

Задача А. 17 стульев

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

64 мегабайта

Остап Бендер снова пытается получить причитающиеся драгоценности, но на этот раз они были заперты в шкатулке, для открытия которой необходимо иметь N ключей. По закономерной случайности каждый из ключей был спрятан в одном из N стульев, распроданных на недавнем аукционе. После аукциона эти стулья были развезены в N городов.

И вот теперь Остап решил на новую безумную затею: заехать в каждый из городов и, провернув в каждом из них аферу, выкрасть необходимые ключи. Чтобы избежать конфликтов с недоброжелателями, Остап не хочет больше одного раза появляться в каком-либо городе. Также у Остапа есть список цен за проезд между каждой парой городов. Изначально Остап находится в городе под номером 1 и после посещения всех городов может незаметно скрыться из этой страны.

Помогите Остапу найти порядок посещения городов, при котором ему потребуется потратить как можно меньше средств на странствия, и тогда, возможно, он поделится с Вами добытыми бриллиантами.

Формат входных данных

Первая строка содержит единственное число N — количество городов ($1 \leq N \leq 17$).

Следующие N строк содержат по N целых неотрицательных чисел. j -тое число в i -й строке означает стоимость проезда из города i в город j ($0 \leq a_{ij} \leq 100$). Если $a_{ij} > 0$, то проезд стоит a_{ij} рублей, иначе — это означает, что из города i в j невозможно проехать напрямую.

Формат выходных данных

В первой строке выведите минимальную сумму денег, необходимую для посещения всех городов Остапом. В следующей строке выведите N чисел — порядок посещения городов, при котором эта сумма достигается. Если затею Остапа невозможно вывести, то в единственной строке выходного файла выведите число -1.

Примеры

стандартный ввод	стандартный вывод
3 0 3 2 3 0 6 2 6 0	8 1 3 2
5 0 6 4 0 0 6 0 7 0 7 4 7 0 0 0 0 0 0 0 2 0 7 0 2 0	20 1 3 2 5 4

Задача В. Миллиардная Функция Васи

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вася — начинающий математик — решил сделать вклад в развитие этой науки и прославиться на весь мир. Но как это сделать, когда самые интересные факты, типа теоремы Пифагора, давно уже доказаны? Правильно! Придумать что-то свое, оригинальное. Вот юный математик и придумал Теорию Функций Васи, посвященную изучению поведения этих самых функций. Функции Васи (ФВ) устроены довольно просто: значением N -й ФВ в точке S будет количество чисел от 1 до N , имеющих сумму цифр S . Вам, как крутым программистам, Вася поручил найти значения миллиардной ФВ (то есть ФВ с $N = 10^9$), так как сам он с такой задачей не справится. А Вам слабо?

Формат входных данных

В единственной строке записано целое число S ($1 \leq S \leq 81$).

Формат выходных данных

Выведите одно число — значение миллиардной Функции Васи в точке S .

Пример

стандартный ввод	стандартный вывод
1	10

Задача С. Дубы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

На аллее перед зданием Министерства Обороны в ряд высажены n дубов. В связи с грядущим приездом главнокомандующего, было принято решение срубить несколько деревьев для придания аллее более милитаристического вида.

Внутренние распоряжки министерства позволяют срубить дуб только в двух случаях:

- если и ближайший дуб слева, и ближайший дуб справа строго ниже, чем данный дуб;
- если и ближайший дуб слева, и ближайший дуб справа строго выше, чем данный дуб.

В частности, согласно этому правилу, нельзя срубить крайний левый и крайний правый дубы.

Министр хочет выработать такой план вырубki, чтобы в итоге осталось несколько дубов, высоты которых образуют неубывающую последовательность, то есть чтобы каждый дуб был не ниже, чем все дубы, стоящие слева от него. При этом, как человек любящий флору, министр хочет, чтобы было срублено минимальное возможное количество деревьев.

Помогите сотрудникам министерства составить оптимальный план вырубki аллеи или выяснить, что срубить дубы соответствующим образом невозможно.

Формат входных данных

Первая строка входного файла содержит целое число n — количество дубов, растущих на аллее ($2 \leq n \leq 200$). Вторая строка содержит n чисел — высоты дубов, приведенные слева направо. Высоты дубов — положительные целые числа, не превышающие 1 000.

Формат выходных данных

Если оставить последовательность дубов с неубывающими высотами невозможно, выходной файл должен содержать только одно число -1 .

В случае, если искомый план существует, в первую строку выходного файла выведите целое число m — минимальное количество дубов, которые необходимо срубить. В следующие m строк выведите оптимальный план вырубki деревьев — номера дубов в том порядке, в котором их следует срубить, по одному номеру на строке.

Дубы нумеруются слева направо натуральными числами от 1 до n .

Если планов с наименьшим числом срубаемых дубов несколько, выведите любой из них.

Пример

стандартный ввод	стандартный вывод
5	2
3 2 4 8 5	4
	2

Задача D. Деловые встречи

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Алексей — успешный предприниматель, и в течение одного дня у него бывает много встреч с разными деловыми партнёрами. К сожалению, встречи бывают разные и не все приносят ему радость, после других же настроение улучшается. Также, на многие встречи не стоит приходить в слишком плохом или хорошем настроении — результат таких встреч может быть не таким, какой хочется Алексею.

К счастью, недавно Алексей научился оценивать своё настроение с помощью целых чисел. После этого для каждой встречи он оценил, при каком максимальном и минимальном настроении стоит на неё приходить, а также как изменится его настроение после этой встречи. Теперь он хочет распланировать порядок встреч так, чтобы в течение дня совершить максимальное число встреч.

Ваша задача — написать программу, которая по информации о всех встречах и настроении Алексея в начале дня находит порядок проведения встреч такой, что их количество при этом максимально.

Формат входных данных

Первая строка входного файла содержит два целых числа n и k ($1 \leq n \leq 20$, $-100 \leq k \leq 100$) — количество встреч и настроение Алексея в начале дня.

Следующие n строк содержат по три целых числа a_i , b_i и c_i ($-100 \leq a_i, b_i, c_i \leq 100$) — минимальное и максимальное настроение, при котором встреча возможна, и изменение настроения по окончании встречи, соответственно.

Формат выходных данных

В первой строке выходного файла выведите число m — максимально возможно число встреч. В следующей строке выведите m целых чисел — номера встреч в порядке их проведения. Встречи пронумерованы в порядке описания во входном файле.

Если ответов с максимальным числом встреч несколько, выведите любой.

Примеры

стандартный ввод	стандартный вывод
3 0 1 3 3 0 1 2 1 3 1	3 2 3 1
3 1 -10 -5 3 -5 5 -2 -3 2 1	2 3 2

Задача Е. Плавные числа

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Назовём натуральное число плавным, если значения соседних цифр отличаются не более, чем на

1. Определите количество N -значных плавных чисел. Запись числа не может начинаться с цифры 0.

Формат входных данных

Программа получает на вход одно целое число N ($1 \leq N \leq 20$).

Формат выходных данных

Программа должна вывести одно целое число — искомое количество плавных чисел.

Пример

стандартный ввод	стандартный вывод
2	26

Задача F. Распил брусьев

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вам нужно распилить деревянный брус на несколько кусков в заданных местах. Распилочная компания берет k рублей за распил одного бруска длиной k метров на две части.

Понятно, что различные способы распила приводят к различной суммарной стоимости заказа. Например, рассмотрим брус длиной 10 метров, который нужно распилить на расстоянии 2, 4 и 7 м, считая от одного конца. Это можно сделать несколькими способами. Можно распилить сначала на отметке 2 м, потом 4 и, наконец, 7 м. Это приведет к стоимости $10+8+6=24$, потому что сначала длина бруса, который пилили, была 10 м, затем она стала 8 м, и, наконец, 6 м. А можно распилить иначе: сначала на отметке 4 м, затем 2, затем 7 м. Это приведет к стоимости $10+4+6=20$, что лучше.

Определите минимальную стоимость распила бруса на заданные части.

Формат входных данных

Первая строка входных данных содержит целое число L ($2 \leq L \leq 10^6$) - длину бруса и целое число N ($1 \leq N \leq 100$) - количество распилов. Во второй строке записано N целых чисел C_i ($0 < C_i < L$) в строго возрастающем порядке - места, в которых нужно сделать распилы.

Формат выходных данных

Выведите одно натуральное число - минимальную стоимость распила.

Пример

стандартный ввод	стандартный вывод
10 3 2 4 7	20

Задача G. Удаление скобок

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 8 мегабайт

Обратите внимание на необычное ограничение по памяти.

Дана строка, составленная из круглых, квадратных и фигурных скобок. Определите, какое наименьшее количество символов необходимо удалить из этой строки, чтобы оставшиеся символы образовывали правильную скобочную последовательность.

Формат входных данных

Входные данные состоят из строки из круглых, квадратных и фигурных скобок. Длина строки не превосходит 700 символов.

Формат выходных данных

Выведите строку максимальной длины, являющуюся правильной скобочной последовательностью, которую можно получить из исходной строки удалением некоторых символов.

Если возможных ответов несколько, выведите любой из них.

Примеры

стандартный ввод	стандартный вывод
([]]	[]
{([({})])}	[({})]

Задача Н. Интересные числа

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Софья считает число интересным, если его цифры идут в неубывающем порядке. Например, числа 123, 1111 или 888999 – интересные.

Софья заинтересовалась, сколько существует интересных положительных чисел, лежащих в диапазоне от L до R включительно. Это число может оказаться довольно большим для больших L и R , поэтому Софья хочет найти остаток от деления этого числа на $10^9 + 7$.

Требуется написать программу, которая по заданным L и R определяет количество интересных чисел, лежащих в диапазоне от L до R включительно, и выводит остаток от деления этого числа на $10^9 + 7$.

Формат входных данных

Входной файл содержит две строки. Первая строка содержит число L , вторая строка содержит число R ($1 \leq L \leq R \leq 10^{100}$).

Формат выходных данных

Выходной файл должен одно целое число — остаток от деления количества интересных чисел, лежащих в диапазоне от L до R включительно, на $10^9 + 7$.

Пример

стандартный ввод	стандартный вывод
1 100	54

Задача I. Острова

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Дана карта островов и мостов, соединяющих некоторые острова. Как все мы знаем, Гамильтонов путь — это путь, проходящий по некоторым мостам, такой что каждый остров будет посещен ровно один раз. На имеющейся карте каждому острову также присвоено некоторое положительное целое число. Назовем Гамильтонов путь лучшим треугольным Гамильтоновым путем, если он максимизирует описанное ниже значение.

Пусть есть n островов. Стоимость Гамильтонова пути C_1, C_2, \dots, C_n вычисляется как сумма трех слагаемых. Пусть V_i — число, присвоенное острову C_i . Первое слагаемое равно сумме значений V_i всех посещенных вершин. Второе слагаемое состоит из суммы произведений $V_i \cdot V_{i+1}$ для каждого ребра $C_i \rightarrow C_{i+1}$ в пути. Третье слагаемое считается следующим образом. Для каждой тройки соседних вершин $C_i \rightarrow C_{i+1} \rightarrow C_{i+2}$ добавим в слагаемое $V_i \cdot V_{i+1} \cdot V_{i+2}$, если между вершинами C_i и C_{i+2} есть ребро.

От вас требуется найти Гамильтонов путь с максимальной стоимостью. Вполне возможно, что существует не один Гамильтонов путь с такой стоимостью, поэтому также найдите количество таких Гамильтоновых путей.

Формат входных данных

В первой строке записано число q , означающее количество тестовых случаев ($1 \leq q \leq 20$). Далее следует описание тестовых случаев в следующем формате.

В первой строке записаны числа n и m ($1 \leq n \leq 13$, $0 \leq m \leq \frac{n(n-1)}{2}$) — количество островов и мостов соответственно.

Во второй строке записаны n целых чисел V_i ($1 \leq V_i \leq 100$) — числа, присвоенные островам.

В каждой из следующих m строк описаны мосты в формате $x y$ ($1 \leq x, y \leq n$, $x \neq y$), что означает, что между островами x и y имеется двунаправленный мост.

Формат выходных данных

Для каждого тестового случая выведите два числа — максимальную стоимость Гамильтонова пути и количество Гамильтоновых путей такой стоимости соответственно.

Если в тестовом случае не существует Гамильтонова пути, выведите 0 0.

Пример

стандартный ввод	стандартный вывод
2	22 3
3 3	69 1
2 2 2	
1 2	
2 3	
3 1	
4 6	
1 2 3 4	
1 2	
1 3	
1 4	
2 3	
2 4	
3 4	

Замечание

Один и тот же путь может быть записан в обратном порядке. Будем считать такие пути одинаковыми.