Министерство науки иллюстрация высшего образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**+**

**ОТЧЕТ**

по лабораторной работе №2

по курсу “Логика и основа алгоритмизации в ИЗ”

на тему “Оценка времени выполнения программ”

Выполнили студенты группы 21ВВ3:

Савосин Владислав

Юсеев Руслан

Приняли:

Митрохин М.А., Юрова О.В.

Пенза 2022

**Название**

Оценка времени выполнения программ

**Цель работы:** Оценить время выполнения программы, выполняя лабораторные указания 1-2.

**Лабораторное задание:**

Задание 1:

1. Вычислить порядок сложности программы (*О*-символику).
2. Оценить время выполнения программы и кода, выполняющего перемножение матриц, используя функции библиотеки time.h для матриц размерами от 20, 50, 75, 100, 150, 175, 200.
3. Построить график зависимости времени выполнения программы от размера матриц и сравнить полученный результат с теоретической оценкой.

Задание 2:

1. Оценить время работы каждого из реализованных алгоритмов на случайном наборе значений массива.
2. Оценить время работы каждого из реализованных алгоритмов на массиве, представляющем собой возрастающую последовательность чисел.
3. Оценить время работы каждого из реализованных алгоритмов на массиве, представляющем собой убывающую последовательность чисел.
4. Оценить время работы каждого из реализованных алгоритмов на массиве, одна половина которого представляет собой возрастающую последовательность чисел, а вторая, – убывающую.
5. Оценить время работы стандартной функции qsort, реализующей алгоритм быстрой сортировки на выше указанных наборах данных.

**Задание 1:**

1. Рассмотрев программу мы вычислили ее теоретическую сложность программы О(n3).
2. Мы оценили время выполнения программы и кода,выполняющего перемножение матриц, используя функции библиотеки time.h для матриц размерами от 50, 100, 150, 200, 300, 400, 600. Рис 1-7.

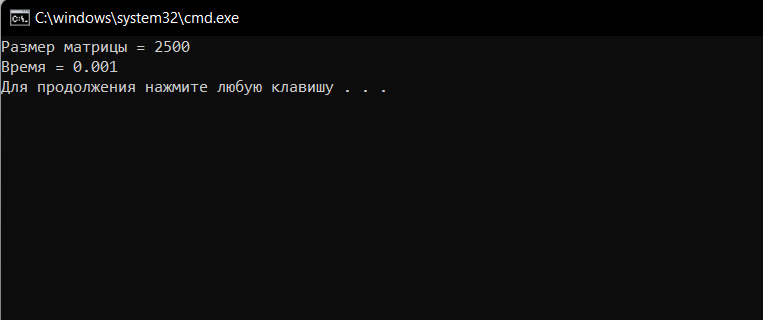


Рис 1

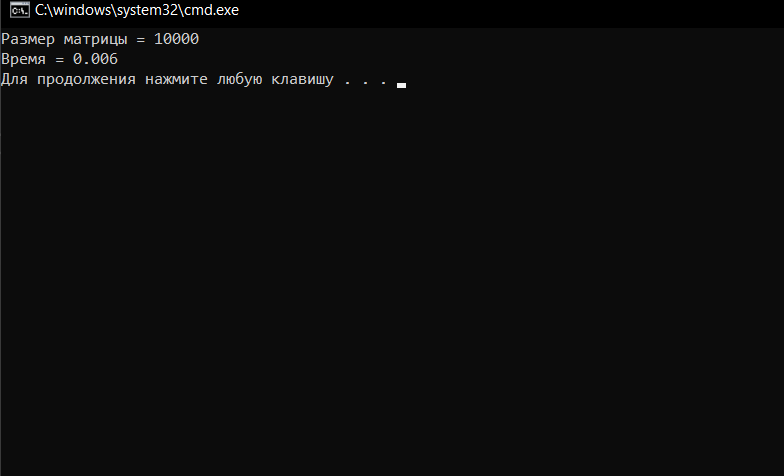


Рис 2

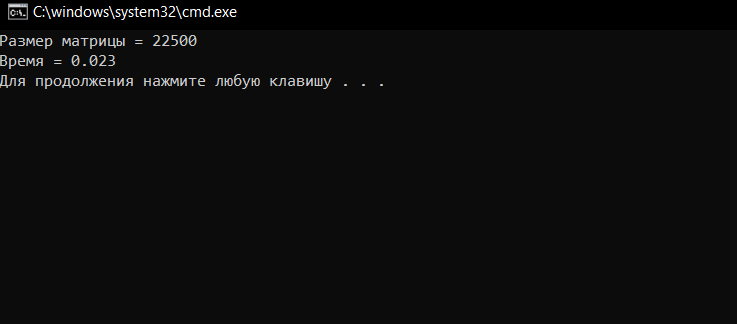


Рис 3

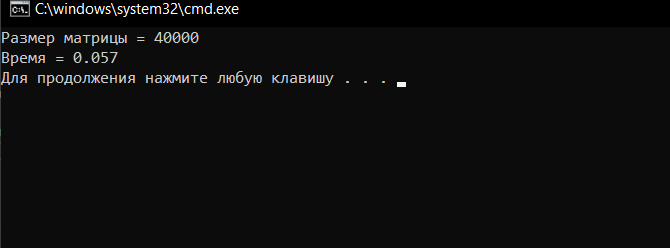


Рис 4



Рис 5

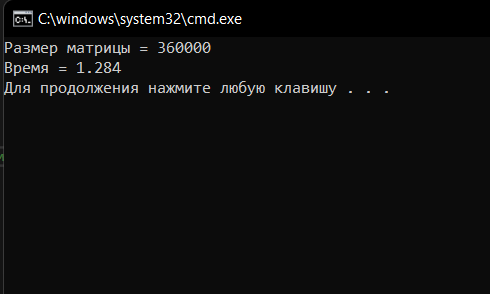


Рис 6

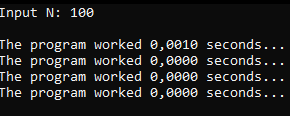
3. Построили график зависимости времени выполнения программы от размера матриц и сравнили полученный результат с теоретической оценкой.

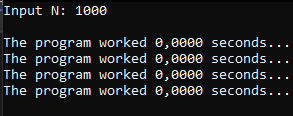
Мы научились вычислять сложность программы. При заданной сложности программы мы увеличивали количество её элементов тем самым получив зависимость времени от количества элементов.

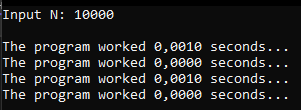
**Задание 2:**

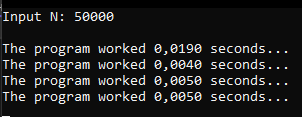
В скриншотах время работы сортировки записываются в последовательности: Случайный массив ,Возрастающий массив ,Убывающий массив ,Возрастающе-убывающий массив

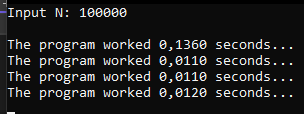
Shell:

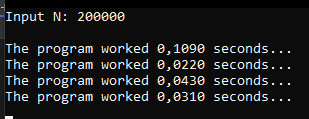


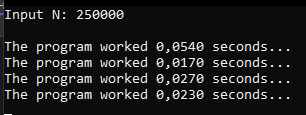






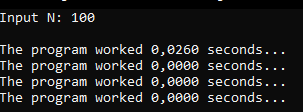


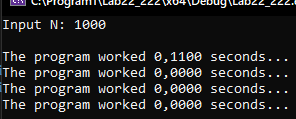


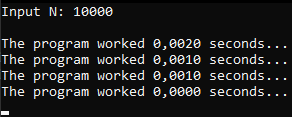


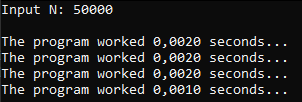
Время выполнения программы увеличивается прямо пропорционально увеличению размера массива. Shell одинаково хорошо справляется со всеми указанными выше наборами данных.

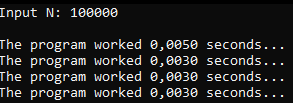
Qs

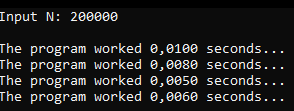


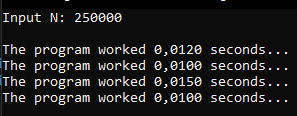






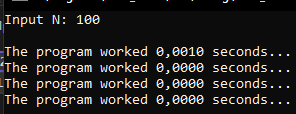


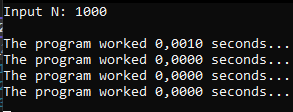


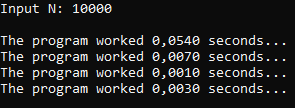


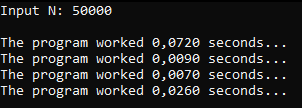
Время выполнения программы увеличивается прямо пропорционально увеличению размера массива. Qs одинаково хорошо справляется со всеми указанными выше наборами данных.

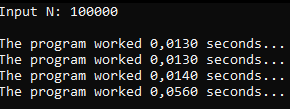
Qsort

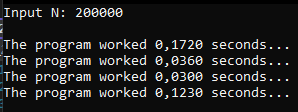


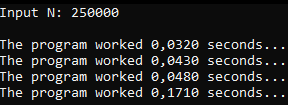












Время выполнения программы увеличивается прямо пропорционально увеличению размера массива. qsort одинаково хорошо справляется со всеми указанными выше наборами данных.

**Вывод:** мы выполнили лабораторные указания 1-2. Оценили и сравнили между собой результаты выполнения программ. Рассмотрев программу мы вычислили ее теоретическую сложность программы О(n3). График экспериментально подтвердил наш ответ. При измерении времени разных сортировок, а именно сортировка Shell, Qs, qsort, нами было выявлена наилучшая сортировка чисел – Qs.