

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

Кафедра инженерной психологии и эргономики

Основы алгоритмизации и программирования
Отчет по лабораторной работе №10
«Алгоритмы сортировки и поиска»

Выполнил: Усов А.М.
Студент группы 310901
Преподаватель: Кабариха В. А.

Минск 2023

Цель: Сформировать знания и умения по изучению методов внутренних сортировок. Приобретение навыков реализации алгоритмов сортировок.

Задание 29. В одномерном массиве упорядочить отрицательные элементы, оставив положительные на местах. Сортировка включением.

Код программы:

```
#include <iostream>
```

```
#include <vector>
```

```
using namespace std;
```

```
// функция сортировки вставками(включением)
```

```
void InsertionSort(int * arr, int size) {
```

```
    for (int i = 1; i < size; i++) {
```

```
        // объявление опорного элемента
```

```
        int key = arr[i];
```

```
        int j = i - 1;
```

```
        //сдвиг всех элементов больше опорного на одну позицию вперед
```

```
        while (j >= 0 && arr[j] > key) {
```

```
            arr[j+1] = arr[j];
```

```
            j--;
```

```
        }
```

```
        // вставка опорного элемента на свое отсортированное место
```

```
        arr[j + 1] = key;
```

```
    }
```

```
}
```

```
int main() {
```

```
    // ввод размера массива
```

```
    cout << "enter a size of array: " << endl;
```

```
    int size;
```

```
    cin >> size;
```

```
    // создание массива и заполнение его элементами
```

```
    int *arr = new int[size];
```

```
    int * arrNegative = new int[size];
```

```
    cout << "enter a array: " << endl;
```

```
    for (int i = 0; i < size; i++) {
```

```
        cin >> arr[i];
```

```
    }
```

```
    // заполнение массива отрицательных чисел
```

```
    int m = 0;
```

```
    for (int i = 0; i < size; i++) {
```

```
        if (arr[i] < 0) {
```

```
            arrNegative[m] = arr[i];
```

```
            m++;
```

```
        }
```

```
    }
```

```

for (int i = m; i < size; i++) {
    arrNegative[i] = 0;
}

// сортировка массива отрицательных чисел
InsertionSort(arrNegative,size);

// замена отрицательных чисел в массиве на отсортированные
int j = 0;
for (int i = 0; i < size; i++) {
    if (arr[i] < 0) {
        arr[i] = arrNegative[j];
        j++;
    }
}

// вывод массива
for (int i = 0; i < size; i++) {
    cout << arr[i] << " ";
}

// освобождение памяти
delete [] arr;
delete [] arrNegative;

return 0;
}

```

Результат работы программы представлен на рисунке 1.

```
enter a size of array:
10
enter a array:
1 -2 3 -4 5 -6 7 -8 9 -10
1 -10 3 -8 5 -6 7 -4 9 -2
```

Рисунок 1 – Результат выполнения программы

Таким образом в результате выполнения лабораторной работы номер 10 «Алгоритмы сортировки и поиска» мы изучили базовые сортировки которые должен знать каждый программист, а так же дополнили этот список более сложными сортировками такие, как быстрая сортировка, сортировка слиянием и пирамидальная сортировка (кучей). На практическом примере разобрали в каких случаях более уместна та, или иная сортировка, что позволит нам в дальнейшем создавать более совершенные алгоритмы.