A. Interactor

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 256Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Лена руководит разработкой тестирующей системы, в которой реализованы интерактивные задачи.

До заверщения очередной стадии проекта осталось написать модуль, определяющий *итоговый вердикт*системы для интерактивной задачи. *Итоговый вердикт*определяется из кода завершения задачи, вердикта интерактора и вердикта чекера по следующим правилам:

* Вердикт чекера и вердикт интерактора — это целые числа от 0 до 7 включительно.
* Код завершения задачи — это целое число от -128 до 127 включительно.
* Если интерактор выдал вердикт 0, итоговый вердикт равен 3 в случае, если программа завершилась с ненулевым кодом, и вердикту чекера в противном случае.
* Если интерактор выдал вердикт 1, итоговый вердикт равен вердикту чекера.
* Если интерактор выдал вердикт 4, итоговый вердикт равен 3 в случае, если программа завершилась с ненулевым кодом, и 4 в противном случае.
* Если интерактор выдал вердикт 6, итоговый вердикт равен 0.
* Если интерактор выдал вердикт 7, итоговый вердикт равен 1.
* В остальных случаях итоговый вердикт равен вердикту интерактора.

Ваша задача — реализовать этот модуль.

Формат ввода

Входной файл состоит из трёх строк. В первой задано целое число r (−128≤r≤127) — код завершения задачи, во второй — целое число i (0≤i≤7) — вердикт интерактора, в третьей — целое число c (0≤c≤7) — вердикт чекера.

Формат вывода

Выведите одно целое число от 0 до 7 включительно — итоговый вердикт системы.

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 0  0  0 | 0 |

Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| -1  0  1 | 3 |

Пример 3

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 42  1  6 | 6 |

Пример 4

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 44  7  4 | 1 |

Пример 5

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 1  4  0 | 3 |

Пример 6

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| -3  2  4 | 2 |

B. Кольцевая линия метро

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Витя работает недалеко от одной из станций кольцевой линии Московского метро, а живет рядом с другой станцией той же линии. Требуется выяснить, мимо какого наименьшего количества промежуточных станций необходимо проехать Вите по кольцу, чтобы добраться с работы домой.

Формат ввода

Станции пронумерованы подряд натуральными числами 1, 2, 3, …, N (1-я станция – соседняя с N-й), N не превосходит 100.

Вводятся три числа: сначала N – общее количество станций кольцевой линии, а затем i и j – номера станции, на которой Витя садится, и станции, на которой он должен выйти. Числа i и j не совпадают. Все числа разделены пробелом.

Формат вывода

Требуется выдать минимальное количество промежуточных станций (не считая станции посадки и высадки), которые необходимо проехать Вите.

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 100 5 6 | 0 |

Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 10 1 9 | 1 |

Примечания

Пояснения к примерам:

1) На кольцевой линии 100 станций; проехать с 5-й на 6-ю станцию Витя может напрямую, без промежуточных станций

2) На кольцевой линии 10 станций; проехать с 1-й на 9-ю станцию Витя может через одну промежуточную, ее номер 10

C. Даты

|  |  |
| --- | --- |
| Ограничение времени | 2 секунды |
| Ограничение памяти | 512Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Как известно, два наиболее распространённых формата записи даты — это европейский (сначала день, потом месяц, потом год) и американски (сначала месяц, потом день, потом год). Системный администратор поменял дату на одном из бэкапов и сейчас хочет вернуть дату обратно. Но он не проверил, в каком формате дата используется в системе. Может ли он обойтись без этой информации?

Иначе говоря, вам даётся запись некоторой корректной даты. Требуется выяснить, однозначно ли по этой записи определяется дата даже без дополнительной информации о формате.

Формат ввода

Первая строка входных данных содержит три целых числа — x, y и z (1≤x≤31, 1≤y≤31, 1970≤z≤2069. Гарантируется, что хотя бы в одном формате запись xyz задаёт корректную дату.

Формат вывода

Выведите 1, если дата определяется однозначно, и 0 в противном случае.

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 1 2 2003 | 0 |

Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 2 29 2008 | 1 |

Примечания

В первом примере при одной системе записи дата равна 1 февраля, при другой - 2 января 2003 года, то есть однозначно назвать дату не получается.

Во втором примере корректный вариант даты может быть только в американском формате, где она задаёт 29 февраля 2008 года.

D. Строительство школы

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

В деревне Интернетовка все дома расположены вдоль одной улицы по одну сторону от нее. По другую сторону от этой улицы пока ничего нет, но скоро все будет – школы, магазины, кинотеатры и т.д.

Для начала в этой деревне решили построить школу. Место для строительства школы решили выбрать так, чтобы суммарное расстояние, которое проезжают ученики от своих домов до школы, было минимально.

План деревни можно представить в виде прямой, в некоторых целочисленных точках которой находятся дома учеников. Школу также разрешается строить только в целочисленной точке этой прямой (в том числе разрешается строить школу в точке, где расположен один из домов – ведь школа будет расположена с другой стороны улицы).

Напишите программу, которая по известным координатам домов учеников поможет определить координаты места строительства школы.

Формат ввода

Сначала вводится число N — количество учеников (0 < N < 100001). Далее идут в строго возрастающем порядке координаты домов учеников — целые числа, не превосходящие *2 × 109* по модулю.

Формат вывода

Выведите одно целое число — координату точки, в которой лучше всего построить школу. Если ответов несколько, выведите любой из них.

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 4  1 2 3 4 | 3 |

Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 3  -1 0 1 | 0 |

E. Точка и треугольник

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

На координатной плоскости расположены равнобедренный прямоугольный треугольник ABC с длиной катета d и точка X. Катеты треугольника лежат на осях координат, а вершины расположены в точках: A (0,0), B (d,0), C (0,d).

Напишите программу, которая определяет взаимное расположение точки X и треугольника. Если точка X расположена внутри или на сторонах треугольника, выведите 0. Если же точка находится вне треугольника, выведите номер ближайшей к ней вершины.

Формат ввода

Сначала вводится натуральное число d (не превосходящее 1000), а затем координаты точки X – два целых числа из диапазона от ­–1000 до 1000.

Формат вывода

Если точка лежит внутри, на стороне треугольника или совпадает с одной из вершин, то выведите число 0. Если точка лежит вне треугольника, то выведите номер вершины треугольника, к которой она расположена ближе всего (1 – к вершине A, 2 – к B, 3 – к C). Если точка расположена на одинаковом расстоянии от двух вершин, выведите ту вершину, номер которой меньше.

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 5  1 1 | 0 |

Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 3  -1 -1 | 1 |

Пример 3

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 4  4 4 | 2 |

Пример 4

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 4  2 2 | 0 |

Примечания

Комментарии к примерам тестов

1. Точка лежит внутри треугольника.

2. Точка лежит вне треугольника и ближе всего к ней вершина A

3. Точка лежит на равном расстоянии от вершин B и C,в этом случае нужно вывести ту вершину, у которой номер меньше, т.е. выведено должно быть число 2

4. Точка лежит на стороне треугольника.