

Задача А. Пути на доске

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

Рассмотрим бесконечную клетчатую доску.

Назовём *путём* из одной клетки в другую последовательность клеток, в которой каждые две идущие подряд клетки являются соседними по стороне. Длина пути — это количество клеток в нём, не считая начальную.

Назовём путь *простым*, если в нём не встречается двух одинаковых клеток.

Зафиксируем какую-то клетку на доске. Сколько существует простых путей заданной длины, начинающихся в этой клетке?

Формат входного файла

В первой строке входного файла задано целое число n ($0 \leq n \leq 22$).

Формат выходного файла

В первой строке выходного файла выведите одно число — количество путей длины n из этой клетки.

Примеры

<code>paths.in</code>	<code>paths.out</code>
0	1
1	4
2	12

Задача В. Функция

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

Вычислите функцию:
$$f(n) = \begin{cases} 1 & \text{если } n \leq 2 \\ f(\lfloor 6 * n / 7 \rfloor) + f(\lfloor 2 * n / 3 \rfloor) & \text{если } n \bmod 2 = 1 \\ f(n - 1) + f(n - 3) & \text{если } n \bmod 2 = 0 \end{cases}$$

Формат входного файла

Входные данные содержат натуральное число n ($1 \leq n \leq 10^{12}$).

Формат выходного файла

Выведите значение функции по модулю 2^{32} .

Пример

<code>function.in</code>	<code>function.out</code>
7	10

Задача С. Кодовый замок

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

Петя опоздал на тренировку по программированию! Поскольку тренировка проходит в воскресенье, главный вход в учебный корпус, где она проходит, оказался закрыт, а вахтёр ушёл куда-то по своим делам. К счастью, есть другой способ проникнуть в здание — открыть снаружи боковую дверь, на которой установлен кодовый замок.

На пульте замка есть d кнопок с цифрами $0, 1, \dots, d - 1$. Известно, что код, открывающий замок, состоит из k цифр. Замок открывается, если последние k нажатий кнопок образуют код.

Поскольку Петя не имеет понятия, какой код открывает замок, ему придётся перебрать все возможные коды из k цифр. Но, чтобы как можно скорее попасть на тренировку, нужно минимизировать количество нажатий на кнопки. Помогите Пете придумать такую последовательность нажатий на кнопки, при которой все возможные коды были бы проверены, а количество нажатий при этом оказалось бы минимально возможным.

Формат входного файла

В первой строке входного файла записаны через пробел два целых числа d и k — количество кнопок на пульте и размер кода, соответственно ($2 \leq d \leq 10$, $1 \leq k \leq 20$).

Формат выходного файла

В первой строке выходного файла выведите искомую последовательность. Если последовательностей минимальной длины, перебирающих все возможные коды, несколько, можно выводить любую из них. Гарантируется, что d и k таковы, что минимальная длина последовательности не превосходит 1 мегабайта.

Пример

<code>codelock.in</code>	<code>codelock.out</code>
2 3	0001011100

Последовательность в примере перебирает все коды длины 3 в следующем порядке: 000, 001, 010, 101, 011, 111, 110, 100.

Задача D. Сортировка вручную

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

Пришло время расставить книжки на полке. На ней стоят n книг, однозначно пронумерованных целыми числами от 1 до n . Требуется расставить их в порядке возрастания номеров. Вы прекрасно знаете, что быстрая сортировка и сортировка вставками — быстрые методы сортировки, однако их непросто исполнять вручную. Они эффективны для компьютеров, а не для людей.

Поэтому вы решили сортировать книги, вставляя i -ю книгу на i -ю позицию. Сколько операций вам придётся выполнить для успешной сортировки?

Формат входного файла

В первой строке входного файла содержится целое число n ($1 \leq n \leq 15$) — количество книг на полке.

Во второй строке записано n различных целых чисел от 1 до n — номера книг в том порядке, в котором они расставлены вначале.

Формат выходного файла

Выведите единственное целое число — минимальное количество операций указанного вида, которое придётся выполнить для сортировки.

Примеры

<code>sort.in</code>	<code>sort.out</code>
3 1 2 3	0
3 2 1 3	1
3 3 2 1	2
9 7 2 9 4 6 8 5 3 1	5