

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №4
по дисциплине «Алгоритмы и структуры данных»
Тема: Сортировки

Студент гр. 9303

Ефимов М.Ю

Преподаватель

Филатов Ар.Ю

Санкт-Петербург

2020

Цель работы.

Изучить алгоритмы сортировки, научиться оценивать эффективность различных сортировок на разных наборах входных данных, реализовать один из алгоритмов сортировки.

Задание.

Вариант 7. Циклическая сортировка. Использовать шаблоны.

Описание алгоритма

В алгоритме используется утверждение, что правильное место в массиве определяется количеством элементов которые меньше его по значению. Фактически мы можем определить правильное место в массиве посчитав элементы, которые меньше заданного. Циклическая сортировка ценна тем, что изменения среди элементов массива происходят только в случаях, когда элемент ставится на своё место. Это важно, если перезапись в массиве обходится слишком дорого и актуальны минимальные изменения физической памяти. При этом мы жертвуем скоростью выполнения. Скорость выполнения $O(n^2 / 2)$.

Выполнение работы.

Была реализована функция `Sort()` выполняющая циклическую сортировку. Функция шаблонная и может использоваться как для примитивных структур типа `int` или `char`, так и для пользовательских классов или структур, с условием что переопределены операторы сравнения. Считывание может происходить как с клавиатуры так и с консоли. Конечный результат и промежуточные значения выводятся на консоль и дублируются в файл. Пример входных данных и результат работы приведен в файлах `input.txt` и `result.txt`.

Выводы.

Была изучена циклическая сортировка, которая может быть использована на практике при программировании микроконтроллеров где очень важно количество перезаписываний, так как после определенного числа перезаписываний микроконтроллер приходит в нерабочее состояние. В ходе

работы была написана программа ,использующая циклическую сортировку для упорядочивания произвольного массива данных.

