

## А. Кузнечик собирает монеты

2 секунды, 256 мегабайт

Кузнечик прыгает по столбикам, расположенным на одной линии на равных расстояниях друг от друга. Столбики имеют порядковые номера от 1 до  $n$ . В начале Кузнечик сидит на столбике с номером 1 и хочет добраться до столбика с номером  $n$ . Он может прыгнуть вперед на расстояние от 1 до  $k$  столбиков, считая от текущего.

На каждом столбике Кузнечик может получить или потерять несколько золотых монет (для каждого столбика это число известно). Определите, как нужно прыгать Кузнечику, чтобы собрать наибольшее количество золотых монет. Учитывайте, что Кузнечик не может прыгать назад.

### Входные данные

В первой строке вводятся два натуральных числа:  $n$  и  $k$  ( $3 \leq n \leq 10\,000$ ,  $1 \leq k \leq 10\,000$ ), разделённые пробелом. Во второй строке записаны через пробел  $n - 2$  целых числа — количество монет, которое Кузнечик получает на каждом столбике, от 2-го до  $n - 1$ -го. Если это число отрицательное, Кузнечик теряет монеты. Гарантируется, что все числа по модулю не превосходят 10 000.

### Выходные данные

В первой строке программа должна вывести наибольшее количество монет, которое может собрать Кузнечик. Во второй строке выводится число прыжков Кузнечика, а в третьей строке — номера всех столбиков, которые посетил Кузнечик (через пробел в порядке возрастания).

Если правильных ответов несколько, выведите любой из них.

входные данные
5 3 2 -3 5
выходные данные
7 3 1 2 4 5

входные данные
10 3 -13 -2 -14 -124 -9 -6 -5 -7
выходные данные
-16 4 1 3 6 8 10

входные данные
12 5 -5 -4 -3 -2 -1 1 2 3 4 5
выходные данные
14 7 1 6 7 8 9 10 11 12

## В. Черепаха и монеты

2 секунды, 256 мегабайт

Черепаха хочет переползти из левого верхнего угла поля размером  $n$  на  $m$  клеток ( $2 \leq n, m \leq 1000$ ) в правый нижний. За один шаг она может переместиться на соседнюю клетку вправо или на соседнюю клетку вниз. Кроме того, проходя через каждую клетку, Черепаха получает (или теряет) несколько золотых монет (это число известно для каждой клетки).

Определите, какое максимальное количество монет может собрать Черепаха по пути и как ей нужно идти для этого.

### Входные данные

В первой строке вводятся два натуральных числа:  $n$  и  $m$  ( $2 \leq n, m \leq 1000$ ), разделённые пробелом. В каждой из следующих  $n$  строк записаны через пробел по  $m$  чисел  $a_{ij}$  ( $|a_{ij}| \leq 10$ ), которые обозначают количество монет, получаемых Черепашкой при проходе через каждую клетку. Если это число отрицательное, Черепашка теряет монеты.

### Выходные данные

В первой строке программа должна вывести наибольшее количество монет, которое может собрать Черепаха. Во второй строке без пробелов выводятся команды, которые нужно выполнить Черепашке: буква 'R' (от слова right) обозначает шаг вправо, а буква 'D' (от слова down) — шаг вниз.

входные данные
3 3 0 2 -3 2 -5 7 1 2 0
выходные данные
6 RRDD

входные данные
4 5 4 5 3 2 9 4 6 7 5 9 5 2 5 -3 -10 3 5 2 9 3
выходные данные
41 RDRDDRR

## С. Наибольшая возрастающая подпоследовательность

2 секунды, 256 мегабайт

Пусть  $a_1, a_2, \dots, a_n$  — числовая последовательность. Длина последовательности — это количество элементов этой последовательности. Последовательность  $a_{i_1}, a_{i_2}, \dots, a_{i_k}$  называется подпоследовательностью последовательности  $a$ , если  $1 \leq i_1 < i_2 < \dots < i_k \leq n$ . Последовательность  $a$  называется возрастающей, если  $a_1 < a_2 < \dots < a_n$ .

Вам дана последовательность, содержащая  $n$  целых чисел. Найдите ее самую длинную возрастающую подпоследовательность.

### Входные данные

В первой строке задано одно число  $n$  ( $1 \leq n \leq 2000$ ) — длина подпоследовательности. В следующей строке задано  $n$  целых чисел  $a_i$  ( $-10^9 \leq a_i \leq 10^9$ ) — элементы последовательности.

#### Входные данные

В первой строке выведите число  $k$  — длину наибольшей возрастающей подпоследовательности. В следующей строке выведите  $k$  чисел — саму подпоследовательность.

входные данные
8 1 4 1 5 3 3 4 2
выходные данные
3 1 4 5

входные данные
3 1 2 3
выходные данные
3 1 2 3

### D. Ход конем

1 секунда, 256 мегабайт

Шахматная ассоциация решила оснастить всех своих сотрудников такими телефонными номерами, которые бы набирались на кнопочном телефоне ходом коня. Например, ходом коня набирается телефон 340-49-27. При этом телефонный номер не может начинаться ни с цифры 0, ни с цифры 8.

Напишите программу, определяющую количество телефонных номеров длины  $n$ , набираемых ходом коня. Поскольку таких номеров может быть очень много, выведите ответ по модулю  $10^9$ .

#### Входные данные

Во входном файле записано целое число  $n$  ( $1 \leq n \leq 100$ ).

#### Выходные данные

Выведите в выходной файл искомое количество телефонных номеров по модулю  $10^9$ .

входные данные
1
выходные данные
8

входные данные
2
выходные данные
16

### E. Расстояние по Левенштейну

2 секунды, 256 мегабайт

Дана текстовая строка. С ней можно выполнять следующие операции:

1. Заменить один символ строки на другой символ.
2. Удалить один произвольный символ.
3. Вставить произвольный символ в произвольное место строки.

Например, при помощи первой операции из строки «СОК» можно получить строку «СУК», при помощи второй операции — строку «ОК», при помощи третьей операции — строку «СТОК».

Минимальное количество таких операций, при помощи которых можно из одной строки получить другую, называется стоимостью редактирования или расстоянием Левенштейна.

Определите расстояние Левенштейна для двух данных строк.

#### Входные данные

Программа получает на вход две строки, длина каждой из которых не превосходит 1000 символов, строки состоят только из заглавных латинских букв.

#### Выходные данные

Требуется вывести одно число — расстояние Левенштейна для данных строк.

входные данные
ABCDEFGH ACDEGHI
выходные данные
3

### F. Покупка билетов

1 секунда, 256 мегабайт

За билетами на премьеру нового мюзикла выстроилась очередь из  $N$  человек, каждый из которых хочет купить  $1$  билет. На всю очередь работала только одна касса, поэтому продажа билетов шла очень медленно, приводя «постояльцев» очереди в отчаяние. Самые сообразительные быстро заметили, что, как правило, несколько билетов в одни руки кассир продаёт быстрее, чем когда эти же билеты продаются по одному. Поэтому они предложили нескольким подряд стоящим людям отдавать деньги первому из них, чтобы он купил билеты на всех.

Однако для борьбы со спекулянтами кассир продавала не более  $3$ -х билетов в одни руки, поэтому договориться таким образом между собой могли лишь  $2$  или  $3$  подряд стоящих человека.

Известно, что на продажу  $i$ -му человеку из очереди одного билета кассир тратит  $A_i$  секунд, на продажу двух билетов —  $B_i$  секунд, трех билетов —  $C_i$  секунд. Напишите программу, которая подсчитает минимальное время, за которое могли быть обслужены все покупатели.

Обратите внимание, что билеты на группу объединившихся людей всегда покупает первый из них. Также никто в целях ускорения не покупает лишних билетов (то есть билетов, которые никому не нужны).

#### Входные данные



Во входном файле записано сначала число  $N$  — количество покупателей в очереди ( $1 \leq N \leq 5000$ ). Далее идет  $N$  троек натуральных чисел  $A_i, B_i, C_i$ . Каждое из этих чисел не превышает 3600. Люди в очереди нумеруются начиная от кассы.

**Выходные данные**

В выходной файл выведите одно число — минимальное время в секундах, за которое могли быть обслужены все покупатели.

входные данные
5 5 10 15 2 10 15 5 5 5 20 20 1 20 1 1
выходные данные
12

выходные данные
260 0 2 3 5

входные данные
3 110 110 110

выходные данные
220 1 1 2

**G. Кафе**

2 секунды, 64 мегабайта

Около Петиного университета недавно открылось новое кафе, в котором действует следующая система скидок: при каждой покупке более чем на 100 рублей покупатель получает купон, дающий право на один бесплатный обед (при покупке на сумму 100 рублей и меньше такой купон покупатель не получает).

Однажды Пете на глаза попался прейскурант на ближайшие  $n$  дней. Внимательно его изучив, он решил, что будет обедать в этом кафе все  $n$  дней, причем каждый день он будет покупать в кафе ровно один обед. Однако стипендия у Пети небольшая, и поэтому он хочет по максимуму использовать предоставляемую систему скидок так, чтобы его суммарные затраты были минимальны. Требуется найти минимально возможную суммарную стоимость обедов и номера дней, в которые Пете следует воспользоваться купонами.

**Входные данные**

В первой строке входного файла записано целое число  $n$  ( $0 \leq n \leq 100$ ). В каждой из последующих  $n$  строк записано одно целое число, обозначающее стоимость обеда в рублях на соответствующий день. Стоимость — неотрицательное целое число, не превосходящее 300.

**Выходные данные**

В первой строке выдайте минимальную возможную суммарную стоимость обедов. Во второй строке выдайте два числа  $k_1$  и  $k_2$  — количество купонов, которые останутся неиспользованными у Пети после этих  $n$  дней и количество использованных им купонов соответственно.

В последующих  $k_2$  строках выдайте в возрастающем порядке номера дней, когда Пете следует воспользоваться купонами. Если существует несколько решений с минимальной суммарной стоимостью, то выдайте то из них, в котором значение  $k_1$  максимально (на случай, если Петя когда-нибудь ещё решит заглянуть в это кафе). Если таких решений несколько, выведите любое из них.

входные данные
5 110 40 120 110 60