**Алгоритмы поиска**

**ОБЩИЙ БАЛЛ 25**

1.

Вопрос 1

Для множества вызван метод equal\_range:

set<string> s = { /\* какие-то строки \*/ };

auto range = s.equal\_range("some string");

Какие утверждения про переменную range *могут быть* верны?

7 баллов



next(range.first) == prev(range.second)



range.first == range.second



next(range.first) == range.second



prev(range.first) == next(range.second)



prev(range.first) == range.second



range.first == prev(range.second)



range.first == next(range.second)

2.

Вопрос 2

Рассмотрим функцию FindSegment:

template <typename RandomIt>

pair<RandomIt, RandomIt> FindSegment(

    RandomIt range\_begin, RandomIt range\_end, int left, int right) {

  // ТЕЛО ФУНКЦИИ

}

Каким должно быть тело функции, чтобы для отсортированного диапазона целых чисел [range\_begin, range\_end) она возвращала границы диапазона (полуинтервала) элементов, больших или равных left и меньших или равных right (то есть принадлежащих отрезку [left, right])?



return {upper\_bound(range\_begin, range\_end, left),

        lower\_bound(range\_begin, range\_end, right)};



return {upper\_bound(range\_begin, range\_end, left),

        upper\_bound(range\_begin, range\_end, right)};



return {upper\_bound(range\_begin, range\_end, left - 1),

        lower\_bound(range\_begin, range\_end, right + 1)};



return {lower\_bound(range\_begin, range\_end, left),

        lower\_bound(range\_begin, range\_end, right)};



return {lower\_bound(range\_begin, range\_end, left),

        upper\_bound(range\_begin, range\_end, right)};



return {next(lower\_bound(range\_begin, range\_end, left - 1)),

        prev(upper\_bound(range\_begin, range\_end, right + 1))};