

Задача 1. Батареи

Входной файл	radiator.in
Выходной файл	radiator.out
Ограничение по времени	1 сек
Ограничение по памяти	256 МиБ
Максимальный балл за задачу	100

Вася решил обновить батареи у себя дома. Будем считать, что его дом имеет форму прямоугольного параллелепипеда со сторонами A , B и C метров. Одна батарея может обогреть X кубических метров жилого пространства. Посчитайте, какое наименьшее количество батарей необходимо, чтобы обогреть весь дом целиком.

Формат входных данных

В первой строке входного файла содержатся 4 числа A , B , C и X ($1 \leq A, B, C \leq 1000, 1 \leq X \leq 10\,000$).

Формат выходных данных

Выведите необходимое количество батарей.

Пример

Входной файл	Выходной файл
3 2 5 7	5

Задача 2. Штрихкод

Входной файл	barcode.in
Выходной файл	barcode.out
Ограничение по времени	1 сек
Ограничение по памяти	256 МиБ
Максимальный балл за задачу	100

Для автоматизации торговли и складского учета в последнее время широко распространены штрих-коды, при помощи которых кодируют числовые индексы товаров. Каждый штрихкод представляет собой последовательность из черных и белых полосок, при этом полоски могут быть узкими и широкими. Узкая полоска (как черная, так и белая) кодирует бит 0, широкая полоска кодирует бит 1.

Полученная битовая последовательность разбивается на группы из 4 бит, каждая группа кодирует одну десятичную цифру в соответствии с ее представлением в двоичной системе счисления (группа 0000 кодирует цифру 0, группа 0001 кодирует цифру 1, группа 0010 кодирует цифру 2, 0011 — цифру 3, 0100 — цифру 4 и т. д.). Если число, которое кодирует группа из 4 бит, больше 9, то эта группа не может быть декодирована и весь штрихкод считается ошибочным.

Напишите программу, которая по данному штрихкоду определяет его числовое значение.

Формат входных данных

Во входном файле содержится изображение штрихкода в виде строки из символов “B” и “W”. Строка состоит из чередующихся черных и белых полосок, каждая из которых может быть узкой или широкой. Узкая черная полоска кодируется символом “B”, широкая черная полоска кодируется символами “BB”, узкая и широкая белые полоски кодируются символами “W” и “WW” соответственно. Общее число полосок в строке равно 16, штрихкод начинается с черной полоски.

Формат выходных данных

Программа должна вывести четыре цифры — результат декодирования данного штрихкода. Если хотя бы одна из четырех цифр не может быть декодирована, выведите слово **ERROR**.

Пример

Входной файл	Выходной файл
BWBWBWBWBWBWBWW	0001
BWBWBWBWBWBWBWBWW	ERROR

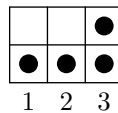
Задача 3. Патрульный робот

Входной файл	robot.in
Выходной файл	robot.out
Ограничение по времени	1 сек
Ограничение по памяти	256 МиБ
Максимальный балл за задачу	100

Дан робот, способный перемещаться по плоскости. За 1 шаг он перемещается на 1 метр в любом из четырех направлений. Программа для этого робота записывается на перфоленте, причем приняты такие обозначения:

- А) отсутствие обоих отверстий в некоторой позиции — движение на север;
- Б) отверстие только сверху ленты — движение на юг;
- В) отверстие только внизу ленты — движение на запад;
- Г) отверстия и сверху, и внизу ленты — движение на восток.

На рисунке изображен кусочек ленты, содержащий две команды движения на запад и одну команду движения на восток.



Для того, чтобы экономить ленту, ее закольцовывают (склеивают начало и конец). Таким образом робот может исполнять записанную серию команд несколько раз, пока у него не кончится заряд батареи.

Однажды техник, который обслуживал робота, допустил ошибку и склеил перфоленту в виде ленты Мёбиуса (перед тем, как склеить концы, он один раз перекрутил ленту). Когда это обнаружилось, возник вопрос: в какой же точке следует теперь ожидать робота? Напишите программу, которая рассчитает координаты робота, если известно, что заряда батареи хватает ровно на N шагов.

Формат входных данных

В первой строке входного файла записано число N (целое положительное, не превышающее 10 000) — на сколько шагов хватает заряда батареи робота. Во второй строке записана программа для робота, содержащая символы:

- 'N' — шаг на север (в положительном направлении оси y),
- 'S' — шаг на юг (в отрицательном направлении оси y),
- 'E' — шаг на восток (в положительном направлении оси x),
- 'W' — шаг на запад (в отрицательном направлении оси x).

Длина программы не превышает 200 символов, программа не пуста. Известно, что в начальный момент времени робот находится в точке с координатами $(0, 0)$. Каждый шаг имеет длину 1.

Формат выходных данных

Выведите координаты точки, где остановится робот.

Пример

Входной файл	Выходной файл
8 NNSS	-2 2
2 ESNW	1 -1

Задача 4. Ханойская сортировка

Входной файл	pyramid.in
Выходной файл	pyramid.out
Ограничение по времени	1 сек
Ограничение по памяти	256 МиБ
Максимальный балл за задачу	100

Дано три колышка. На первый колышек надета пирамидка из N дисков разного диаметра, размещенных в произвольном порядке, два других колышка пусты. Разрешается перекладывать по одному верхнему диску с одного колышка на другой (**независимо от их размера**). Необходимо разместить все диски на первом колышке в порядке возрастания их диаметра, если считать снизу вверх.

Формат входных данных

Первая строка входного файла содержит число N ($N \leq 50$) — количество дисков в пирамидке. Во второй строке записаны числа $1, \dots, N$ по одному разу в некотором порядке — диаметры дисков, в порядке снизу вверх.

Формат выходных данных

Программа должна вывести последовательность перекладываний дисков. Каждое перекладывание состоит из двух чисел, разделенных пробелами: номер колышка с которого берется диск и номер колышка на который кладется диск. На одной строке выводится одно перекладывание.

Количество выведенных строк должно быть не больше 10^5 .

Пример

Входной файл	Выходной файл
2	1 2
2 1	1 3
	2 1
	3 1

Задача 5. Игрушечный лабиринт

Входной файл	labirint.in
Выходной файл	labirint.out
Ограничение по времени	1 сек
Ограничение по памяти	256 МиБ
Максимальный балл за задачу	100

Дан игрушечный лабиринт, в нем находится шарик. Лабиринт можно наклонять влево, вправо, к себе или от себя, после чего шарик катится в сторону наклона до тех пор, пока не встретится с каким-нибудь препятствием, с отверстием в полу, или с границей лабиринта. Целью игры является, наклоняя лабиринт, загнать шарик в любое из таких отверстий-выходов. Определите, какое минимальное количество наклонов необходимо сделать, чтобы этого добиться. Гарантируется, что решение существует. В начале шарик находится в левом верхнем углу лабиринта. Гарантируется, что левый верхний угол лабиринта всегда свободен.

Формат входных данных

В первой строке входного файла записаны числа N и M — размеры лабиринта (целые положительные числа, не превышающие 100). Затем идет N строк по M чисел в каждой — описание лабиринта. Число 0 в описании означает свободное место, число 1 — препятствие, число 2 — отверстие.

Например, лабиринту, изображенному на рисунке, будет соответствовать такое описание:

	4 5 0 0 0 0 1 0 1 1 0 0 0 0 1 0 1 0 2 1 0 2
--	---

Формат выходных данных

Выведите единственное число — минимальное количество наклонов, которые необходимо сделать, чтобы шарик покинул лабиринт через какое-нибудь из отверстий.

Пример

Входной файл	Выходной файл
4 5 0 0 0 0 1 0 1 1 0 0 0 0 1 0 1 0 2 1 0 2	2