

## Задача А. Кузнечик

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

У одного из преподавателей параллели С в комнате живёт кузнечик, который очень любит прыгать по клетчатой одномерной доске. Длина доски —  $N$  клеток. К его сожалению он умеет прыгать только на  $1, 2, \dots, k$  клеток вперёд.

Однажды преподавателям стало интересно, сколькими способами кузнечик может допрыгать из первой клетки до последней. Помогите им ответить на этот вопрос.

### Формат входного файла

В первой и единственной строке входного файла записано два целых числа —  $N$  и  $k$  ( $1 \leq N \leq 30, 1 \leq k \leq 10$ ).

### Формат выходного файла

Выведите одно число — количество способов, которыми кузнечик может допрыгать из первой клетки до последней.

### Примеры

stdin	stdout
8 2	21

## Задача В. Рюкзак

Имя входного файла: **stdin**  
Имя выходного файла: **stdout**  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 16 мегабайт

Найдите максимальный вес золота, который можно унести в рюкзаке вместительностью  $S$ , если есть  $N$  золотых слитков с заданными весами.

### Формат входного файла

В первой строке входного файла записаны два числа —  $S$  и  $N$  ( $1 \leq S \leq 10\,000, 1 \leq N \leq 300$ ). Далее следует  $N$  неотрицательных целых чисел, не превосходящих  $100\,000$  — веса слитков.

### Формат выходного файла

Выведите искомый максимальный вес.

### Пример

stdin	stdout
10 3 5 7 4	9

## Задача С. Черепахоконь

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

Дана клетчатая доска размером  $N \times M$  в каждой клетке которой записано натуральное число. В верхнем левом углу доски сидит черепашка. Черепашка умеет делать ход конём по направлению вниз и вправо. То есть либо перемещаться на одну клетку вправо и на две вниз, либо на одну клетку вниз и на две вправо. Помогите черепашке добраться в правый нижний угол доски, собрав максимальную сумму чисел. Считается, что черепашка собирает только те числа, на которых завершает ход, а не все, по которым проползает.

### Формат входного файла

В первой строке входного файла два целых числа  $N$  и  $M$  ( $1 \leq N, M \leq 100$ ), задающие размеры доски. Далее следуют числа, записанные на доске —  $N$  строк по  $M$  положительных чисел, не превышающих  $10\,000$ , в каждой.

### Формат выходного файла

Выведите одно число, равное искомой максимальной сумме, либо -1, если черепашка не может добраться до правого нижнего угла.

### Примеры

stdin	stdout
2 3 3 2 7 1 9 5	8

## Задача D. 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 входного файла

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

### Формат выходного файла

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

## Примеры

stdin	stdout
4 4	2
15 14	7884330

## Задача Е. Лесенка

Имя входного файла: **stdin**  
Имя выходного файла: **stdout**  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 16 мегабайт

На каждой из  $N + 2$  ступенек лестницы записано целое число, причем на первой и последней ступеньке записано число 0. На первой ступеньке стоит человек, которому необходимо подняться на последнюю ступеньку. За один шаг он может подниматься на любое число ступенек, не превосходящее  $K$ .

Подсчитаем сумму всех чисел, написанных на ступеньках, на которые наступил человек. Найдите наибольшее возможное значение этой суммы.

### Формат входного файла

В первой строке входного файла записано число  $N$ ,  $0 \leq N \leq 1000$ . Во второй строке записано  $N$  целых чисел, не превосходящих по модулю 1000, разделенных пробелами — числа, записанные на ступеньках (за исключением первой и последней ступеньки, на которых записаны нули). В третьей строке записана максимальная величина шага человека  $K$ ,  $1 \leq K \leq N$ .

### Формат выходного файла

Выведите максимально возможную сумму чисел, записанных на ступеньках, на которые наступил человек.

### Пример

stdin	stdout
3 1 -1 1 2	2

## Задача F. Нескучная последовательность

Имя входного файла: **stdin**  
Имя выходного файла: **stdout**  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 16 мегабайт

Числовую последовательность назовем *нескучной*, если в этой последовательности идет не более двух одинаковых элементов подряд. По данным числам  $N$  и  $K$  определите количество нескучных последовательностей длины  $N$ , составленных из натуральных чисел от 1 до  $K$ .

## Формат входного файла

В единственной строке находятся два числа  $N$  и  $K$  ( $1 \leq N \leq 20$ ,  $1 \leq K \leq 10$ ).

## Формат выходного файла

Нужно вывести одно число — количество искомых последовательностей. Гарантируется, что это число меньше  $2^{31}$ .

### Пример

stdin	stdout
3 3	24

## Задача G. Переворот

Имя входного файла: **stdin**  
Имя выходного файла: **stdout**  
Ограничение по времени: 85 мгновений  
Ограничение по памяти: 64 терабайта

Дано натуральное число  $N$  и последовательность из  $N$  элементов. Требуется вывести эту последовательность в обратном порядке.

Примечание. В программе запрещается объявлять массивы и использовать циклы (даже для ввода).

### Формат входного файла

В первой строке входного файла записано натуральное число  $N$  ( $N \leq 10^{12}$ ). Во второй строке через пробел идут  $N$  целых чисел, по модулю не превосходящих  $1^{13}$ , — элементы последовательности.

### Формат выходного файла

В выходной файл выведите заданную последовательность в обратном порядке.

### Пример

stdin	stdout
3 0 1 0	0 1 0