ГУАП

КАФЕДРА № 53

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| старший преподаватель |  |  |  | Ушаков В.А. |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №7 |
| Оценка сложности алгоритмов |
| Вариант 3 |
| по курсу: Основы программирования |
|  |

РАБОТУ ВЫПОЛНИЛ(А)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 5138 |  |  |  | Воробьев В.А. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2022

**Задание:** реализовать алгоритм на языке C/С++, выполняющий поставленную задачу. Глобальные параметры использовать запрещено; допустимо использование дополнительных функций. Разработанный алгоритм должен быть реализован в виде цельной программной функции (или нескольких функций) так, чтобы мог быть многократно применением с различными исходными данными и при этом не включал команды, не относящиеся к решаемой задаче, например, ввод и вывод исходных данных на консоль или в файл. Произвести теоретическую оценку количества используемых операций разработанного алгоритма. Произвести экспериментальную проверку времени работы разработанного алгоритма, определив его класс сложности для среднего случая. Измерить среднее время для нескольких тестов повторений при различных размерностях входных данных.

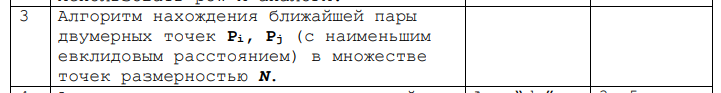


Рисунок - вариант задания

**Выполнение задания:**

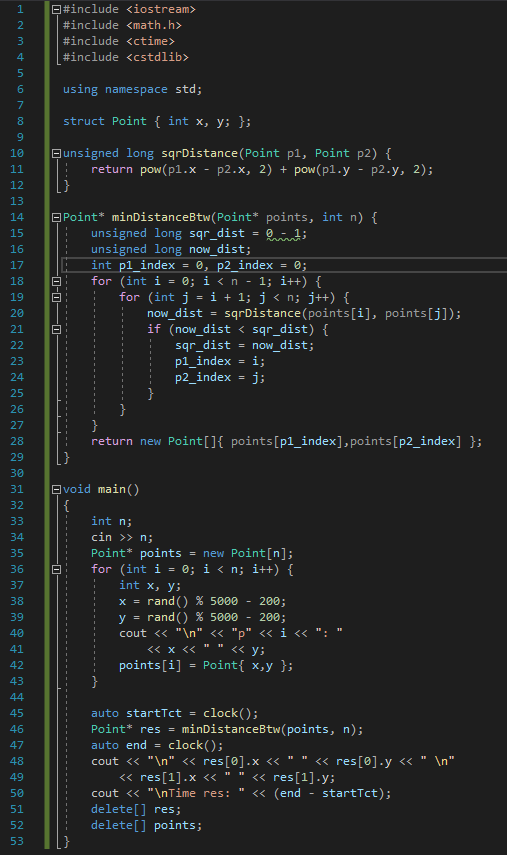


Рисунок 2 – код

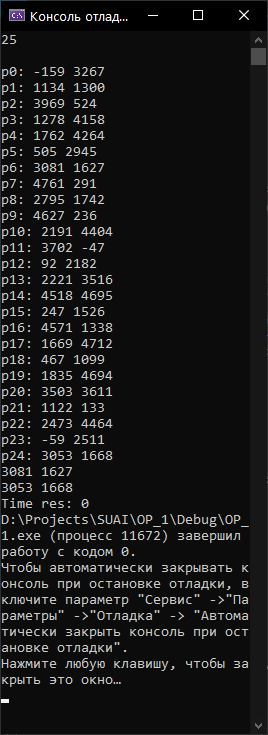


Рисунок 3 - ввод и вывод



Рисунок 4 - Блок-схема

**Анализ сложности алгоритма:**

Сделаем замеры скорости выполнения для N = 80; 150; 250; 500; 1000; 2500.

Таблица 1 – измерения tacts(N)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| N | 80 | 150 | 250 | 500 | 1000 | 2500 | 5000 |
| tacts | 1 | 2 | 4 | 16 | 58 | 380 | 1480 |

Построим график и оценим сложность реализованного алгоритма.

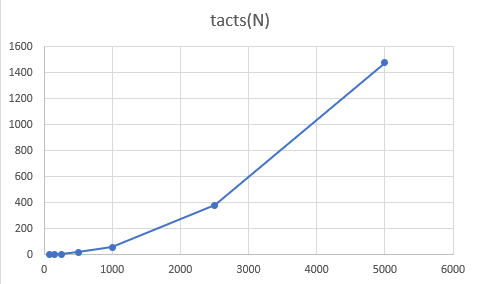


Рисунок 5 – такты от числа входных данных

**Вывод:** текущий алгоритм имеет временную сложность O(N2), это понятно из графика (рис.5), которые стремительно растет с увеличением количества входных данных. В ходе выполнения задания был выполнен алгоритм, на основе которого мы научились измерять временную сложность алгоритма. Также освоили метод «грубой силы».