Задача W. Сумма

Имя входного файла: sum . in

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

Заданы два целых числа: *а п Ь.*

Требуется написать программу, которая вычисляет их сумму.

Формат вхоdлоао *файла*

Входной файл содержит разделенные пробелом целые числа п и *b* (1 п *b \_<* 109).

Формат *выходного файла*

Выходной файл должен содержать одно число — сумму чисел *а* и *Ь.*

### *Пример входных и выходных файлов*

**sum.in**

2 3

**sum.out**

5

*Система* оцелхь ь *описание подзадач*

Подзадача *1 (50 баллов)*

1 < *а < b <* 1000.

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

Пodзadaчa 2 *(50* баллов) 1 < п < *b <* 109.

В этой подзадаче 5 тестов, каждый тест оценивается в 10 баллов. Баллы за каждый

тест начисляются независимо.

/7олучелье ьлформацпь о *результатах окончательной проверки*

По запросу сообщается результат окончательной проверки на каждом тесте.

# Задача Х. Сумма квадратов

Имя входного файла: sums qr . in

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

#### Заданы два целых числа: *а п Ь.*

Требуется написать программу, которая вычисляет сумму их квадратов.

Формат вхоdлоао *файла*

Входной файл содержит разделенные пробелом целые числа п и *b* (1 п *b \_<* 109).

Формат *выходного файла*

Выходной файл должен содержать одно число — сумму квадратов чисел *а н Ь.*

*Пример входных и выходных файлов*

|  |  |
| --- | --- |
| **sumsqr.in** | **sumsqr.out** |
| 2 3 |  |

*Система* оцелхь ь *описание подзадач*

В этой задаче две подзадачи. Баллы за подзадачу начисляются только в случае, если все тесты этой подзадачи успешно пройдены.

Пodзadaчa I *(50* dаллов)

1 п *b \_<* 1000.

Пodзadaчa 2 *(50* баллов)

1 < п < *b <* 109.

/7олучелье ьлформацпь о *результатах окончательной проверки*

По запросу сообщается результат окончательной проверки на каждом тесте.

# Задача У. Робинзон и крокодилы

ИМявходногофаЗла: alligator.in

ИмявыходногофаЙла: alligator.out Ограничение по времени. 2 секунды Ограничение по памяти. 256 мегабаит

Робинзон живет на острове, которыи представляет собои прямоугольник размером

#### u \* ш клеток.

На остров Робинзона выползли погреться на солнышке и задремали несколько крокодилов. Робинзон хочет прогнать неприятных соседеи, не поднимая шума. Для этого он кидает в дремлющих крокодилов орехи.

В каждои клетке острова находится не более одного крокодила. Напуганныи орехом крокодил быстро бежит строго по прямои, пока не окажется в воде. Для каждого крокодила известно направление, в котором он побежит, если его напугать. Направления, в которых будут убегать крокодилы, параллельны сторонам острова.

Если на пуги напуганного крокодила окажется другои крокодил, то, столкнувшись, они разозлятся, и нападут на Робинзона. Поэтому надо тщательно выбирать очередного крокодила, чтобы на его пути были только пустые клетки.

Робинзон не кидает очереднои opex, пока предыдущии крокодил не окажется в воде. Требуется написать программу, определяющую максимальное количество

К]ЭОКОДИЛОВ, КОТО]ЭЫХ МОЖНО П]ЭОГНІІТЬ, HE ]ЭПЗОЗЛИВ ИХ.

Формат вхоdлосо *файла*

В первой строке входного файла записаны числа п и ш — размеры острова с севера на юг и с запада на восток. Последующие п строк по *т* символов в каждой описывают текущее расположение крокодилов на острове. Если клетка свободна, то она обозначается точкой

« . », а если там находится крокодил, то в ней указано направление, в котором побежит этот крокодил. Направления обозначаются буквами: «N» — север, «S» — юг, «Е» — восток, «W»

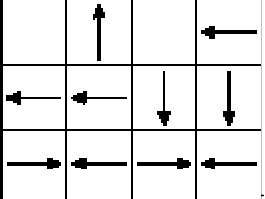
Формат выхоdлоао *файла*

Выходной файл должен содержать одно число — максимальное количество крокодилов, которых можно прогнать, не разозлив.

*Примеры входных и выходных файлов*

|  |  |
| --- | --- |
| **alligator.in** | **alligator.out** |
| 1 5  WN.SE | 4 |
| E.W | 0 |
| 3 4  .N.W WWSS EWEW | 4 |

Лояслелье х лрьмеру

Рисунок показывает исходное расположение крокодилов в третьем примере.

N

W  Е

S

### *Система оценки и описание подзадач*

Данная задача содержит три подзадачи. Баллы за подзадачу начисляются только в том случае, если все тесты этой подзадачи успешно пройдены.

Пodзadaчa I *(30 баллов)*

#### 1 < u, m < 30

##### *Подзадача 2 (30 баллов)*

1 п, *m* \_< 500

Пodзadaчa 3 *(40 баллов)*

#### 1 < u, m < 2000

*Получение информации о результатах окончатепьной проверки*

По запросу сообщается результат окончательной проверки на каждом тесте.

# Задача Z. Конфеты

Имя входного файла: sweet s . in

Имя выходного файла: sweet s . out Ограничение по времени: 2 секунды Ограничение по памяти: 256 мегабайт

Кондитерская фабрика города П, в котором живет Петя делает очень вкусные конфеты. Как-то раз, Петя собрался в гости к своему другу Bace, который живет в городе М. От города П до города М Петя решил доехать на поезде и взять с собой в подарок как можно больше коробок вкусных конфет.

Каждая коробка конфет имеет размер *а Х b* с сантиметров, где *а —* длина, *b —* ширина

и г — высота коробки. Для перевозки конфет Петя хочет использовать один большой ящик в форме прямоугольного параллелепипеда. В ящик должны быть уложены все коробки конфет. Для того чтобы не повредить их, все коробки в ящике должны сохранять исходную ориентацию и располагаться в одном направлении. Петя может использовать ящик любого размера, но по правилам железнодорожных перевозок размер ящика по сумме трех измерений не может превышать *N* сантиметров.

Требуется написать программу, которая по заданным числам *N, а, b н с* определяет такой размер ящика, который должен использовать Петя, чтобы в него поместилось максимальное количество коробок конфет.

Формат вхоdлосо *файла*

Первая строка входного файла содержит разделенные пробелами четыре целых числа:

N, п, *b,* с (1 < *N, а, b, с <* 109).

Формат выхоdлоао *файла*

Выходной файл должен содержать три числа — длину, ширину и высоты ящика, который должен выбрать Петя и в который поместится максимальное количество коробок конфет. Если подходящих ответов несколько, необходимо вывести любой.

/7рьмеры вхоdлых ь выхоdлых *файлов*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **sweets.in** | | | | **sweets.out** | | |
| 10 | 1 | 2 | 3 | 3 | 4 | 3 |
| 14 | 8 | 3 | 2 | 9 | 3 | 2 |

/7ояслелья If *ПR<мера м*

В первом примере выгоднее всего взять ящик размером 3 \* 4 х 3 сантиметров, в который поместится три коробки конфет в длину, две коробки конфет в ширину и одна коробка конфет в высоту.

Во втором примере для того, чтобы разместить хотя бы две коробки, нужен ящик размером хотя бы 8 3 4, у которого сумма измерений равна 15. Поэтому в подходящий ящик поместится максимум одна коробка конфет. В том числе для этого подходит ящик размером 9 3 2, хотя он и не является минимальным.

*Система* оцелхь ь *описание подзадач*

##### *Подзадача 1 (20 баллов)*

I *< N <* 300

В этой подзадаче 10 тестов, каждый тест оценивается в 2 балла. Баллы за каждый тест

НІІЧИGЛЯЮТСЯ HeЗ іІВИGИМО.



##### *Подзадача 2 (20 баллов)*

I *< N <* 5000

В этой подзадаче 10 тестов, каждый тест оценивается в 2 балла. Баллы за каждый тест

НІІЧИGЛЯЮТСЯ HeЗdBИGИMO.

##### *Подзадача 3 (30 баллов)*

1 N \_< 100 000

В этой подзадаче 15 тестов, каждый тест оценивается в 2 балла. Баллы за каждый тест начисляются независимо.

Подзадача *4 (30 баллов)*

I *< N <* 109

В этой подзадаче 15 тестов, каждый тест оценивается в 2 балла. Баллы за каждый тест

НІІЧИСЛЯЮТСЯ НeЗIIBИCИMO.

*Получение информации о результатах окончательной проверки*

По запросу сообщаются баллы за каждую подзадачу.