

### Задача №1. Двоичный калькулятор. (100 баллов)

У вас есть очень простой калькулятор с дисплеем и двумя кнопками: "-2" и "+3". При нажатии на кнопку соответствующая операция применяется к числу на дисплее. Например, если на дисплее число 10, то после нажатия на кнопку "-2" на дисплее будет число 8, а при нажатии на кнопку "+3" - число 13.

В начальный момент времени на дисплее число  $A$ . Определите минимальное количество нажатий на кнопки, которое преобразует это число в  $B$ .

*Входные данные.*

В стандартном входном потоке данных (*чтение с клавиатуры*) в первой строке через пробел записаны два числа  $A$  и  $B$ .

*Выходные данные.*

В стандартный выходной поток данных (*запись на экран*) ваша программа должна вывести одно число - минимальное количество нажатий на кнопки калькулятора, преобразующее число  $A$  в число  $B$ .

*Ограничения.*

Время выполнения программы не должно превышать 1 секунды для каждого теста.

$$1 \leq A \leq 100$$

$$1 \leq B \leq 100$$

*Примеры работы программы.*

<b>Вход</b>	10	23	18
	14	23	12
<b>Выход</b>	3	0	3

### Задача №2. Черно-белый пасьянс. (100 баллов)

Вы раскладываете пасьянс из  $N$  карт, которые с одной стороны белые, а с другой - черные. Изначально все карты лежат перед вами в ряд произвольной стороной вверх (черной или белой).

Определите минимальное количество карт, которое вам необходимо перевернуть на другую сторону (поменять черную на белую или белую на черную) так, чтобы перед вами получился ряд карт переменных цветов (любая пара соседних карт должна быть разных цветов).

*Входные данные.*

В стандартном входном потоке данных (*чтение с клавиатуры*) в первой строке записана строка  $S$ , задающая стороны карт в ряду. Каждый символ

строки равен или 'B' (карта лежит черной стороной вверх), или 'W' (карта лежит белой стороной вверх).

*Выходные данные.*

В стандартный выходной поток данных (*запись на экран*) ваша программа должна вывести одно целое число - минимальное количество карт, которые нужно перевернуть, чтобы получить ряд карт переменных цветов.

*Ограничения.*

Время выполнения программы не должно превышать 1 секунды для каждого теста. Количество символов в строке  $S$  не превышает 50.

*Примеры работы программы.*

<b>Вход</b>	BBBW	WBWBW	WWWWWWWWW
<b>Выход</b>	1	0	4

### **Задача №3. Бесконечная последовательность.** (100 баллов)

Вася придумал бесконечную последовательность из 0 и 1, получаемую следующим образом: вначале записывается 0, после чего бесконечно повторяется следующий процесс:

- Скопируйте уже записанную последовательность, заменяя в ней все 0 на 1, а 1 на 0;
- Допишите полученную копию после уже записанной последовательности.

Первые итерации процесса выглядят следующим образом:

0  
01  
0110  
01101001  
0110100110010110  
0110100110010110100101100110100

Позиции в последовательности пронумерованы слева направо начиная с 0.

У Васи возник вопрос: сколько единиц в последовательности в позициях с  $L$  до  $R$  включительно? Помогите Васе ответить на этот вопрос.

*Входные данные.*

В стандартном входном потоке данных (*чтение с клавиатуры*) записано через пробел два числа  $L$  и  $R$ , задающие интервал  $[L;R]$  в последовательности.

*Выходные данные.*

В стандартный выходной поток данных (*запись на экран*) ваша программа должна вывести единственное число - количество единиц в последовательности в позициях от  $L$  до  $R$  включительно.

*Ограничения.*

Время выполнения программы не должно превышать 1 секунды для каждого теста.

$$0 \leq L \leq R \leq 1\,000\,000\,000$$

*Примеры работы программы.*

<b>input.txt</b>	5 15	0 0	1 1
<b>output.txt</b>	5	0	1

#### **Задача №4. Игра с закрытыми глазами.** (100 баллов)

Вася и Маша играют в игру на прямоугольном поле шириной  $W$  клеток и высотой  $H$  клеток. Ряды и колонки пронумерованы начиная с 1. Клетка с координатами (1, 1) находится в левом верхнем углу поля. В некоторых клетках находятся стены, остальные пустые.

В начале игры у Васи и Маши есть по одной фишке на поле. В начале фишки всегда находятся на разных пустых клетках. Игроки ходят по очереди, Вася ходит первым. На каждом ходу игрок обязан переместить свою фишку на 1 клетку влево, вправо, вверх или вниз на пустую клетку (оставить фишку на месте во время своего хода нельзя, также как и нельзя ходить на клетку со стеной).

Если в любой момент времени обе фишки оказались на одной клетке, то выигрывает Вася. Если же Вася сдаётся (объяснение дальше), то выигрывает Маша.

Вася играет с закрытыми глазами, а Маша - с открытыми. Поскольку Вася все равно не видит происходящего на поле, то он выбирает свои ходы заранее и записывает на бумажке, которую видит Маша.

Вам дается описание игрового поля и выбранные заранее ходы Васи (гарантируется, что все ходы корректны и фишка Васи не выйдет за пределы поля и не попадет на клетку со стенкой).

Если Вася сделал все свои ходы, но так и не выиграл, то он сдаётся и выигрывает Маша. Маша в игре может использовать своё знание поля и ходов Васи, записанных на бумажке.

Определите, может ли Маша выиграть при оптимальной игре.

*Входные данные.*

В стандартном входном потоке данных (*чтение с клавиатуры*) в первой строке записаны через пробел 2 числа  $H$  и  $W$  - высота и ширина поля.

В следующих  $H$  строках по  $W$  символов в каждой записано состояние поля. Каждый символ - один из следующих:

- '.' (точка) - пустая клетка
- '#' - стена
- 'A' (латинская) - стартовая фишка Васи
- 'B' (латинская) - стартовая фишка Маши

В последней строке входного файла находится описание выбранных заранее ходов Васи. Каждый символ - один из следующих:

- 'U' - шаг вверх
- 'D' - шаг вниз
- 'L' - шаг влево
- 'R' - шаг вправо

*Выходные данные.*

В стандартный выходной поток данных (*запись на экран*) ваша программа должна вывести YES, если Маша может выиграть, и NO - если независимо от ходов Маши выигрывает Вася.

*Ограничения.*

Время выполнения программы не должно превышать 1 секунды для каждого теста.

$$1 \leq W, H \leq 50$$

Количество символов в строке ходов Васи - не больше 2500.

*Примеры работы программы.*

<b>Вход</b>	3 5 A.B.. ##.## ##.## RRDUR	3 5 A.B.. ##.## ##... RRRLD	5 5 ###.# ###.. A..B# ###.. ###.# RRRUDD
<b>Выход</b>	NO	YES	NO