Задача 5. Три сына

Имя входного файла: division . in

Имя выходного файла: division. out Ограничение по времени: 1 секунда Ограничение по памяти: 256 мегабайт

Во владениях короля Флатландии находится прямая дорога длиной п километров, по одну сторону от которой расположен огромный лесной массив. Король Флатландии проникся идеями защиты природы и решил превратить свой лесной массив в заповедник. Но сыновья стали сопротивляться: ведь им хотелось получить эти земли в наследство.

У короля три сына: младший, средний и старший. Король решил, что в заповедник не войдут участки лесного массива, которые он оставит сыновьям в наследство. При составлении завещания король хочет, чтобы для участков выполнялись следующие условия:

* каждый участок должен иметь форму квадрата, длина стороны которого выражается целым положительным числом. Одна из сторон каждого квадрата должна лежать на дороге. Пусть участки имеют размеры *а а, b Х b* и с с;
* стороны квадратов должны полностью покрывать дорогу: величина *а + b + с*

должна быть равна u;

* участок младшего сына должен быть строго меньше участка среднего сына, а участок среднего сына должен, в свою очередь, быть строго меньше участка старшего сына, то есть должно выполняться неравенство *а < b < с;*
* суммарная площадь участков *а’ + b’ + c2* должна быть минимальна.

Требуется написать программу, которая по заданной длине дороги определяет

размеры участков, которые следует выделить сыновьям короля.

Формат вхоdлосо *файла*

Входной файл содержит одно целое число п (6 < u < 109).

Формат выхоdлоао *файла*

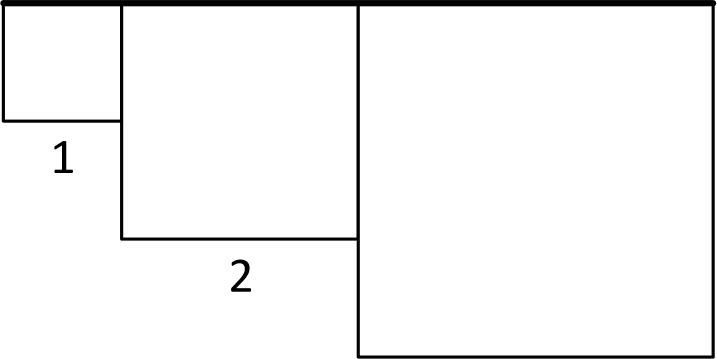
Выходной файл должен содержать три целых положительных числа, разделенных пробелами: п, *b н с —* длины сторон участков, которые следует выделить младшему, среднему и старшему сыну, соответственно. Если оптимальных решений несколько, разрешается вывести любое.

### *Пример входных и выходных файлов*

|  |  |
| --- | --- |
| **division.in** | **division.out** |
| 6 | 1 2 3 |

/7ояслелье х л£\*•меру

6



### *Описание подзадач и системы оценивания*

В этой задаче четыре подзадачи. Баллы за подзадачу начисляются только в случае, если все тесты для данной подзадачи пройдены.

*Подзадача 1 (25 баллов)*

п 50.

Подзадача *2 (25 баллов)*

< 2000.

###### *Подзадача 3 (25 баллов)*

л 40 000.

Пodзadaчa 4 *(25* баллов)

< 109.

*Получение информации о результатах окончательной проверки*

По запросу сообщается результат окончательной проверки на каждом тесте.

**Задача 6. Гипершашки**

Имя входного файла: qame . in

Имя выходного файла: qame . out Ограничение по времени: 1 секунда Ограничение по памяти: 256 мегабайт

Андрей работает судьей на чемпионате по гипершашкам. В каждой игре в гипершашки участвует три игрока. По ходу игры каждый из игроков набирает некоторое положительное целое число баллов. Если после окончания игры первый игрок набрал *а* баллов, второй — *b,* а третий г, то говорят, что игра закончилась со счетом *a:b:c.*

Андрей знает, что правила игры гипершашек устроены таким образом, что в результате игры баллы любых двух игроков различаются не более чем в *k* раз.

После матча Андрей показывает его результат, размещая три карточки с очками игроков на специальном табло. Для этого у него есть набор из п карточек, на которых написаны числа , <з› • •› л- Чтoбы выяснить, насколько он готов к чемпионату, Андрей хочет понять, сколько различных вариантов счета он сможет показать на табло, используя имеющиеся карточки.

Требуется написать программу, которая по числу *k* и значениям чисел на карточках, которые имеются у Андрея, определяет количество различных вариантов счета, которые Андрей может показать на табло.

Формат вхоdлосо *файла*

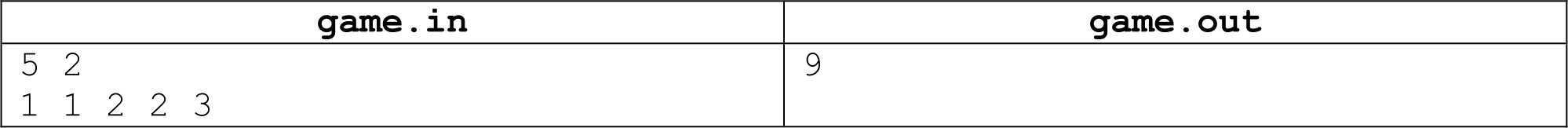
Первая строка входного файла содержит два целых числа: u и k (3 u 100 000,

l *< k <* 109).

Вторая строка входного файла содержит п целых чисел , , ., х, (1 х, 109).

#### *Формат выходного файла*

Выходной файл должен содержать одно целое число — искомое количество различных вариантов счета.

/7рьмер вхоdлых п выхоdлых *файлов*

Поясиеиие к примеру

В приведенном примере Андрей сможет показать следующие варианты счета: 1:1:2, 1:2:1, 2:1:1, 1:2:2, 2:1:2, 2:2:1, 2:2:3, 2:3:2, 3:2:2. Другие тройки чисел, которые можно составить с использованием имеющихся карточек, не удовлетворяют заданному условию, что баллы любых двух игроков различаются не более чем в *k ——* 2 раза.

### *Описание подзадач и системы оценивания*

В этой задаче четыре подзадачи. Баллы за подзадачу начисляются только в случае, если все тесты для данной подзадачи пройдены.

**Внимание!** Тест из примера не подходит под ограничения для подзадач 1 и 3, но решение принимается на проверку только в том случае, если оно выводит правильный ответ на тесте из примера. Решение должно выводить правильный ответ на тест, даже если оно рассчитано на решение только каких-либо из подзадач 1 и 3.

|  |  |  |  |
| --- | --- | --- | --- |
| *Подзадача* | *1 (15 баллов)* |  | |
| 3 п | 100 000, *k ——* 1, 1 | х, | 100 000 |

Подзадача *2 (23 балла)*

3 < < 100, 1 < I < 100, 1 < х < 100

###### *Подзадача 3 (30 баллов)*

3 п 100 000, 1 *k \_<* 109, 1 х; 109, все х, различны.

Пodзadaчa 4 *(32 балла)*

3 < < 100 000, 1 < /г < 109, 1 < < 109

*Получение информации о результатах окончательной проверки*

По запросу сообщается результат окончательной проверки на каждом тесте.

# Задача 7. Интересные числа

Имя входного файла: number s . in

Имя выходного файла: number s . out Ограничение по времени: 1 секунда Ограничение по памяти: 256 мегабайт

Софья считает число *интересным,* если его цифры идут в неубывающем порядке.

Например, числа 123, 1111 или 888999 — интересные.

Софья заинтересовалась, сколько существует интересных положительных чисел, лежащих в диапазоне от *L* до Л включительно. Это число может оказаться довольно большим для больших *L* и Л, поэтому Софья хочет найти остаток от деления этого числа на 109 + 7.

Требуется написать программу, которая по заданным *L* и Л определяет количество интересных чисел, лежащих в диапазоне от *L* до Л включительно, и выводит остаток от деления этого числа на 10’ + 7.

Формат вхоdлосо *файла*

Входной файл содержит две строки. Первая строка содержит число *L,* вторая строка содержит число Л (1 < *L < R <* 10100)

Формат выхоdлоао *файла*

##### Выходной файл должен одно целое число — остаток от деления количества интересных чисел, лежащих в диапазоне от до Л включительно, на 10’ + 7.

*Примеры входных и выходных файлов*

|  |  |
| --- | --- |
| **numbers.in** | **numbers.out** |
| 1  100 | 54 |

*Описание подзадач и системы оценивания*

Пodзadaчa ’I *(21 балл) L —* 1, Л < 1000

Баллы за подзадачу начисляются только в случае, если все тесты подзадачи пройдены.

Пodзadaчa 2 *(do* 22 dаллов)

1 < *L* < it < 1o18

В этой подзадаче 11 тестов, каждый тест оценивается в 2 балла. Баллы за каждый тест начисляются независимо.

###### *Подзадача 3 (до 24 баллов)*

*L ——* 1, Л = 10k для некоторого целого *k,* 2 < *k <* 100.

В этой подзадаче 8 тестов, каждый тест оценивается в 3 балла. Баллы за каждый тест начисляются независимо.

###### *Подзадача 4 (до 33 баллов)*

1 < *L < R <* 10100

В этой подзадаче 11 тестов, каждый тест оценивается в 3 балла. Баллы за каждый тест

НІІЧИGЛЯЮТСЯ HeЗ іІВИGИМО.

*Получение информации о результатах окончательной проверки*

По запросу сообщается результат окончательной проверки на каждом тесте.

# Задача 8. Гармоничная последовательность

Имя входного файла: s еqueпсе . in

Имя выходного файла: se quen ее . out Ограничение по времени: 1 секунда Ограничение по памяти: 256 мегабайт

Цикл лекций в университете Флатландии посвящен изучению последовательностей.

Профессор называет последовательность целых чисел , , . .., *an* гармоничной, если каждое число, кроме *ai* и *and* Равно сумме соседних: п —— m + пЗ, пЗ = *аz +* п4. ., nп-1 - *an-2 + an* Например, последовательность [1, 2, 1, —1] является гармоничной, поскольку 2 = 1 + 1, и

1 = 2 + (—1).

Рассмотрим последовательности равной длины: *А —— [ai ,* 2, ., *an]* и *В [b , b z z bn]* • Расстоянием между этими последовательностями будем называть величину *d(A, В) ——*

— *be +* !п — *bz + + \a — b .* Например, *d([1,* 2 ,1, —1], [1, 2, 0, 0]) 1 — 1 + 2 — 2 +

+ 1 — 0 + —1 — 0 = 0 + 0 + 1 + 1 = 2.

В конце лекции профессор написал на доске последовательность из u целых чисел *В [b(z b z z bn]* и попросил студентов в качестве домашнего задания найти гармоничную последовательность *А [ , 2, ., а ],* такую, что *d(A, В)* минимально. Чтобы облегчить себе проверку, профессор просит написать в качестве ответа только искомое минимальное расстояние *d(A, В).*

Требуется написать программу, которая по заданной последовательности *В* определяет, на каком минимальном расстоянии от последовательности *В* найдется гармоничная последовательность *А.*

Формат вхоdлосо *файла*

Первая строка входного файла содержит целое число п — количество элементов в последовательности (3 u 300 000).

Вторая строка содержит п целых чисел *b , b2 , .. ., be* (—109 < *b, <* 109).

Формат выхоdлоао *файла*

Выходной файл должна содержать одно целое число: минимальное возможное расстояние от последовательности во входном файле до гармоничной последовательности.

### *Примеры входных и выходных файлов*

|  |  |
| --- | --- |
| **sequence.in** | **sequence.out** |
| 4  1 2 0 0 | 2 |

/7ояслелье х л£\*•меру

В приведенном примере оптимальной является, например, гармоничная последовательность [1, 2, 1, —1].

### *Описание подзадач и системы оценивания*

В этой задаче пять подзадач. Баллы за подзадачу начисляются только в случае, если все тесты для данной подзадачи пройдены.

**Внимание!** Тест из примера не подходит под ограничения для подзадачи 1, но решение принимается на проверку только в том случае, если оно выводит правильный ответ на тесте из примера. Решение должно выводить правильный ответ на тест, даже если оно рассчитано на решение только подзадачи 1.

*Подзадача 1 (14 баллов)*

п = 3, —10 *b, \_<* 10

Подзадача *2 (14 баллов)*

3 < < 500, —100 < *b,<* 100

*Подзадача 3 (16 баллов)*

3 п 100 000, —100 *be<* 100

Пodзadaчa 4 *(16 баллов)*

3 < л < 1000, —109 < *b, <* 109

###### *Подзадача 5 (40 баллов)*

3 п 300 000, —109 *b, \_<* 109

### *Получение информации о результатах окончательной проверки*

По запросу сообщается баллы за каждую подзадачу.