

1 2.2. Заявки на переговоры

В большой *IT*-фирме есть только одна переговорная комната. Желаящие посоветоваться заполняют заявки с желаемым временем начала и конца. Ваша задача определить максимальное количество заявок, которое может быть удовлетворено. Число заявок ≤ 100000 .

2 Описание алгоритма

- Сортируем массив запросов (массив пар начало - конец) по времени окончания
- Проходимся по отсортированному массиву по следующему принципу:
 - Изначально счетчик равен единице
 - Храним пару, с которой сравниваем элементы массива(текущую)
 - Если встречаем элемент массива, начало которого \geq конца текущего, то инкрементируем счетчик на единицу и обновляем текущий элемент на данный.

3 Доказательство корректности работы

Отсортировав все запросы по правому краю, можно заключить следующее утверждение: между двумя различными смежными окончаниями заседаний может быть не более одного заседания, которое можно удовлетворить. Действительно, если таких заседаний более одного, то существует заседание, конец которого находится между ними. Таким образом, мы, проходясь по массиву, ищем любой запрос, который может быть удовлетворен, так как больше одного мы удовлетворить не можем.

4 Время работы и доп. память

- Время работы $O(n \log(n))$, где n - кол-во запросов.
- Доп. память $O(n)$.

5 Доказательство времени работы

Время работы равно сумме работ сортировки $O(n \log(n))$ и прохода по массиву $O(n) \implies$ суммарное время работы $O(n \log(n))$. Доп память $O(n)$ для хранения запросов. В алгоритме используется стандартная сортировка `std :: sort`, время ее работы смотри по [ссылке](#) .