

Задача С. Монетки

Имя входного файла: `money.in`
Имя выходного файла: `money.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Участник олимпиады «Юниор» в перерыве между турами решил сходить в кофе-пойнт возле площадки проведения. У участника с собой одна купюра некоторого номинала, а у продавца кофе-пойнта к концу недели осталась только мелочь, чтобы выдать сдачу. Вместе с тем в кофе-пойнте очередь, и выдавать сдачу необходимо как можно более оперативно, вследствие чего перед продавцом стоит задача отсчитать точное число необходимых монет для этого.

Для данных номиналов монет, имеющихся в распоряжении у продавца, и сдачи, которую нужно собрать, определите минимальное количество монет, которое необходимо отсчитать продавцу кофе-пойнта.

Формат входных данных

В первой строке дано целое число n ($1 \leq n \leq 10$) — количество номиналов монет, которые могут быть в кассе.

В следующей строке вводятся n целых чисел m_i ($1 \leq m_i \leq 2\,000$), обозначающих соответствующий номинал монеты.

В последней строке дано одно целое число s ($1 \leq s \leq 1\,000\,000$) — сдача, которую требуется выдать.

Формат выходных данных

Выведите минимальное количество монет, которыми можно выдать сдачу нужного размера, или -1 в случае, если это нельзя сделать.

Система оценки

Каждый тест данной задачи оценивается в $\frac{1}{3}$ балла с округлением итоговой суммы баллов до 2-х знаков после запятой по математическим правилам, при этом в случае, если программа на одном из тестов выдала неправильный ответ, дальнейшее тестирование задачи не продолжается.

Примеры

<code>money.in</code>	<code>money.out</code>
4 5 3 10 1 1	1
4 3 1 5 10 239	26

Замечание

В этой задаче будем считать, что монеты могут быть не только привычных номиналов, а любых значений от 1 до 2000.

Задача D. Волонтёры

Имя входного файла: `volunteers.in`
Имя выходного файла: `volunteers.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

На олимпиаде работает V волонтёров. Время работы поделено на 10 000 одинаковых отрезков времени, и известно, когда конкретный волонтёр приходит на свой участок и уходит с олимпиады. К примеру, если волонтёр приходит в отрезок времени 1, а уходит в отрезок 4, это означает его присутствие в 1, 2, 3 и 4-м отрезках времени. Накануне олимпиады стало известно, что её участников будет несколько меньше, чем было изначально зарегистрировано.

Для каждого известного набора данных о времени работы волонтёров проверьте, что в течение всего времени проведения олимпиады на ней присутствует хотя бы один волонтёр, и в случае ухода любого из них появляется отрезок времени, когда на олимпиаде отсутствуют волонтёры.

Формат входных данных

В первой строке дано натуральное число D ($1 \leq D \leq 100$) — количество наборов данных о волонтёрах.

Каждый набор данных начинается с новой строки и содержит число V , за которым следует V пар неотрицательных целых чисел S и F — время прихода на олимпиаду и ухода ($0 \leq S < F \leq 10\,000$) соответствующего волонтёра.

Формат выходных данных

Выведите D строк, в каждой из которых находится слово `Good`, если для данного набора данных о времени работы волонтёров обеспечиваются необходимые условия проведения олимпиады. Иначе выводится слово `Bad`.

Система оценки

Каждый тест данной задачи оценивается в $\frac{10}{19}$ балла, кроме тестов из условия. Тестирование задачи в случае неправильного ответа на одном из тестов идёт до конца по всему набору тестов.

Пример

<code>volunteers.in</code>	<code>volunteers.out</code>
2	Bad
3 0 3000 2500 7000 2700 10000	Good
2 0 3000 2700 10000	

Замечание

В примере входных данных в первом случае несмотря на покрытие всей олимпиады хотя бы одним волонтёром, не выполняется второе условие, что при удалении любого волонтёра появляется промежуток времени без волонтёров. В этом случае при удалении второго волонтёра такой промежуток не появляется — поэтому ответ на этот случай `Bad`.

Задача Е. Несложная задача

Имя входного файла: `easy.in`
Имя выходного файла: `easy.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

В этой задаче строку, состоящую только из символов 0, 1, 2, будем называть *несложной*, если количество символов, обозначающих эти три цифры, одинаково, при этом любые нули должны идти левее единиц, а любые единицы левее двоек.

Например, строка 001122 является *несложной*, а строка 0000111122220 — сложной.

Формат входных данных

В первой строке вводится число n ($n \leq 10$) — количество строк для проверки на несложность.

В следующих n строках вводится строка из символов 0, 1, 2. Длина каждой из строк не превосходит 30 000 символов.

Каждая из строк состоит хотя бы из одного символа. Гарантируется, что других символов, помимо указанных, в строке быть не может.

Формат выходных данных

В ответе выведите n строк, в каждой из которых выведите соответственно слово **EASY**, если строка на соответствующей позиции является несложной, или слово **HARD** в противном случае.

Система оценки

Каждый тест данной задачи оценивается в $\frac{2}{3}$ балла с округлением итоговой суммы баллов до 2-х знаков после запятой по математическим правилам, при этом тестирование задачи идёт до конца по всему набору тестов.

Примеры

<code>easy.in</code>	<code>easy.out</code>
3 001122 00011122222 000111222	EASY HARD EASY
2 0000111122220 012	HARD EASY

Задача F. Похожие строки

Имя входного файла: `strings.in`
Имя выходного файла: `strings.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Дизайнер создаёт графические элементы из букв некоторого слова. Он считает, что, если переставить некоторые символы из начала слова в его конец, а оставшиеся символы в начало, получится интересное сочетание для типографики.

Если говорить более формально, то в некоторой строке s (длина s не превышает 50 000 символов), которую можно представить как последовательность символов $s_0s_1 \dots s_is_{i+1} \dots s_n$, удаляются i символов из начала и переставляются в конец, в результате чего получается строка вида $s_{i+1} \dots s_ns_is_{i-1} \dots s_1$, где $i \leq n$.

По данным двум строкам выясните, можно ли из первой строки получить вторую такими преобразованиями.

Формат входных данных

Каждый тест состоит из двух строк: первая является исходной; вторая — возможным результатом преобразования исходной.

Формат выходных данных

В случае, если преобразование первой строки во вторую возможно, выведите в первой строке слово **Yes**, а во второй i — количество подряд идущих символов, которые необходимо переместить по строке (в ответ необходимо взять такое минимальное значение i).

Система оценки

Каждый тест данной задачи оценивается в $\frac{5}{7}$ балла с округлением итоговой суммы баллов до 2-х знаков после запятой по математическим правилам. В случае неправильного ответа на произвольный тест дальнейшее тестирование решения приостанавливается.

Примеры

<code>strings.in</code>	<code>strings.out</code>
wpwdwpw wdwpwpw	Yes 2
dWOddWd dOdWdWd	No

Задача G. Таблица

Имя входного файла:	<code>table.in</code>
Имя выходного файла:	<code>table.out</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта

Рассмотрим прямоугольную таблицу размера $x \times y$, разбитую на единичные клетки. Требуется полностью покрыть эту таблицу одинаковыми фигурами специального вида.

Каждая фигура представляет собой связную конфигурацию из 6 клеток внутри квадрата 3×3 . *Связность* означает, что из любой клетки фигуры можно попасть в любую другую, переходя только по сторонам клеток. Фигура задаётся шаблоном из символов «Н» (занятая клетка) и «о» (пустая клетка).

Все фигуры одинаковы и могут размещаться на поле только в исходной ориентации (запрещено поворачивать и отражать фигуру). Фигуры располагаются так, чтобы их клетки совпадали с клетками таблицы.

Покрытием таблицы называется такое размещение фигур, при котором каждая клетка таблицы покрыта хотя бы одной фигурой. Допускается перекрытие фигур (одна клетка может быть покрыта несколькими фигурами), а также допускается выход частей фигур за границы таблицы.

Требуется определить минимальное количество фигур, необходимое для покрытия всей таблицы.

Формат входных данных

В первой строке заданы два целых числа x и y ($1 \leq x, y \leq 9$) — размеры таблицы.

В следующих трёх строках задан шаблон фигуры размером 3×3 .

Каждая строка содержит ровно три символа двух видов: «Н» — клетка фигуры или «о» — пустая клетка.

Гарантируется, что фигура содержит ровно 6 клеток «Н» и является связной.

Формат выходных данных

Выведите одно число — минимальное количество фигур, необходимое для покрытия всей таблицы.

Система оценки

Каждый тест данной задачи оценивается в $\frac{5}{47}$ балла с округлением до 2-х знаков после запятой по математическим правилам, при этом в случае неверных ответов на произвольный тест задача тестируется до конца по всему набору тестов.

Примеры

table.in	table.out
3 3 oHo HHH HHo	3
3 2 HHH HHH ooo	1
2 3 HHH HHH ooo	2
2 2 HHH HoH ooH	2