ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчёт по лабораторной работе № 4

«Поиск подмассивов размера K в массиве размера N, сумма элементов которых равна нулю»

Выполнил работу

Степанов Виктор

Академическая группа №j3110

Принято

ментор, Вершинин Владислав Константинович

Санкт-Петербург

2024

**Введение.**

1. Цель работы: понять минусы неоптимизированного кода.

Задача: найти подмассивы размера K, сумма элементов которых равна нулю.

1. В коде используется алгоритм перебора массива. Используется тип данных int.
2. Реализация кода довольно проста, заносим данные в вектор, который перебирается семью циклами.
3. Экспериментальная часть

В этом разделе вам необходимо привести результаты работы вашего алгоритма, с таблицами и графиками, демонстрирующими выполнения алгоритма с различными условиями и наборами данных. Оценивается производительность и сравниваются результаты с теоретическими оценками.

Выделяемая память на один проход: (n \* 4)(для исходного массива размера n) + 28(для переменных цикла) байт + 8(две переменные)байт.

Выделенная память в наихудшем случае при n=25, проходов соответственно будет 25^7, а память = 25^7 \* 128 + 8 байт.

Асимптотика = O(N^7)(тк вложенные циклы 7 раз)

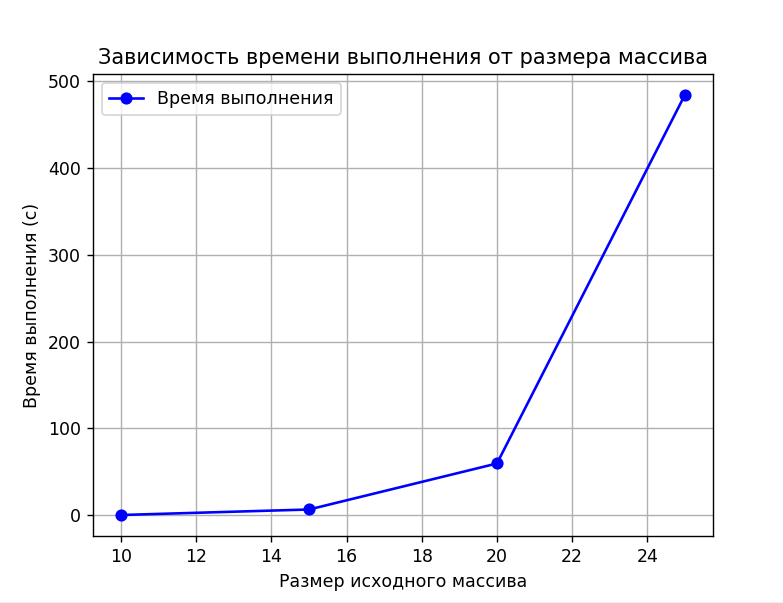
График зависимости времени от числа элементов. Пример выполнения:

Требования моего варианта, размер подмассива k=7, размер исходного массива N до 25. Сложность не менее О(N^7). Для тестирования алгоритма была собрана статистика, приведенная в таблице №1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Размер исходного массива | 10 | 15 | 20 | 25 |
| Время выполнения программы(с) | 0.1 | 6.45 | 59.5 | 484.4 |
| О(N^k) | О(10^7) | О(15^7) | О(20^7) | О(25^7) |

Таблица №1 - Подсчёт сложности реализованного алгоритма.

График представляющий визуально удобный формат данных из таблицы №1 представлен на изображении №1.



Изображение №1 - График работы алгоритма

До 15 элементов программа ведет себя стабильно, с 15 до 20 несильно увеличивается время выполнения, что вполне ожидаемо. С 20 до 25 элементов происходит скачок в плане объема работы и времени ее выполнения соответственно.

1. Заключение

В процессе выполнения работы мною был реализован алгоритм полного перебора исходного массива. Цель была достигнута: неоптимизированный алгоритм может выполняться огромное кол-во времени. Полученные результаты совпадают с теоретическими оценками сложности алгоритма.

ПРИЛОЖЕНИЯ

