

## Задача А. 17 стульев

Имя входного файла: **trader.in**  
Имя выходного файла: **trader.out**  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 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  $i$ -той 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

trader.in	trader.out
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. Раскраска графа

Имя входного файла: coloring.in  
Имя выходного файла: coloring.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 32 мегабайта

Дан граф из  $n$  вершин, раскрасьте его в минимально возможное число цветов так, чтобы никакие две вершины соединенные ребром не были одного цвета.

### Формат входных данных

В первой строке содержится число  $t$  — количество тестовых примеров ( $1 \leq t \leq 5$ ).

Далее содержится  $t$  тестовых случаев, заданных в следующем формате:

В первой строке записаны числа  $n$  и  $m$  — количество вершин и ребер соответственно ( $1 \leq n \leq 17$ ,  $0 \leq m \leq \frac{n \cdot (n-1)}{2}$ ).

Затем идет  $m$  строк, в которых содержится по два числа  $v_i$   $u_i$ , что означает, что вершины  $v_i$  и  $u_i$  соединены ребром ( $1 \leq v_i, u_i \leq n$ ,  $v_i \neq u_i$ ).

Гарантируется, что все ребра в каждом тестовом случае различны.

### Формат выходных данных

Для каждого тестового случая в первой строке выведите минимальное число цветов  $k$ .

Во второй строке выведите  $n$  чисел  $a_i$  — цвета вершин ( $1 \leq a_i \leq k$ ).

### Примеры

coloring.in	coloring.out
3	3
3 3	3 2 1
1 2	2
2 3	1 2 2 1 1
3 1	3
5 3	1 3 1 1 2 1
2 1	
3 1	
4 2	
6 7	
1 2	
1 5	
2 5	
2 3	
2 4	
5 6	
5 4	

## Задача С. Обобщенные числа Фибоначчи

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Мы чуть-чуть обобшили для вас последовательность Фибоначчи, теперь:

$$f_1 = f_2 = 1$$

$$f_i = a \cdot f_{i-1} + b \cdot f_{i-2} + c \cdot 2^i + d \cdot i + e, i > 2$$

Дано  $n$ , найдите значение  $f_n$ , взятое по модулю  $10^9 + 7$ .

### Формат входных данных

Неотрицательные целые числа:  $a, b, c, d, e, n$ . ( $0 \leq a, b, c, d, e \leq 10^9$ ;  $1 \leq n \leq 10^{18}$ )

### Формат выходных данных

Выведите  $f_n$ , взятое по модулю  $10^9 + 7$ .

### Примеры

стандартный ввод	стандартный вывод
1 1 0 0 0 8	21
1 2 3 4 5 6	775

## Задача D. Задобрите Инти

Имя входного файла: makeintihappy.in  
Имя выходного файла: makeintihappy.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Как вы помните, племя Майя вымерло, племя Йюля вынуждено было распасться и покинуть обжитые территории, и вот теперь племени Авгуйя тоже угрожает опасность. Задобрав бога дождя Тлалока, они напрочь забыли про бога солнца Инти, который будучи обделённым вниманием грозиться ниспослать испепеляющую жару, если жители племени не порадуют его божественный взор новой коллекцией симпатичных узоров, выложенных из огромных гранитных плит, покрашенных в чёрный и белый цвета. Напомним, что симпатичным узором называется прямоугольник шириной  $W$  и высотой  $H$  ни какой квадрат 2 на 2 которого не покрашен в один цвет.

В данной задаче таскать гранитные плиты Вам не придётся, однако потребуется определить  $K$ -й лексикографически симпатичный узор. Один узор будем считать лексикографически меньше другого, если при просмотре узоров по столбцам сверху вниз найдётся такая позиция, что все плиты рассмотренные ранее совпадают попарно по цвету, а плита на данной позиции у первого узора белая, а у второго — чёрная.

### Формат входных данных

Во входном файле заданы числа  $W$ ,  $H$  и  $K$ . ( $1 \leq W \leq 1000$ ,  $1 \leq H \leq 10$ ,  $1 \leq K \leq 10^{18}$ )

### Формат выходных данных

Выведите требуемый симпатичный узор, обозначая белые плиты символом ' $w$ ', а чёрные — символом ' $b$ '. Если не существует узора с таким номером — выведите "Impossible"(без кавычек).

### Примеры

makeintihappy.in	makeintihappy.out
5 1 17	bwwww
7 4 1234567	wbwbbwb wbbwwwb wwwbbwb wbwbwwwb