

## 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, распространенных 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 входного файла

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 выходного файла

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. Симпатичные узоры

Имя входного файла: `tilings.in`  
Имя выходного файла: `tilings.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Компания BrokenTiles планирует заняться выкладыванием во дворах у состоятельных клиентов узор из черных и белых плиток, каждая из которых имеет размер  $1 \times 1$  метр. Известно, что дворы всех состоятельных людей имеют наиболее модную на сегодня форму прямоугольника  $M \times N$  метров.

Однако при составлении финансового плана у директора этой организации появилось целых две серьезных проблемы: во первых, каждый новый клиент очевидно захочет, чтобы узор, выложенный у него во дворе, отличался от узоров всех остальных клиентов этой фирмы, а во вторых, этот узор должен быть симпатичным.

Как показало исследование, узор является симпатичным, если в нем нигде не встречается квадрата  $2 \times 2$  метра, полностью покрытого плитками одного цвета.

Для составления финансового плана директору необходимо узнать, сколько клиентов он сможет обслужить, прежде чем симпатичные узоры данного размера закончатся. Помогите ему!

## Формат входного файла

На первой строке входного файла находятся два положительных целых числа, разделенные пробелом —  $M$  и  $N$  ( $1 \leq M \times N \leq 30$ ).

## Формат выходного файла

Выведите в выходной файл единственное число — количество различных симпатичных узоров, которые можно выложить во дворе размера  $M \times N$ . Узоры, получающиеся друг из друга сдвигом, поворотом или отражением считаются различными.

### Примеры

tilings.in	tilings.out
1 1	2
1 2	4
4 1	16
2 3	50

### Задача С. Леденящая игра

Имя входного файла: `game.in`  
Имя выходного файла: `game.out`  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 256 мегабайт

Чтобы попасть в команду к Шкиперу пингвин должен пройти ряд испытаний: полоса препятствий от Шкипера, спарринг с Рико, расшифровка кода от Прапора и задача от Ковальски.

Вы, пингвин-новобранец успешно дошли до последнего испытания. Ковальски предлагает вам сыграть в следующую игру. Вам дается  $m$  наборов разноцветных льдинок, каждая одного из  $n$  цветов. Различные цвета обозначаются различными прописными буквами латинского алфавита. Вы можете взять какое-то подмножество этих наборов при условии, что льдинка каждого цвета будет встречаться не более одного раза в этом подмножестве. Пусть вы выбрали  $k$  наборов с индексами  $i_1, i_2, \dots, i_k$ , тогда ваш выигрыш составляет  $\sum_{j=1}^k l_{i_j} - k$  баллов, где  $l_{i_j}$  — количество льдинок в наборе  $i_j$ .

Ковальски требует найти подмножество с максимальным количеством баллов.

От вас требуется найти любое подмножество, подходящее под условия Ковальски.

### Формат входного файла

В первой строке входного файла находится число  $n$  ( $1 \leq n \leq 17$ ) — количество различных цветов. Вторая строка входного файла содержит число  $m$  ( $1 \leq m \leq 200000$ ) — количество различных наборов льдинок. В следующих  $m$  строках перечислены сами наборы. Набор с номером  $i$  задаётся строкой из первых  $n$  строчных латинских букв. Длина каждой строки не больше 17 символов.

### Формат выходного файла

В первой строке выходного файла выведите  $k$  — количество наборов в ответе. Во второй строке выходного файла выведите  $k$  чисел — индексы наборов, входящих в ответ, в произвольном порядке.

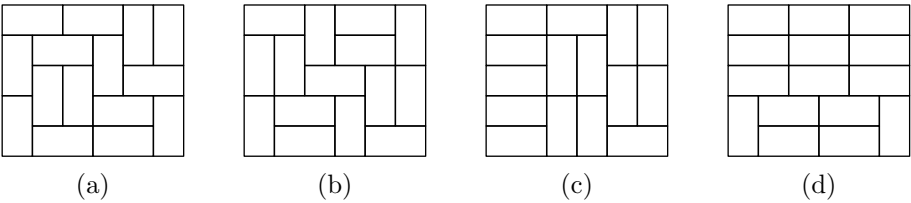
### Примеры

game.in	game.out
1 3 aaa aaaa a	0
1 2 aaa aaaa	0
3 3 aba ab c	1 2

### Задача D. Прочные замощения

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

Замощение прямоугольника  $m \times n$  доминошками  $2 \times 1$  будем называть прочным, если не существует прямой, пересекающей внутренность прямоугольника  $m \times n$  и не пересекающей внутренность ни одной доминошки. Например, приведенные на иллюстрации замощения (a) и (b) — прочные, а замощения (c) и (d) — нет.



А сколько существует прочных замощений прямоугольника  $m \times n$ ?

### Формат входного файла

В первой строке два натуральных числа  $m$  и  $n$  ( $1 \leq m \leq 8; 1 \leq n \leq 16$ ) — ширина и высота прямоугольника.

### Формат выходного файла

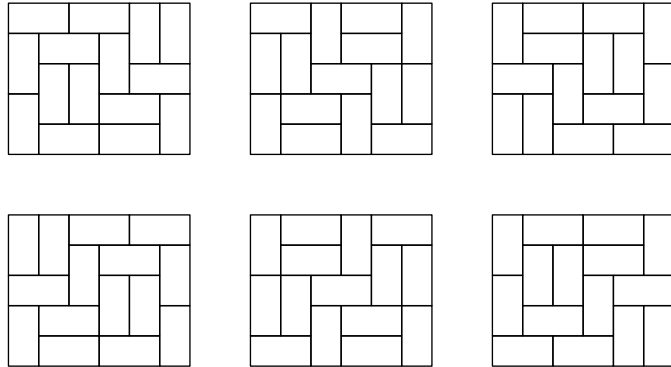
Выведите одно число — количество прочных замощений данного прямоугольника.

### Примеры

<code>solid.in</code>	<code>solid.out</code>
2 2	0
5 6	6

### Note

Приведем все прочные замощения прямоугольника  $5 \times 6$ :



### Задача Е. Длинные домино

Имя входного файла: `longdominoes.in`  
Имя выходного файла: `longdominoes.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Найдите количество способов замостить прямоугольник  $m \times n$  доминошками размерами  $3 \times 1$ . Каждая доминошка должна лежать полностью внутри прямоугольника. Никакие две доминошки не должны накладываться друг на друга. Доминошки могут быть ориентированы как горизонтально, так и вертикально.

#### Формат входного файла

В единственной строке входного файла даны два целых числа  $m$  и  $n$  ( $1 \leq m \leq 9$ ,  $1 \leq n \leq 30$ ).

#### Формат выходного файла

В выходной файл выведите единственное число — количество способов замостить прямоугольник  $n \times m$  доминошками.

### Примеры

<code>longdominoes.in</code>	<code>longdominoes.out</code>
3 3	2
3 10	28