

Задача А. Ханойские башни

Имя входного файла: `hanoi.in`
Имя выходного файла: `hanoi.out`
Ограничение по времени: 6 секунды
Ограничение по памяти: 64 мегабайта

Даны три стержня. На первом стержне находятся несколько дисков сверху вниз по возрастанию размера диска. Два другие пустые. Требуется перенести все диски с первого стержня на второй. Переносить диски разрешается только по одному. Не разрешается класть больший диск на меньший.

Формат входных данных

Вводится одно число n ($1 \leq n \leq 19$) — количество дисков на первом стержне.

Формат выходных данных

Выведите по два числа в строке — номера стержней, откуда и куда переносится диск. Решение должно быть оптимальным по количеству действий.

Примеры

<code>hanoi.in</code>	<code>hanoi.out</code>
3	1 2 1 3 2 3 1 2 3 1 3 2 1 2

Замечание

В программе нельзя пользоваться циклами.

Задача В. Монетки

Имя входного файла: `coins.in`
Имя выходного файла: `coins.out`
Ограничение по времени: 2.2 секунды
Ограничение по памяти: 64 мегабайта

В Волшебной стране используются монетки достоинством $A_1, A_2, \dots, A_M$. Волшебный человечек пришел в магазин и обнаружил, что у него есть ровно по две монетки каждого достоинства. Ему нужно заплатить сумму N . Напишите программу, определяющую, сможет ли он расплатиться без сдачи.

Формат входных данных

Сначала вводится целое число N ($1 \leq N \leq 10^9$), затем — целое число M ($1 \leq M \leq 10$) и далее M попарно различных целых чисел $A_1, A_2, \dots, A_M$ ($1 \leq A_i \leq 10^9$).

Формат выходных данных

Выведите сначала K — количество монет, которое придется отдать Волшебному человечку, если он сможет заплатить указанную сумму без сдачи. Далее выведите K чисел, задающих достоинства монет. Если решений несколько, выведите вариант, в котором Волшебный человек отдаст наименьшее возможное количество монет. Если таких вариантов несколько, выведите любой из них.

Если без сдачи не обойтись, то выведите одно число 0. Если же у Волшебного человечка не хватит денег, чтобы заплатить указанную сумму, выведите одно число -1 (минус один).

Примеры

coins.in	coins.out
5 2 1 2	3 1 2 2
7 2 1 2	-1
5 2 3 4	0

Задача С. Перестановки

Имя входного файла: `permutations.in`
Имя выходного файла: `permutations.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Выведите все перестановки чисел от 1 до N в лексикографическом порядке.

Формат входных данных

Во входном файле задано число N ($1 \leq N \leq 8$).

Формат выходных данных

Выведите все перестановки, разделяя числа пробелами, по одной в каждой строке.

Примеры

<code>permutations.in</code>	<code>permutations.out</code>
3	1 2 3 1 3 2 2 1 3 2 3 1 3 1 2 3 2 1

Задача D. Очередь

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

Реализуйте работу очереди. Для каждой операции изъятия элемента выведите ее результат.

На вход программе подаются строки, содержащие команды. Каждая строка содержит одну команду. Команда — это либо “+ N”, либо “-”. Команда “+ N” означает добавление в очередь числа N , по модулю не превышающего 10^9 . Команда “-” означает изъятие элемента из очереди.

Формат входных данных

В первой строке содержится количество команд — M ($1 \leq M \leq 10^6$). В последующих строках содержатся команды, по одной в каждой строке.

Формат выходных данных

Выведите числа, которые удаляются из очереди, по одному в каждой строке. Гарантируется, что изъятий из пустой очереди не производится.

Примеры

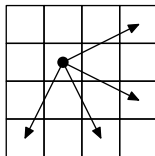
queue.in	queue.out
4	1
+ 1	10
+ 10	
-	
-	

Задача Е. Ход конём - 2

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Дана прямоугольная доска $N \times M$ (N строк и M столбцов). В левом верхнем углу находится шахматный конь, которого необходимо переместить в правый нижний угол доски.

При этом конь может ходить следующим образом:



Необходимо определить, сколько существует различных маршрутов, ведущих из левого верхнего в правый нижний угол.

Формат входных данных

Входной файл содержит два натуральных числа N и M ($1 \leq N, M \leq 50$).

Формат выходных данных

В выходной файл выведите единственное число — количество способов добраться конём до правого нижнего угла доски.

Примеры

stdin	stdout
4 4	2
2 3	1