Задача A. Вполоборота

Максимальное время работы на одном тесте: 1 секунда
Максимальный объем используемой памяти: 64 мегабайта

Словом считается непустая последовательность больших и маленьких латинских букв. Вам необходимо развернуть все слова во входном файле задом наперед.

Формат входных данных

Во входном файле записано все что угодно. Его размер не превышает 1 Мб.

Формат выходных данных

В выходном файле должен содержаться входной файл, в котором все слова развернуты. Все прочие символы, в том числе переводы строк, должны остаться нетронутыми

Пример

Входные данные	Выходные данные
Hello World	olleH dlroW

Система оценки

Тесты разделены на четыре группы, баллы за группу выставляются только при корректной работе программы на всех тестах группы.

1. *Тест из условия* **(0 баллов)**

2. *В этой группе тестов входной файл содержит единственную строку, состоящую из латинских букв и пробелов. Между словами ровно один пробел. Пробелов перед первым словом и после последнегослова отсутствуют. Размер файла не превышает 500 байт.*
   (30 баллов)

3. *Во входном файле записано все, что угодно. Размер файла не превышает 10 Кб.*
   (30 баллов)

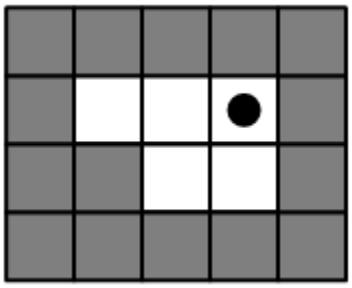
4. *Во входном файле записано все, что угодно. Размер файла не превышает 1 Мб.*
   (40 баллов)

Задача B. Вслепую по лабиринту

Максимальное время работы на одном тесте: 1 секунда
Максимальный объем используемой памяти: 64 мегабайт

Это интерактивная задача.

Ваша цель — написать программу, управляющую роботом, идущим вслепую по лабиринту.



Лабиринт состоит из N на M ($1 \le N, M \le 30$) клеток. Каждая из клеток может быть свободной или заблокированной (непроходимой). Все клетки на границе лабиринта непроходимые. Робот начинает работу в свободной клетке, он может переместиться на юг, запад, север или восток в свободную клетку. Робот не имеет оптических сенсоров, только сенсор удара, так что при попытке перемещения в заблокированную клетку сработает сенсор и робот останется в той же клетке.

Робот должен побывать во всех проходимых клетках лабиринта. Из начальной клетки гарантированно можно попасть во все достижимые клетки лабиринта.

Протокол интерактивного взаимодействия

Программа должна выводить на стандартный вывод одну строку с действием робота и ждать строки в стандартном вводе с ответом, затем выводить очередную строку с действием и считывать ответ и так далее до тех пор, пока все проходимые клетки лабиринта не будут посещены. Программа должна завершать работу только когда все проходимые клетки будут посещены. Проходимые клетки могут быть посещены несколько раз. Допустимо передвигаться даже если все проходимые клетки уже посещены.

Формат выходных данных

Каждая строка выходных данных должна представлять собой команду для робота. Это должна быть одна из пяти возможных строк: SOUTH, WEST, NORTH, EAST, или DONE. Строка DONE должна быть напечатана после посещения роботом всех проходимых клеток. После вывода DONE программа должна завершать свою работу. Необходимо очищать выходной буфер после вывода каждой команды (flush). Для этого используйте

1. `flush(output)` в Паскале или Delphi; (не исключено, что `writeln` делает это сам);
2. `fflush(stdout)` или `cout.flush()` в C/C++; (`endl` также это делает)
3. `stdout.flush()` в Python (опять же, стандартный `print` может делать это сам)

Формат входных данных

Каждая строка входных данных представляет собой ответ на действие робота. Это может быть строка EMPTY если робот успешно переместился в заданном направлении или строка BLOCKED если робот не смог переместиться из-за того, что клетка, в которую он хотел попасть, непроходима.

Примеры

Входные данные	Выходные данные
NORTH	BLOCKED
EAST	BLOCKED
SOUTH	EMPTY
EAST	BLOCKED
SOUTH	BLOCKED
WEST	EMPTY
SOUTH	BLOCKED
WEST	BLOCKED
NORTH	EMPTY
WEST	EMPTY
WEST	BLOCKED
NORTH	BLOCKED
EAST	EMPTY
NORTH	BLOCKED
DONE	

Задача C. Лестница

Максимальное время работы на одном тесте: 1 секунда
Максимальный объем используемой памяти: 64 мегабайта

Археологи раскопали Древний Храм, ко входу в который ведет лестница, шириной в 1 (один) метр, из M ступенек различной длины и высоты. Лестница построена из каменных блоков 1x1x1 метр. Археологи хотят для удобства туристов, чтобы лестница состояла из меньшего количества ступенек *N*. Для этого они могут также устанавливать каменные блоки 1x1x1. Какое минимальное количество блоков необходимо, чтобы сделать лестницу в *N* ступенек, если известны начальная длина и высота каждой ступеньки. Высоты и длины ступенек новой лестницы могут различаться.

Формат входных данных

В первой строке через пробел заданы два целых числа M и N ($1 \le N < M \le 100$). Далее идут M строк, содержащих пару целых чисел L и H – длина и высота i-ой ступеньки соответственно ($1 \le L, H \le 101$). Ступеньки нумеруются снизу вверх.

Формат выходных данных

В выходной файл выведите единственное число - ответ на задачу.

Пример

Входные данные	Выходные данные
5 3 4 2 1 2 5 2 1 2 2 1	3

Система оценки

Тесты разделены на четыре группы, баллы за группу выставляются только при корректной работе программы на всех тестах группы.

1. *Тест из условия* **(0 баллов)**
2. *В этой группе тестов все L = H = 1, N < M <= 100* **(30 баллов)**
3. *В этой группе тестов 1 <= L, H <= 101, N < M <= 10* **(30 баллов)**
4. *Дополнительных ограничений нет* **(40 баллов)**

Задача D. План уничтожения

Максимальное время работы на одном тесте: 2 секунды
Максимальный объем используемой памяти: 64 мегабайта

Секретный бункер уходит на N этажей вниз. Под нижним этажом бункера находится сверхсекретная лаборатория. Злобный диверсант хочет вывести лабораторию из строя, залив её водой (даже очень небольшого количества воды хватит, чтобы запоганить сверхточные приборы). Для этого он использует лужицы воды, остающиеся от жизнедеятельности обитателей бункера. В лужицах i-го этажа находится E_i воды. Диверсанту известно, что если на нём скопится больше C_i воды, то перегородка не выдержит и вся вода сольется на этаж ниже. Он может проделать отверстия в некоторых перегородках, по которым вода также стечет вниз. Проделать отверстие в полу i-го этажа стоит P_i у.е. Помогите диверсанту уничтожить лабораторию с минимальными материальными затратами.

Формат входных данных

Первая строка входного файла содержит натуральное число N ($1 \le N \le 500\,000$) – количество этажей в бункере, в следующих N строках находятся тройки целых чисел C_i, E_i, P_i ($0 < E_i \le C_i < 1\,000\,000$; $E_1+E_2+...+E_N < 2\,000\,000\,000$; $P_i > 0$; $P_1+P_2+...+P_N < 2\,000\,000\,000$). Этажи нумеруются сверху вниз.

Формат выходных данных

В первой строке выходного файла выдать количество денег, которое придется потратить злобному диверсанту, в следующих строках выведите номера этажей, в полу которых следует проделать отверстия.

Пример

Входные данные	Выходные данные
4 1 1 1 1 1 3 3 1 2 3 1 10	3 1 3

Система оценки

Тесты разделены на три группы, баллы за группу выставляются только при корректной работе программы на всех тестах группы.

1. *Тест из условия* **(0 баллов)**
2. *В этой группе тестов* $N <= 50000$ **(50 баллов)**
3. *Дополнительных ограничений нет* **(50 баллов)**