

Задание 1. Тестовые данные. 40 баллов

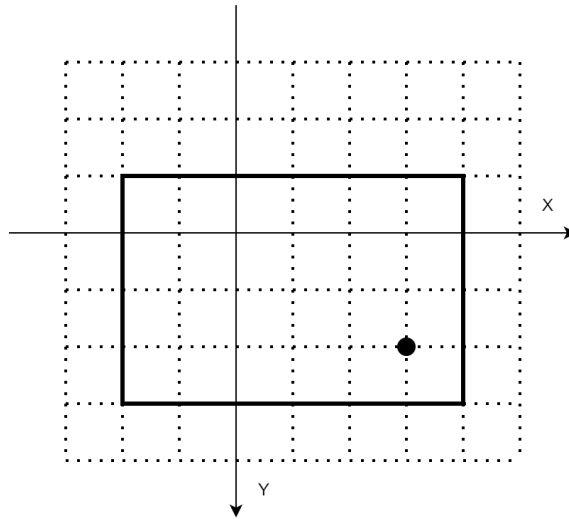
Важно! В этой задаче не нужно писать программу.

Составьте набор входных и выходных данных, позволяющих протестировать работу программы, определяющей положение точки относительно прямоугольника. Программу писать в этом задании не нужно.

Вам дана эталонная (правильная) программа **PointAndRect.exe**, которая выводит положение точки относительно прямоугольника.

Формат входных данных программы PointAndRect.exe

Программа считывает в окне консоли 6 целых чисел, разделенных пробелами, табуляциями или символами перевода строки: координаты X , Y левого верхнего угла прямоугольника, его ширину и высоту, а также координаты X и Y точки. Координатная ось X направлена вправо, а ось Y – вниз.



Последовательность чисел -2 -1 6 4 3 2 задаёт прямоугольник размерами 6x4 с координатами верхнего левого угла (-2, -1) и точку с координатами (3, 2)

Формат выходных данных

После считывания данных правильная программа должна вывести в консоль одну из строк:

- **TOP-LEFT**, если точка находится в верхнем левом углу прямоугольника;
- **BOTTOM-LEFT**, если точка находится в нижнем левом углу прямоугольника;
- **TOP-RIGHT**, если точка находится в верхнем правом углу прямоугольника;
- **BOTTOM-RIGHT**, если точка находится в нижнем правом углу прямоугольника;

- **LEFT**, если точка находится на левой стороне прямоугольника;
- **TOP**, если точка находится на верхней стороне прямоугольника;
- **RIGHT**, если точка лежит на правой стороне прямоугольника;
- **BOTTOM**, если точка лежит на нижней стороне прямоугольника;
- **OUTSIDE**, если точка лежит вне прямоугольника;
- **INSIDE**, если точка лежит внутри прямоугольника;
- **ERROR**, если при считывании целых чисел произошла ошибка (например, введено не число);
- **ERROR**, если ширина или высота прямоугольника не натуральное число (от 1 и больше).

Примеры входных и выходных данных

Ввод	Вывод	Примечание
-2 -1 6 4 3 2	INSIDE	Точка (3, 2) лежит внутри прямоугольника (-2, -1, 6, 4). Вводимые числа могут располагаться на разных строках.
-2 -1 6 4 -2 -1	TOP-LEFT	Точка (-2, -1) лежит в верхнем левом углу прямоугольника (-2, -1, 6, 4)
1 abc 2 3 4 5	ERROR	В качестве координаты Y верхнего левого угла прямоугольника было введено не число (abc)

Указания

Изучите как работает данная вам **правильная программа** PointAndRect.exe. Запустите её и посмотрите какие данные она выведет в ответ.

Предоставьте папку с набором файлов с именами **input<N>.txt** и **output<N>.txt**, где <N> – некоторое натуральное число. Файлы **input<N>.txt** содержат входные данные, а **output<N>.txt** ожидаемые выходные данные.

Например, в файле *output3.txt* запишите данные, которые правильная программа должна вывести, если ей подать на вход данные из файла *input3.txt*.

Созданные вами файлы будут использованы для проверки работы правильных и неправильных версий программы PointAndRect.exe.

Правильная программа – та, которая выдаёт результат в соответствии с требованиями.

Неправильная программа – та, которая содержит выдаёт результат, отличный от ожидаемого.

Набор подготовленных вами файлов должен быть таким, чтобы:

- **все правильные программы** для каждого *input<N>.txt* вывели результат, совпадающий с *output<N>.txt*;

- **каждая из неправильных программ** для данных из хотя бы одного из входных input<N>.txt вывела результат, не совпадающий с output<N>.txt.

Рекомендации

Разместите ваши файлы со входными и выходными данными в папке tests, давая им имена:

input1.txt и output1.txt, input2.txt и output2.txt, и так далее, пока не решите, что написали достаточно тестовых данных.

В файлах input*.txt разместите входные данные, а в соответствующих им output*.txt - данные, которые правильно работающая программа должна вывести при этих входных данных.

Чтобы проверить, действительно ли ваша программа ведёт себя верно на тестовых данных, запустите в консоли файл test.bat с параметром, задающим номер теста.

Например:

```
test.bat 1
```

Проверит, что программа при передаче ей данных из tests\input1.txt выведет текст, равный содержимому файла tests\output2.txt.

Если выведется ОК, значит всё хорошо. В противном случае будет выведено сообщение об ошибке.

Запуск test.bat без указания номера теста, равнозначен запуску теста №1.

Задание 3. 80 баллов

На далекой планете Шахмантия очень почитают игру в Шахматы. Все ее жители, шахмантианцы, знают правила этой игры наизусть и могут посоревноваться с лучшими игроками соседних планет.

Юный шахмантианец Вася хочет сдать экзамен по шахматам, тем самым доказать, что готов к соревнованиям с игроками ближайших планет.

Экзамен проводится следующим образом: даётся список, состоящий из позиций белых фигур трех типов: слон, ферзь, конь. Нужно нарисовать поле с этими фигурами и отметить на этом поле возможные места для постановки черного короля (его не должны бить).

Помогите Васе сдать экзамен, чтобы он прославился не только в Шахмантии, но и во всей Галактике. Для этого напишите программу, рисующую поле с белыми фигурами и местами где можно поставить черного короля.

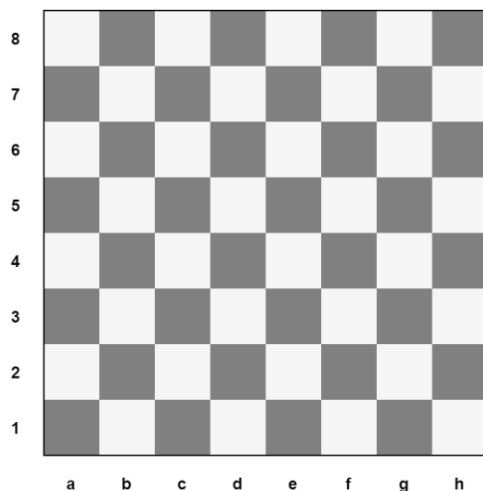
Формат входных данных

На вход программе построчно вводятся через пробел координата и фигура. Координата записана в виде буквы и цифры (слитно. Например, "a3 B").

Обозначения фигур:

- K – конь (knight)
- Q – ферзь (queen)
- B – слон (bishop)

Номер столбца задаётся при помощи строчных букв английского алфавита a..h, а номер ряда – цифрами от 1 до 8.



Координатная система шахматной доски

Например, так указывается белый ферзь в позиции A3:

a3 Q

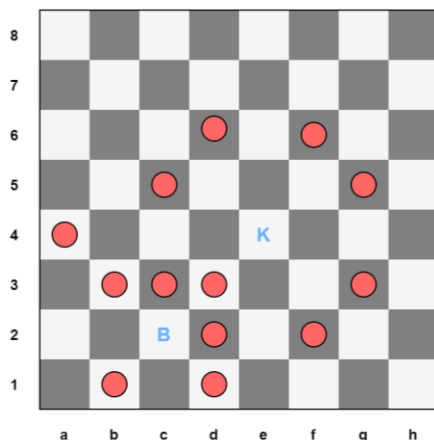
Фигуры на вход программы могут подаваться в произвольном порядке, количество фигур каждого типа может быть произвольным. Гарантируется, что на вход программе подаются корректные координаты фигур, и в одной клетке находится не более одной белой фигуры.

Формат выходных данных

Программа должна вывести шахматное поле 8x8 клеток, где клетка, на которой может стоять черный король отображается символом "#", а фигуры обозначаются буквами "K", "Q", "B" (K - конь, Q - ферзь, B - слон). Битые клетки отображаются символом "." (точка).

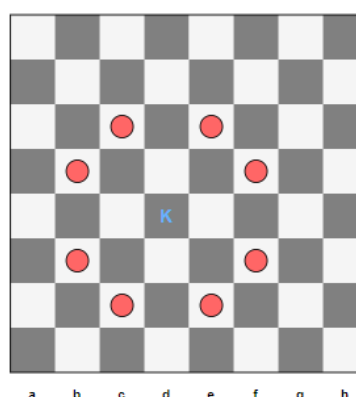
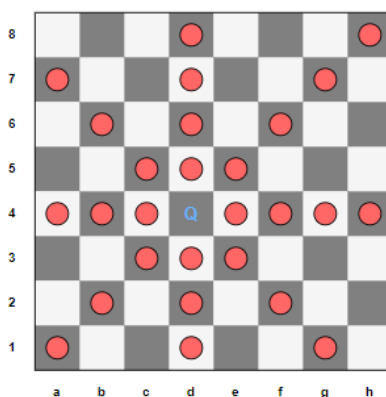
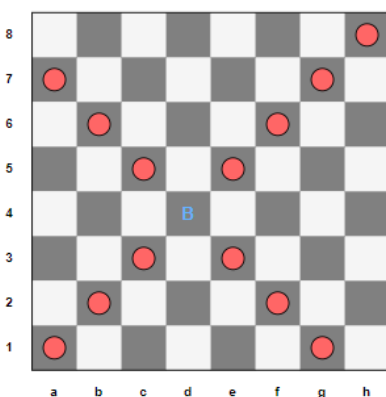
Никаких других символов выводиться не должно.

Учтите, что атака слона или ферзя прекращается, когда на его пути встречается другая фигура, а конь может перепрыгивать через другие фигуры:



Слон на c2 не бьет клетки f5, g6 и h7, так как их закрывает конь e4

Как ходят фигуры



Конь (K) ходит буквой `Г`, Слон (B) ходит только по диагоналям, а Ферзь (Q) – по диагоналям, горизонтальной и вертикальной линиям.

Пример ввода-вывода:

Ввод	Вывод	Примечание
------	-------	------------

<p>a1 K b7 Q e5 B</p>	<pre> ...####. .Q.....#.# #.#.B### #.#...## #..##...# #..####. K.##### </pre>	
-------------------------------	--	--