



В очень большом мегаполисе решили запустить сеть автономных такси без водителей. Планируется открыть 10 станций аренды.

**Задача**: определить, где их разместить, чтобы охватить максимум клиентов.

#### Условия:

- Клиенты могут арендовать машину на любой станции в радиусе R км включительно. Если станция дальше расстояния R км — клиент не поедет.
- Станции будут строить только в точках возможного расположения клиентов.

#### Требования к решению и отправке тестового задания

- Основной способ запуска программы классический метод main.
- Метод main должен поддерживать несколько вызовов подряд.
- Алгоритм решения должен быть описан в JavaDoc основного класса, в котором находится метод main.
- В случае ошибки программа должна бросить исключение

Файл с решением разместите в облачном хранилище — например, Яндекс.Диск, и прикрепите ссылку к анкете на стажировку. Не забудьте проверить доступ по ссылке.

Пожалуйста, не размещайте решение в открытых источниках. Его могут скопировать и выдать за свое — так отбор потеряет объективность.

### Ограничения для алгоритма

- Ограничение по времени: 2 секунды.
- Ограничение по памяти: 16 мегабайт.

# NAUMENI

#### Формат входных данных

- 1. Имя входного файла: input.txt.
- 2. Кодировка файла UTF-8.
- 3. Файл должен находится в рабочем каталоге программы, без указания пути до файла, т.е. просто **new File("input.txt")**.
- 4. В первой строке записано 2 числа через пробел:
  - количество клиентов  $1 \le N \le (2^{31}-1)$ ,
  - значение параметра  $0 < R \le (1.7*10^{100})$ .
- 5. Начиная со второй строки следует ровно N строк, в которых написано два числа координаты (-1.7\*10¹⁰°) < Xi,Yi ≤ (1.7\*10¹⁰°) і-го клиента. Нумерация ведется с нуля.

### Формат выходных данных

- 1. Имя выходного файла: output.txt.
- 2. Кодировка файла UTF-8.
- 3. Файл должен находится в рабочем каталоге программы, без указания пути до файла, т.е. просто **new File("output.txt")**.
- 4. В выходном файле должно присутствовать не более 10 строк. Каждая строка должна содержать номер клиента, у которого выгоднее всего строить станцию, и количество обслуживаемых этой станцией клиентов, отличное от нуля. Выводимые строки должны быть упорядочены от наибольшего количества обслуживаемых клиентов к наименьшему. Если две и более станции могут обслужить одинаковое число клиентов, то выше в списке должна находиться станция с меньшим номером.

#### Критерии оценки тестового задания

- Качество кода: применение ООП и принципов SOLID.
- Скорость алгоритма.
- Потребление памяти.
- Обработка ошибок.
- Наличие и качество тестов.

# NAUMEN

## Примеры

input.txt	output.txt
5 3	0 2
0 0	1 1
2 -2	2 1
5 3	3 1
_ 7 7	1 1

input.txt		Ol	ıtpı	ut.t	xt
10 3.000	900	5	4		
3.168070	1.752490	1	3		
0.500730	6.436580	4	3		
0.089300	0.112720	6	3		
2.275440	7.508780	9	3		
0.779230	4.377090	0	2		
0.644400	1.381650	2	2		
1.844920	1.430420	3	2		
8.079870	5.225030	7	1		
7.823270	5.317290	8	1		
1.788400	5.426120				

Файлы с решением размести на облачном хранилище (например, Яндекс.Диск) и прикрепи ссылку к анкете на стажировку.

Перед отправкой проверь доступ по ссылке и не размещай файлы в открытом доступе.

Желаем удачи!

