Министерство науки и высшего образования Российской Федерации  
Пензенский государственный университет  
Кафедра вычислительная техника

**ОТЧËТ**  
по лабораторной работе №2  
по дисциплине: «Оценка времени выполнения программ»

Выполнили студенты группы 22ВВВ2:  
Беляев Д.

Приняли:  
Акифьев И. В.

Митрохин М. А.

Пенза 2023

**Название**

Оценка времени выполнения программ

**Цель работы**

Научиться измерять время работы и сложность кода.

**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 100, 200, 400, 1000, 2000, 4000, 10000.
3. Построить график зависимости времени выполнения программы от размера матриц и сравнить полученный результат с теоретической оценкой.

Задание 2:

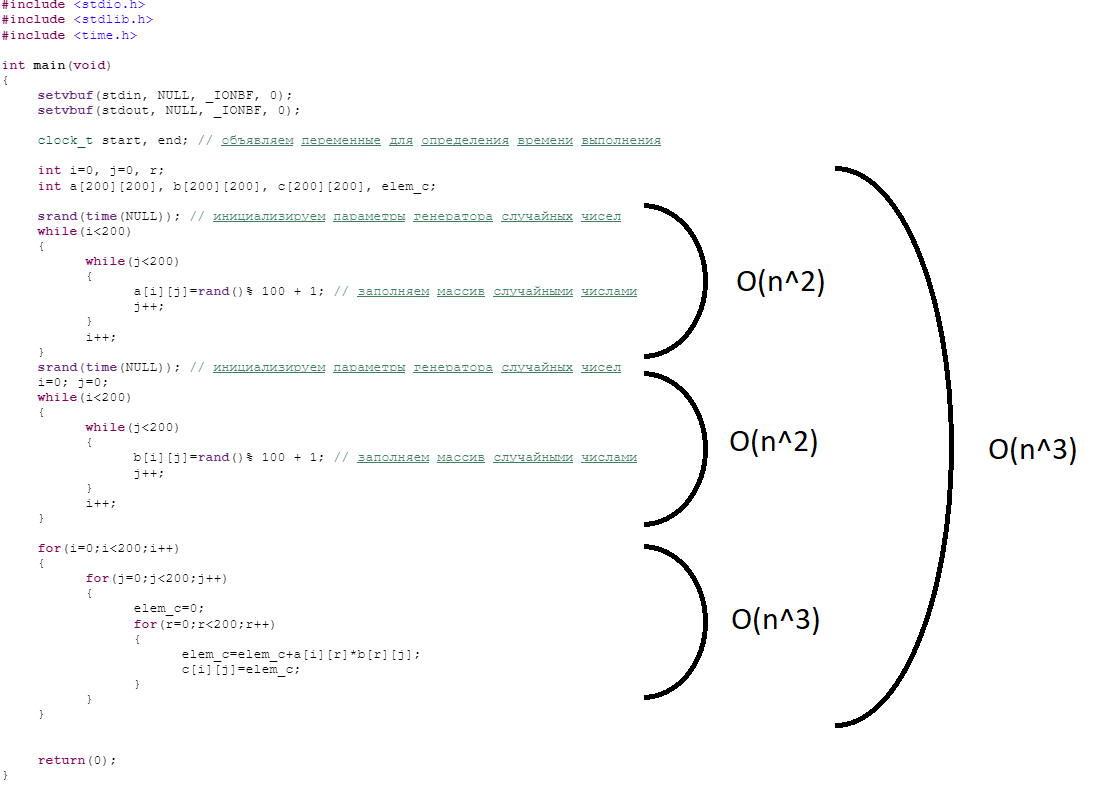
1. Оценить время работы каждого из реализованных алгоритмов на случайном наборе значений массива.
2. Оценить время работы каждого из реализованных алгоритмов на массиве, представляющем собой возрастающую последовательность чисел.
3. Оценить время работы каждого из реализованных алгоритмов на массиве, 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.

**Листинг**

**Задания**

Задание 1

1. Сложность программы: O(n^3)



1. Измерение время выполнения работы (Release)

Матрица размером 100х100: 0.000000sec

Матрица размером 200х200: 0.006000sec

Матрица размером 400х400: 0.045000sec

Матрица размером 1000х1000: 0.830000sec

Матрица размером 2000х2000: 11.241000sec

Матрица размером 4000х4000: 220.994000sec

Матрица размером 10000х10000: n/a

1. График зависимости

Задание 2 (100000 чисел):

1. Случайный набор чисел

Shell sort: 0.515000sec

Quick sort: 0.005000sec

1. Возрастающая последовательность чисел

Shell sort: 0.001000sec

Quick sort: 0.002000sec

1. Убывающая последовательность чисел

Shell sort: 1.007000sec

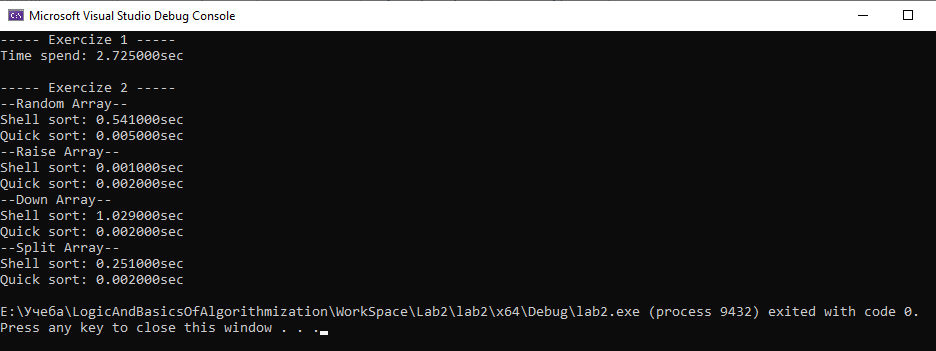
Quick sort: 0.002000sec

1. Половина - возрастающая последовательность чисел, половина – убывающая

Shell sort: 0.250000sec

Quick sort: 0.002000sec

**Результат работы программы**

****

**Вывод**

Я научился измерять время работы и сложность кода.