1

Последовательности из 0 и 1 без k одинаковых символов подряд

2

Количество правильных скобочных последовательностей

3

Плавные числа

4

Числа

5

Рюкзак с восстановлением ответа

адача 1.

Лёгкая

**Последовательности из 0 и 1 без k одинаковых символов подряд**

По данным натуральным n и k определите количество последовательностей длины n из 0 и 1, не содержащих k одинаковых символов подряд. n≤106,2≤k≤n+1. Гарантируется, что ответ не превосходит 231−1.

ВХОДНЫЕ ДАННЫЕ

Вводятся два натуральных числа n и k.

ВЫХОДНЫЕ ДАННЫЕ

Выведите ответ на задачу.

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 3 3 | 6 |

Задача 2.

Лёгкая

**Количество правильных скобочных последовательностей**

По данному натуральному **n** определите количество правильных скобочных последовательностей, составленных из **n** открывающихся и **n** закрывающихся круглых скобок.

## **Входные данные**

Программа получает на вход натуральное число **n**, не превосходящее 1000.

## **Выходные данные**

Необходимо вывести остаток от деления числа искомых последовательностей на 109+7.

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 3 | 5 |

Начало формы

Задача решена верно

OK. Your score is = 100, 25/25 tests passed

Решение задачи

Первый параметр динамики - длина уже построенной последовательности. Второй - текущий баланс последовательности. Закрывающую можно ставить когда баланс больше нуля. Открывающую - когда мы успеем закрыть текущий баланс, то есть оставшегося места после постановки этой скобки не меньше чем текущий баланс плюс 1.

Конец формы

Задача 3.

Лёгкая

**Плавные числа**

**Ограничение по времени работы программы: 1 секунда**

Назовем число плавным, если его две соседние цифры различаются не более, чем на 1. По данному натуральному n определите количество плавных натуральных чисел, имеющих длину n.

Обратите внимание, что запись натурального числа не может начинаться с цифры 0.

ВХОДНЫЕ ДАННЫЕ

Программа получает на вход число n⩽19.

ВЫХОДНЫЕ ДАННЫЕ

Программа должна вывести количество плавных чисел, состоящих из n цифр.

ПРИМЕР

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 1 | 9 |
| 2 | 26 |

Начало формы

Задача решена верно

OK. Your score is = 100, 19/19 tests passed

Решение задачи

Решим задачу при помощи динамического программирования. Целевая функция: F(k,n) - количество плавных чисел длины n, заканчивающихся на цифру k (0⩽k⩽9).

Рекуррентные соотношения:

F(k,n)=F(k−1,n−1)+F(k,n−1)+F(k+1,n−1). Случай k=0 и k=9 нужно обрабатывать отдельно.

Начальные значения - F(k,1)=1 при k≠0, F(0,1)=0.

n = int(input())  
F = [[0] \* (n + 1) for i in range(10)]  
for i in range(1, 10):  
    F[i][1] = 1  
for i in range(2, n + 1):  
    for d in range(10):  
        F[d][i] = F[d][i - 1]  
        if d > 0:  
            F[d][i] += F[d - 1][i - 1]  
        if d < 9:  
            F[d][i] += F[d + 1][i - 1]  
s = 0  
for i in range(10):  
    s += F[i][n]  
print(s)

**Подсказка**

Используйте метод динамического программирования. Количество продолжений уже построенного префикса числа зависит от того, на какую цифру заканчивается префикс.

Целевая функция: F(k,n) - количество плавных чисел длины n, заканчивающихся на цифру k (0⩽k⩽9).

**Подсказка**

Рекуррентные соотношения:

F(k,n)=F(k−1,n−1)+F(k,n−1)+F(k+1,n−1). Случай k=0 и k=9 нужно обрабатывать отдельно.

Начальные значения - F(k,1)=1 при k≠0, F(0,1)=0.

Конец формы

Задача 4.

Сложная

**Числа**

Решая задачу по информатике, Вова в очередной раз допустил ошибку. Он снова вывел в выходной файл числа, забыв разделить их пробелами. Увидев полученный результат, Вова сначала огорчился, а потом задумался над следующим вопросом: сколько существует различных последовательностей неотрицательных целых чисел, таких что, если выписать их без пробелов, то получится тот же результат, что и у него. Он вспомнил также, что его программа смогла вывести не произвольные числа, а только те, что не превосходят C и не имеют ведущих нулей. Чтобы ответить на поставленный вопрос, Вова решил написать программу, которая позволит ему найти число различных последовательностей неотрицательных целых чисел, в каждой из которых любое число не превосходит C. Он понимал, что такое число могло быть достаточно большим, поэтому ограничился поиском только последних k цифр этого числа. Требуется написать программу, которая покажет Вове, как можно правильно решить поставленную им задачу.

ВХОДНЫЕ ДАННЫЕ

Первая строка входного файла содержит три целых числа — n,C и  k (1≤n≤50000,1≤C≤108,1≤k≤18). Во второй строке этого файла содержится результат работы Вовиной программы, состоящий из n цифр.

ВЫХОДНЫЕ ДАННЫЕ

В выходной файл выведите последние k цифр искомого количества последовательностей (без ведущих нулей).

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 3 11 2  111 | 3 |
| 19 9 1  0123456789876543210 | 1 |
| 1 8 3  9 | 0 |

Задача 5.

Сложная

**Рюкзак с восстановлением ответа**

Дано *N* предметов массой *m*1, …, *mN* и стоимостью *c*1, …, *cN* соответственно.

Ими наполняют рюкзак, который выдерживает вес не более *M*. Определите набор предметов, который можно унести в рюкзаке, имеющий наибольшую стоимость.

Входные данные

В первой строке вводится натуральное число *N*, не превышающее 100 и натуральное число *M*, не превышающее 10000.

Во второй строке вводятся *N* натуральных чисел *mi*, не превышающих 100.

В третьей строке вводятся *N* натуральных чисел *сi*, не превышающих 100.

Выходные данные

Выведите номера предметов (числа от 1 до *N*), которые войдут в рюкзак наибольшей стоимости.

Примеры

входные данные

4 6

2 4 1 2

7 2 5 1

выходные данные

1

3

4