# Лабораторная работа №2 по курсу «Алгоритмы и структуры данных»

## Описание алгоритмов

### Алгоритм перебора

**Подготовка:** О(1)

**Поиск:** О(N), где N – количество прямоугольников

**Описание:** Для каждой точки происходит полный перебор всех прямоугольников и проверяется ее принадлежность ко всем прямоугольникам

### Алгоритм на карте

**Подготовка:** О(N^3)

**Поиск:** О(logN), где N – количество прямоугольников

**Описание:** По сжатым координатам прямоугольников строим карту, и для каждой точки проверяем какое число будет на карте по ее сжатым координатам. Оно и будет являться ответом.

### Алгоритм на дереве

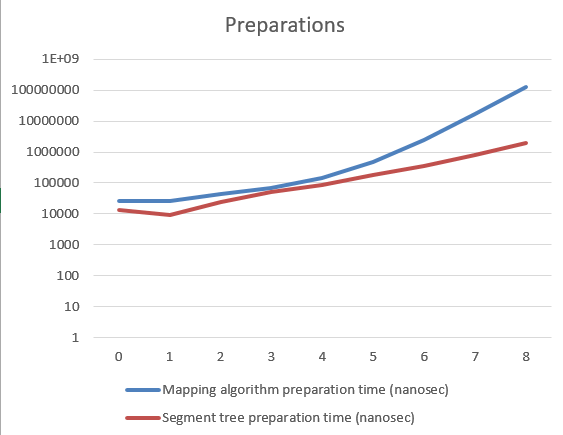
**Подготовка:** О(N \* logN)

**Поиск:** О(logN), где N – количество прямоугольников

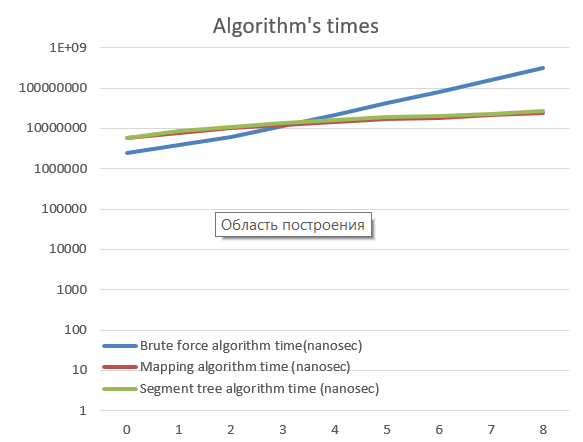
**Описание:** По сжатым координатам оси Y строим персистентное дерево. В массиве корней дерева находим нужный по сжатой координате для X и в дереве этого корня спуститься до листа сжатого Y, попутно собирая модификаторы (считая прямоугольники).

## Тестирование и вывод

**Сравнение времени подготовки:** С ростом количества входных данных растет и время подготовки карты и дерева для последних двух алгоритмов. Построение персистентного дерева отрезков будет немного быстрее, чем карты, но чем больше данных, тем больше разница во времени с выигрышем у первого.



**Сравнение поиска ответа:** На небольших данных алгоритм грубой силы работает немного быстрее, но при их увеличении он начинает проигрывать алгоритмам с картой и деревом. А последние два практически не отличаются во времени поиска ответа для точки.



## Итог

Алгоритм грубой силы неплохо показывает себя на небольших данных (мало прямоугольников и мало точек), и в таких случаях действительно лучшим вариантом будет использовать именно его.

Алгоритм на карте быстро работает при любом количестве точек, но лучше всего его использовать при маленьком количестве прямоугольников, так как он проигрывает дереву на больших данных.

Алгоритм на дереве работает быстро при любых данных, но он немного сложен в написании.