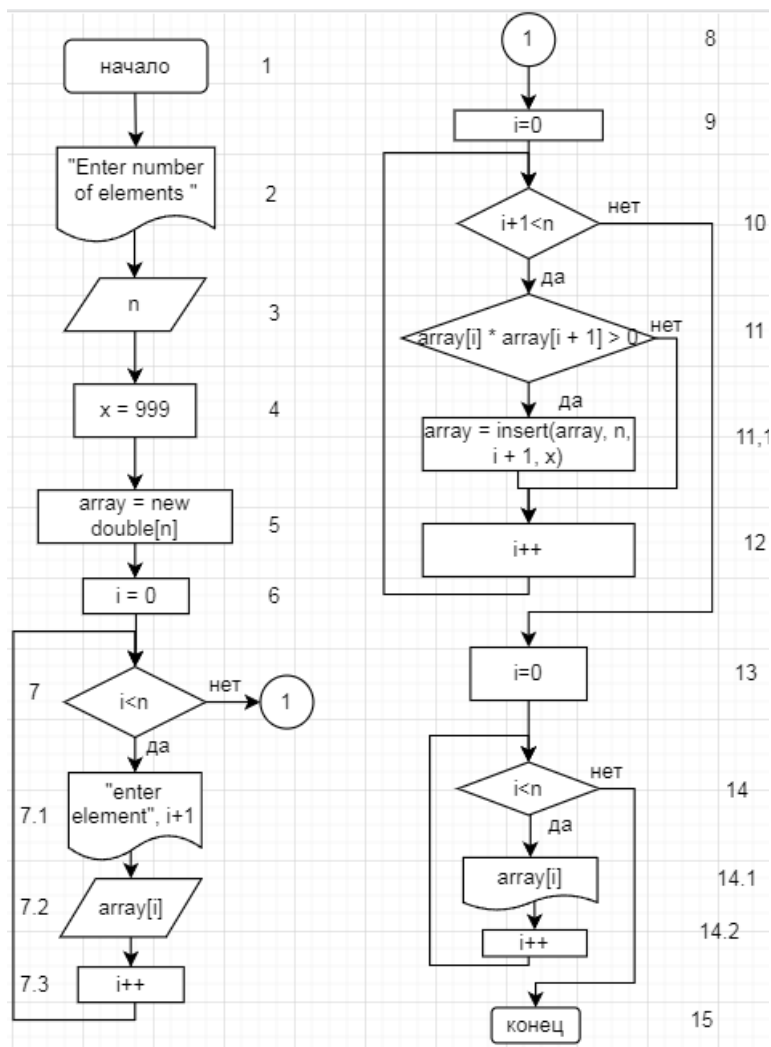
Задан одномерный массив целых чисел. Вставить число **999** между всеми элементами, имеющими одинаковый знак.

### 1. Таблица имён:

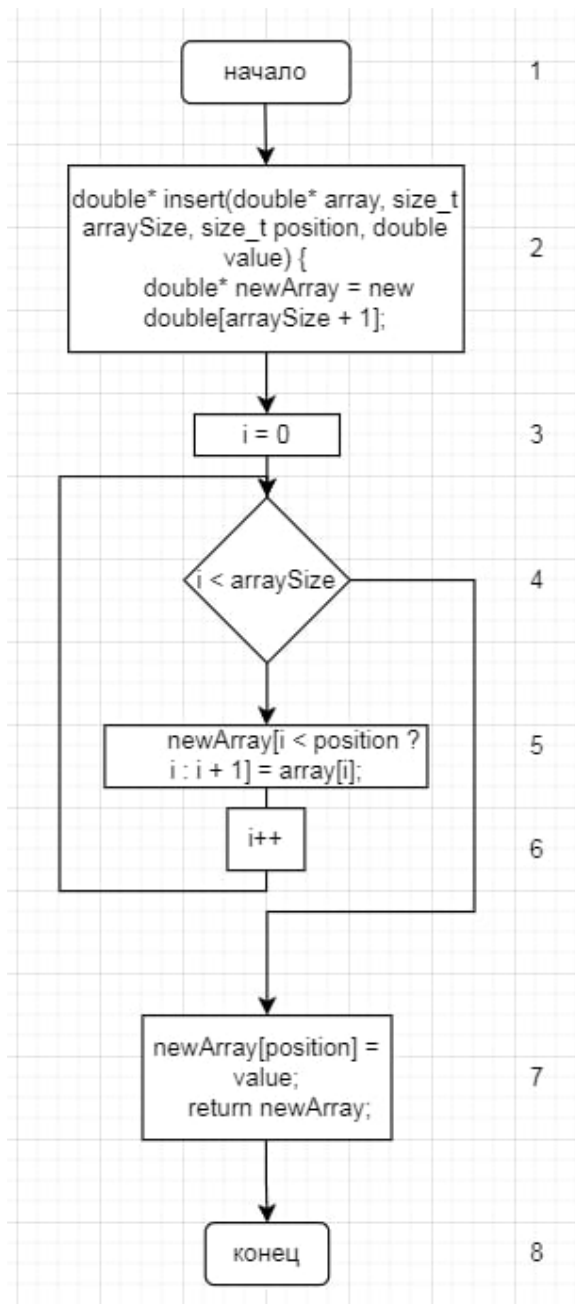
Исходные данные		
value	Double	-
n	Double	Количество элементов
x	Double	Вставляемое число
Рабочие переменные		
newArray	Double	-
Результат		
array[i]	double	-

### 2. Блок-схема:

(основная функция)



(блок-схема внешней функции)



### 3. Отладочный пример:

1. Начало
2. Вывод "enter number of elements"
3. Ввод n (4)
4. Присвоение переменной x значения 999
5. Присвоение переменной array значения `new double[n]`
6. Присвоение переменной `i` значения 0
7. Проверка условия  $(i < n)$        $(0 < 4) \Rightarrow$  истинно
  - 7.1 Вывод "enter element", `i+1`
  - 7.2 Ввод `array[i]` (1)
  - 7.3 Присвоение переменной `i` значения `i+1`
7. Проверка условия  $(i < n)$        $(0 < 4) \Rightarrow$  истинно

7.1 Вывод 'enter element', i+1

7.2 Ввод array[i] (2)

7.3 Присвоение переменной i значения i+1

.....

8. Выход из цикла ввода элементов, переход к циклу обработки

9. Присвоение переменной i значения 0

10. Проверка условия (i+1 < n) (1<4) - истинно

11. Проверка условия (array[i] \* array[i + 1] > 0) (1\*2>0) => истинно

11.1 вызов внешней функции и присвоение переменной array  
значения переменной insert(array, n, i + 1, x)

12. Присвоение переменной i значения i+1

13. Присвоение переменной i значения 0

14. Проверка условия (i<n) (0<4) => истинно

14.1 Вывод array[i]

14.2 Присвоение переменной i значения i+1

.....

15. Конец

#### 4. Код программы:

```
#include <iostream> //подключение библиотеки ввода-вывода
```

```
using namespace std; //подключение библиотеки для работы со  
строками
```

```
double* insert(double* array, size_t arraySize, size_t position, double  
value) { //объявление переменных типа double  
    double* newArray = new double[arraySize + 1]; //создание нового  
массива с размером на 1 больше
```

```
    for(size_t i = 0; i < arraySize; i++) { //цикл заполнения нового  
массива значениями старого
```

```
        newArray[i < position ? i : i + 1] = array[i];  
    }
```

```
    newArray[position] = value; //вставка нужного значения  
    return newArray; //возвращаем новый массив  
}
```

```
int main() {  
    size_t n; // присвоение переменной size_t значения n  
    double x; // объявление переменной типа integer  
    cout << "enter number of elements : ";  
    cin >> n; // ВВОД КОЛИЧЕСТВА СИМВОЛОВ
```

```

x = 999; // вставляемое число

double* array = new double[n];

for(size_t i = 0; i < n; i++) // цикл ввода элементов массива
{
    cout << "enter element [" << i + 1 << "] : ";
    cin >> array[i]; //ввод i-того элемента массива
}

for(size_t i = 0; i + 1 < n; i++) { // цикл обработки и сдвига
элементов массива
    if(array[i] * array[i + 1] > 0) { // проверка знака соседних
элементов
        array = insert(array, n, i + 1, x); // вставка между ними
переменной x (в случае, если условие выше верно)
        ++n;
        ++i;
    }
}

for(size_t i = 0; i < n; i++) // цикл вывода конечного массива
    cout << array[i] << " ";

cout << endl;
return 0; //успешное завершение программы
}

```

##### 5. Результат выполнения работы программы:

enter number of elements : 4	enter number of elements : 4	enter number of elements : 4
enter element [1] : 1	enter element [1] : 1	enter element [1] : -1
enter element [2] : 2	enter element [2] : -1	enter element [2] : -1
enter element [3] : 3	enter element [3] : 1	enter element [3] : -1
enter element [4] : 4	enter element [4] : -1	enter element [4] : -1
1 999 2 999 3 999 4	1 -1 1 -1	-1 999 -1 999 -1 999 -1

##### 6. Вывод:

В ходе выполнения работы были изучены базовые алгоритмы языка C++.

Был проделан анализ работы, на основании которого была составлена блок-схема. На контрольных примерах мы убедились, что программа работает корректно.

Был оформлен комплект документации на программный код.