

Задача А. Башни

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 megabytes

Маленький Вася недавно получил в подарок от старшего брата набор юного строителя. Этот набор состоит из нескольких деревянных брусков, для каждого из которых известна его длина. Бруски можно класть сверху один на другой, если длины брусков совпадают.

Вася хочет соорудить из всех брусков минимальное количество башенок. Помогите Васе расположить бруски оптимальным образом.

Формат входного файла

В первой строке записано целое число N ($1 \leq N \leq 1000$) — количество брусков, имеющих в распоряжении у Васи. Во второй строке через пробел записано N целых чисел l_i — длины брусков. Все длины — натуральные числа, не превосходящие 1000.

Формат выходного файла

В одной строке выведите два числа — высоту наибольшей из башен и их общее количество. Помните, что Вася должен использовать все бруски.

Примеры

<code>stdin</code>	<code>stdout</code>
3 1 2 3	1 3
4 6 5 6 7	2 3

Задача В. Компьютерная игра

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 megabytes

Петя, старший брат Васи, очень любит играть в компьютерные игры. В одной из своих любимых игр Петя дошел до последнего уровня, на котором происходит битва с боссом.

Во время прохождения игры Петя находил свитки с заклинаниями, и теперь ему предстоит их использовать. Опишем, как происходит бой на этом уровне:

1) У босса есть два параметра max — начальное количество здоровья, и reg — ежесекундная регенерация.

2) У каждого свитка также есть два параметра: pow_i — сила заклинания, измеряемая в процентах — максимальное количество здоровья босса, отсчитываемое от исходного, при котором можно использовать данный свиток (т. е. если у босса больше pow_i процентов здоровья, свиток применить нельзя); и dmg_i — урон в секунду, который получает босс, если использовать этот свиток. Как только какой-то свиток использован, он пропадает, и на босса накладывается еще одно заклинание, которое будет наносить ему dmg_i урона каждую секунду до конца игры.

Действия в бою в течение одной секунды, происходят в следующем порядке: сначала босс получает урон от всех заклинаний, наложенных на него, затем восстанавливает reg здоровья (при этом, больше чем max здоровья получиться не может), а затем игрок может использовать очередной свиток (не больше одного в секунду).

Босс считается поверженным, если в конце какой-то секунды количество здоровья стало неположительным (≤ 0).

Помогите Пете определить, может ли он выиграть с имеющимся набором свитков, и если может, то за какое минимальное количество секунд он сможет это сделать.

Формат входного файла

В первой строке записаны три целых числа N , max и reg ($1 \leq N, max, reg \leq 1000$) — количество свитков, и параметры босса. В следующих N строках записаны по два целых числа pow_i и dmg_i — параметры i -го свитка ($0 \leq pow_i \leq 100$, $1 \leq dmg_i \leq 2000$).

Формат выходного файла

В случае если Петя не сможет пройти этот уровень, в единственной строке выведите `NO`.

Иначе в первой строке выведите `YES`. Во второй строке выведите минимальное время, через которое босс может быть повержен и количество использованных свитков. В следующих строках для каждого использованного свитка выведите через пробел количество секунд, прошедших с начала боя, до его использования и номер использованного свитка. Свитки нумеруются с 1 в том порядке, в котором они заданы во входных данных. Считается, что первый свиток можно использовать через 0 секунд.

Свитки выводите в том порядке, в котором их нужно использовать. Запрещается использовать свитки после того как босс повержен.

Примеры

<code>stdin</code>	<code>stdout</code>
2 10 3 100 3 99 1	NO
2 100 10 100 11 90 9	YES 19 2 0 1 10 2

Задача С. Древнеберляндский язык

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 megabytes

Берляндские ученые знают, что в древнеберляндском языке было ровно n слов. Эти слова имели длины l_1, l_2, \dots, l_n букв. В словах древнеберляндцы использовали две буквы, 0 и 1. Древние берляндцы говорили быстро и не делали пауз между словами, но при этом всегда могли однозначно понять друг друга. Это было возможно потому, что никакое слово не было префиксом никакого другого (префиксом строки называется некоторая её подстрока, начинающаяся с первого символа).

Помогите ученым определить, можно ли восстановить все слова из древнеберляндского языка, и если можно, выведите сами слова.

Формат входного файла

В первой строке записано одно целое число N ($1 \leq N \leq 1000$) — количество слов в древнеберляндском языке. Во второй строке, через пробелы, записаны N целых чисел — длины этих слов. Все длины — натуральные числа, не превосходящие 1000.

Формат выходного файла

Если такого набора слов не существует, в единственной строке выведите `NO`. Иначе, в первой строке выведите `YES`, а в следующих N строках выведите сами слова в том порядке в каком даны их длины во входном файле. Если решений несколько, выведите любое.

Примеры

<code>stdin</code>	<code>stdout</code>
3 1 2 3	YES 0 10 110
3 1 1 1	NO

Задача D. Расписание занятий

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 megabytes

В свободное от компьютерных игр время Петя ходит в университет на занятия. Каждый день занятия на факультете у Пети состоят из двух пар. Этаж на котором проходят занятия у Пети, представляет из себя длинный коридор, вдоль которого расположены M аудиторий, пронумерованных от 1 до M .

Поток, на котором учится Петя, разбит на N групп. Недавно, Петя заметил, что расписание занятий у этих групп обладает следующей особенностью: номер аудитории, в которой проходит первое занятие у какой-то группы не превосходит номера аудитории для второго занятия этой группы.

Как-то раз, Петя решил посчитать, сколькими способами можно составить расписание занятий для всех групп. Расписание — это набор из $2N$ чисел: для каждой группы номер аудитории для первого занятия, и номер аудитории для второго занятия. К сожалению, он быстро сбился со счета, и решил, считать только расписания, удовлетворяющие следующим условиям:

- 1) На первом занятии в i -ой аудитории должно находиться ровно X_i групп.
- 2) В i -ой аудитории может поместиться не более чем Y_i групп.

Помогите Пете посчитать количество расписаний удовлетворяющих всем этим условиям. Так как и таких расписаний может быть много, выведите ответ по модулю $10^9 + 7$.

Формат входного файла

В первой строке записано одно целое число M ($1 \leq M \leq 100$) — количество аудиторий.

Во второй строке, через пробел, записаны M целых чисел — X_i ($1 \leq X_i \leq 100$) количество групп, находящихся в i -ой аудитории во время первого занятия.

Во третьей строке, через пробел, записаны M целых чисел — Y_i ($1 \leq Y_i \leq 100$) максимальное количество групп, которые могут одновременно находиться в i — аудитории.

Гарантируется, что все $X_i \leq Y_i$, и что сумма всех X_i положительна и не превосходит 1000.

Формат выходного файла

В единственной строке выведите ответ на задачу по модулю $10^9 + 7$.

Примеры

<code>stdin</code>	<code>stdout</code>
3 1 1 1 1 2 3	36
3 1 1 1 1 1 1	6

Note

Во втором тесте из примера первое и второе занятия у каждой группы должны проходить в одной и той же аудитории, поэтому расписания будут отличаться только перестановкой номеров этих аудиторий для каждой группы, то есть $3! = 6$.

Задача E. Испытание для вождя

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 megabytes

Разгадав словарь древнеберляндского языка, ученые смогли прочитать записи летописцев того времени. В частности, они узнали, как происходил выбор вождя древнеберляндского племени.

Как только на место вождя набиралось достаточное количество претендентов, среди них проводилось следующее испытание: шаман племени брал плиту, разделенную вертикальными и горизонтальными полосами на одинаковые клетки (плита состояла из N строк и M столбцов), и красил каждую клетку в черный или белый цвет. Затем каждому испытуемому выдавалась плита такого же размера, но покрашенная полностью в белый цвет. За один день он мог перекрасить любое связанное по стороне множество клеток этой плиты в некоторый цвет. Множество называется связным, если для любых двух клеток этого множества существует путь, все клетки которого принадлежат этому множеству, и любые две соседние клетки на этом пути имеют общую сторону. Цель каждого из претендентов — получить в точности такую же раскраску как у шамана. Тот, кто получал такую раскраску раньше всех, становился новым вождем племени.

Ученые обнаружили плиту, раскрашенную шаманом древнеберляндского племени. Помогите им определить, за какое минимальное количество дней мог определиться новый вождь, если ему нужно было получить заданную раскраску.

Формат входного файла

В первой строке записаны два целых числа N и M ($1 \leq N, M \leq 50$) — количество строк и столбцов у плиты. В следующих N строках записаны по M символов — итоговая раскраска плиты. Символ W означает, что клетка должна быть покрашена в белый цвет, а B — в черный.

Формат выходного файла

В единственной строке выведите минимальное количество перекрашиваний связных областей, необходимых для получения искомой раскраски плиты.

Примеры

stdin	stdout
3 3 WBW BWB WBW	2
2 3 BBB BWB	1