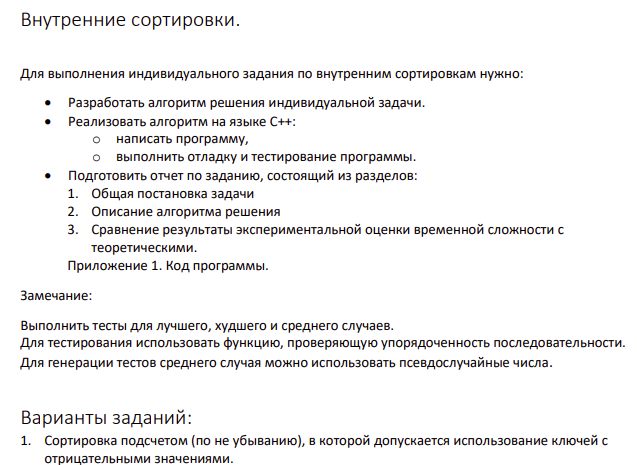
# **Практикум. Вариант 1**

# **Отчет по заданию «Сортировки» (сортировка подсчетом)**

Костин Андрей Константинович, группа 3530904/20001

1. **Общая постановка задачи**

****

1. **Описание алгоритма**

Алгоритм **сортировки подсчетом** заключается в **подсчёте количества элементов** каждого значения в массиве, **создании двух счетчиков** для положительных и отрицательных значений, **записи элементов в счетчики** и последующей **записи элементов в выходной массив** в порядке неубывания.

* 1. Находим **максимальное** и **минимальное** значения в массиве.
  2. Создаем **два счетчика**: для положительных значений от 0 до максимального значения, и для отрицательных значений от 0 до абсолютного значения минимального значения.
  3. Проходим по всем элементам массива и увеличиваем соответствующий счетчик на 1.
  4. Записываем элементы из **отрицательного** счетчика в выходной массив в порядке убывания (от максимального до минимального значения).
  5. Записываем элементы из **положительного** счетчика в выходной массив в порядке возрастания (от 0 до максимального значения).

1. **Сравнение результаты экспериментальной оценки временной сложности с теоретическими**

* Временная сложность построенного алгоритма - **O(n+k),** где n - длина массива, k - диапазон значений элементов в массиве (разница между максимальным и минимальным элементами).
* Временная сложность алгоритма соответствует теоретическим значениям для массивов с **любыми** значениями.
* **Лучший случай**: **длина массива совпадает с диапазоном значений** элементов в массиве, т.е. n = k. Временная сложность будет **O(n).**
* **Средний случай**: массив содержит **случайные значения**, но **не все элементы различны**. В этом случае при подсчете элементов в массиве некоторые значения будут иметь большее количество повторений и будут быстрее записаны в выходной массив. Однако все равно нужно будет пройти по обоим счетчикам и записать элементы в выходной массив. Временная сложность будет также **O(n+k)**.
* **Худший случай**: все элементы массива **различны** и находятся в **случайном порядке**. В этом случае при подсчете элементов в массиве все элементы будут иметь различные значения и будут записаны в два счетчика (положительных и отрицательных значений). Затем нужно будет пройти по обоим счетчикам и записать элементы в выходной массив. Временная сложность будет **O(n+k)**.