Московский Авиационный Институт

(Национальный Исследовательский Университет)

Институт № 8 информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

**Лабораторная работа №8 по курсу**

**«Дискретный анализ»**

**Жадный алгоритм**

Студент: Пермяков Никита Александрович

Группа: М80 – 208Б-19

Вариант: 2

Преподаватель: [*Кухтичев Антон Алексеевич*](https://mai.ru/education/schedule/ppc.php?guid=a3f854e1-f771-11e7-ae95-485b3919ee6d)

Оценка: \_\_\_\_\_\_\_\_\_\_\_

Дата: \_\_\_\_\_\_\_\_\_\_\_

Подпись: \_\_\_\_\_\_\_\_\_\_\_

Москва, 2020

**Содержание**

1. Постановка задачи
2. Метод и алгоритм решения
3. Описание программы
4. Дневник отладки
5. Тестирование производительности
6. Вывод

**Постановка задачи**

**Задача**: Разработать жадный алгоритм решения задачи, определяемой своим вариантом. Доказать его корректность, оценить скорость и объем затрачиваемой оперативной памяти.

Реализовать программу на языке С или С++, соответствующую построенному алгоритму. Формат входных и выходных данных описан в варианте задания.

На координатной прямой даны несколько отрезков с координатами [Li, Ri]. Необходимо выбрать минимальное количество отрезков, которые бы полностью покрыли интервал [0, M].

**Метод и алгоритм решения**

Структура данных - вектор, в который добавляем такой отрезок при итерации по всем отрезкам, левый конец которого принадлежит уже покрытой области, а правый максимален. Итерируемся до тех пор, пока отрезки не покроют всю область.

**Описание программы**

struct TSection - структура хранит левую, правую границу и id.

void SelectorSection - выбор отрезка и вставка в результатирующий вектор

bool Compare() - сравнение отрезков по id

**Дневник отладки**

1-2)Понимание формата входных данных

3-5) Написание вывода отдельно от сравнения - дублирование при начальном состоянии.

6-9) Оптимизация

10) Рефакторинг, вынос за циклы, вызов функций один раз, рассинхронизация потока вывода.

**Тестирование производительности**

генератор: gen\_tests.py

Формат входных данных

В первой строке задано 1 ≤ n ≤ 100, 1000, 10000, 100000, 1000000.

В последующих n строках через пробел заданы пара случайных чисел Li, Ri, до 1000.

В последней строке случайное число 1 ≤ m ≤ 1000.

**Таблица 1.** Зависимость величины тестовых данных от времени работы жадного алгоритма в задаче наибольшего покрытия отрезков.

|  |  |
| --- | --- |
| кол-во отрезков | Жадный алгоритм (ms) |
| 100 |  |
| 1000 |  |
| 10000 |  |
| 100000 |  |
| 1000000 |  |

**Вывод**

Алгоритм зависит от входных данных. Сложность меняется от О(n) до О(n2).