

## Problem A. Грузовики

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 секунда  
Memory limit: 512 мегабайт

Необходимо поместить несколько ящиков в грузовики. Для этого планируется разделить имеющиеся ящики на две одинаковые группы (в случае нечетного количества получаются две группы, в одной из которых на один ящик больше), потом разделить каждую из этих групп аналогичным образом, и так далее, пока мы не получим группы, которые влезают в грузовик. Как только какая-то из получившихся групп влезает в грузовик, производится загрузка ящиков этой группы, и грузовик уезжает. Требуется определить, сколько грузовиков потребуется, чтобы увезти все ящики.

### Input

Во входном файле два числа  $n$  и  $k$  ( $2 \leq n \leq 10\,000, 1 \leq k \leq n - 1$ ) — количество ящиков и емкость грузовика.

### Output

Выведите требуемое количество грузовиков.

### Examples

standard input	standard output
14 3	6
15 1	15
1024 5	256

## Problem B. Сгенерируйте перестановки

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 секунда  
Memory limit: 512 мебибайт

По случаю проведения апрельских интенсивов организаторы решили подарить всем участникам перестановки. В связи с эпидемиологической обстановкой числа в перестановках обязаны соблюдать социальную дистанцию. Это обозначает, что любые два соседних числа должны различаться не менее чем на  $k$ . Помогите организаторам сгенерировать все подходящие перестановки.

### Input

В единственной строке входных данных содержатся два целых числа  $n, k$  ( $0 \leq k \leq n \leq 10$ ) — длина перестановок и максимальная разность между соседними числами.

### Output

Выведите все перестановки в возрастающем лексикографическом порядке. Каждую перестановку следует выводить в отдельной строке. Гарантируется, что размер выходного файла не превосходит 10-и мегабайт.

### Examples

standard input	standard output
4 2	2 4 1 3 3 1 4 2
10 5	5 10 4 9 3 8 2 7 1 6 6 1 7 2 8 3 9 4 10 5
5 2	1 3 5 2 4 1 4 2 5 3 2 4 1 3 5 2 4 1 5 3 2 5 3 1 4 3 1 4 2 5 3 1 5 2 4 3 5 1 4 2 3 5 2 4 1 4 1 3 5 2 4 2 5 1 3 4 2 5 3 1 5 2 4 1 3 5 3 1 4 2

## Problem C. Генерация скобочных последовательностей

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 секунды  
Memory limit: 512 мегабайт

С незапамятных времен хорошей традицией апрельских интенсивов является сжигание правильных скобочных последовательностей. К сожалению, в этом году сборы проходят в онлайн-формате, поэтому было принято решение всего лишь их сгенерировать. Эта обязанность легла на ваши плечи.

### Input

Во входных данных записано единственное целое число  $n$  ( $1 \leq n \leq 14$ ) — количество отрывающих скобок в требуемых правильных скобочных последовательностях.

### Output

Выведите несколько строк длины  $2 \cdot n$ , каждая должна быть правильной скобочной последовательностью. Выводить их требуется в лексикографическом порядке. Считайте, что открывающая скобка лексикографически меньше закрывающей.

### Examples

standard input	standard output
1	()
2	(( )()
3	((() (()) (())( (())( (())()

## Problem D. Различные разбиения

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 секунды  
Memory limit: 512 мегабайт

Найдите количество различных разбиений натурального числа  $n$  на натуральные слагаемые таких, что для любых двух различных чисел  $a \neq b$ , входящих в разбиение, верно, что количества чисел  $a$  и  $b$  в разбиении различны. Разбиения, отличающиеся только порядком слагаемых, различными не считаются.

Например, если  $n = 4$ , то из пяти возможных разбиений этому условию удовлетворяют все, кроме разбиения на слагаемые 1 и 3: в этом разбиении количество единиц равно количеству троек.

4	=	1 + 1 + 1 + 1	4 единицы
4	=	1 + 1 + 2	3 единицы, 1 тройка
4	=	1 + 3	1 единица и 1 тройка!
4	=	2 + 2	2 двойки
4	=	4	1 четвёрка

### Input

В первой строке входного файла записано натуральное число  $n$  ( $1 \leq n \leq 100$ ).

### Output

В первой строке выходного файла выведите количество разбиений числа  $n$ , удовлетворяющих заданным ограничениям.

### Examples

standard input	standard output
4	4
6	7

## Problem E. Разбиение на слагаемые

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 секунда  
Memory limit: 512 мегабайт

Требуется разложить заданное число  $n$  на натуральные слагаемые **всеми** возможными способами.

### Input

В первой строке входных данных записано единственное целое число  $n$  ( $1 \leq n \leq 50$ ) — число, которое вам требуется разложить.

### Output

Выведите все возможные разложения числа  $n$  на натуральные слагаемые, каждое в отдельной строке. Чтобы не запутаться, каждое разложение должно быть отсортировано по неубыванию слагаемых, а сами разложения требуется выводить в лексикографическом порядке.

### Examples

standard input	standard output
3	1 1 1 1 2 3
4	1 1 1 1 1 1 2 1 3 2 2 4
5	1 1 1 1 1 1 1 1 2 1 1 3 1 2 2 1 4 2 3 5

## Problem F. Куча камней

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 секунды  
Memory limit: 512 мегабайт

У вас есть несколько камней известного веса  $w_1, \dots, w_n$ . Напишите программу, которая распределит камни в две кучи так, что разность весов этих двух куч будет минимальной.

### Input

Ввод содержит количество камней  $n$  ( $1 \leq n \leq 20$ ) и веса камней  $w_1, \dots, w_n$  ( $1 \leq w_i \leq 100\,000$ ) – целые, разделённые пробельными символами.

### Output

Ваша программа должна вывести одно число – минимальную разность весов двух куч.

### Examples

standard input	standard output
5 5 8 13 27 14	3

## Problem G. Devil's Hell deLivery

Input file: *standard input*  
 Output file: *standard output*  
 Time limit: 2 секунды  
 Memory limit: 512 мебибайт

Компания Devil's Hell deLivery доставляет буквально всё: ящики, коробки, пакеты, датаграммы и так далее.

Процесс доставки из города А в город В устроен следующим образом. Пусть нужно доставить  $K$  предметов, используя  $N$  грузовиков. Вместимость грузовика с номером  $i$  равна  $c_i$ . Вес предмета с номером  $j$  равен  $w_j$ . Грузовики доставляют предметы за несколько шагов.

За один шаг некоторые предметы целиком загружаются в грузовики. Все предметы можно перевозить только целиком. Вместимость каждого грузовика должна быть не меньше суммарного веса предметов, которые в него загружены. Все грузовики отправляются одновременно.

По окончании шага все грузовики отправляются на исходную позицию. Если какие-то предметы остались не доставленными, предпринимается очередной шаг.

Ваша задача — распределить предметы по шагам и грузовикам так, чтобы минимизировать количество шагов.

### Input

Во вводе задано не более ста тестовых случаев.

Каждый тестовый случай начинается со строки с двумя целыми числами  $N$  и  $K$  ( $1 \leq N \leq 5$ ,  $1 \leq K \leq 9$ ): количеством грузовиков и количеством предметов, соответственно. В следующей строке записано  $N$  целых чисел  $c_1, \dots, c_N$  ( $1 \leq c_i \leq 10^8$ ): вместимости грузовиков. В следующей строке записано  $K$  целых чисел  $w_1, \dots, w_K$  ( $1 \leq w_i \leq 10^8$ ): веса предметов.

### Output

Для каждого тестового случая, если доставка невозможна, выведите единственную строку с числом  $-1$ .

В остальных случаях сначала выведите строку с целым числом  $S$  — необходимым количеством шагов. Это число нужно минимизировать. Затем выведите  $S$  строк, по одной на каждый шаг. Описание шага должно начинаться с целого числа  $I_i$  — количества предметов, которые перевозятся на этом шаге. Далее запишите  $I_i$  пар  $a_j b_j$ . Каждая пара  $a_j b_j$  означает, что предмет  $a_j$  перевозится на грузовике  $b_j$ .

Если возможных оптимальных ответов несколько, выведите любой из них.

### Example

standard input	standard output
2 4	1
10 20	4 1 1 2 1 3 2 4 2
5 5 5 5	-1
2 1	
10 10	
20	