

Задача 1. Стремление к совершенству

Ограничение по времени 2 с
Ограничение по памяти 64 Мб
Баллов за задачу 10
Пример названия программы p99_a1.cpp

Андрей решил вплотную заняться решением проблемы десятилетиям — проблемой *совершенности* многоугольников. Но так как в школе по геометрии у него была тройка, то он решил обратиться к Вам за помощью.

Все знают, что круг является *совершенной* фигурой. Правильный n -угольник очень близок к совершенству, но все же не достиг этого почетного звания. Для того, чтобы правильный n -угольник стал совершенным ему необходимо "набрать вес".

Формат входных данных

В первой строке входа содержится целое число n ($3 \leq n \leq 1000$) — количество вершин правильного многоугольника. Во второй строке написано вещественное число $0 < R < 1000$ — радиус описанной, вокруг данного многоугольника, окружности с точностью до 10^{-6} .

Формат выходных данных

Выведите отношение площади круга к площади n -угольника с точностью не менее 10^{-6} . Тем самым Вы поможете Андрею разрешить проблему десятилетия!

Пример

Вход	Выход
4 1.41421356237	1.57079632
3 1	2.4183992

Задача 2. Разные числа

Ограничение по времени 2 с
Ограничение по памяти 64 МБ
Баллов за задачу 10
Пример названия программы p99_a2.cpp

В этой задаче нет никакого описания.

Формат входных данных

В первой строке входа содержится целое число a ($1 \leq a \leq 10^6$). Во второй строке входа содержится целое число b ($1 \leq b \leq 10^6$).

Формат выходных данных

Необходимо вывести количество различных вещественных чисел среди множества чисел: $\{a + b; b + a; a - b; b - a; a \cdot b; b \cdot a; a/b; b/a; a^b; b^a\}$.

Например, если $a = 2$, $b = 4$, то получится следующее:

$\{a + b = 6; b + a = 6; a - b = -2; b - a = 2; a \cdot b = 8; b \cdot a = 8; a/b = 0.5; b/a = 2; a^b = 16; b^a = 16\}$.

Итого 6 различных чисел.

Пример

Вход	Выход
2 4	6
1 1	3

Задача 3. Пересечение

Ограничение по времени 2 с
Ограничение по памяти 64 Мб
Баллов за задачу 15
Пример названия программы p99_a3.cpp

Геометрия становится все более популярной наукой, поэтому Вам предстоит помочь решить Андрею еще одну геометрическую задачу.

На плоскости заданы два невырожденных прямоугольника со сторонами, параллельными осям координат. Требуется найти форму пересечения прямоугольников.

Формат входных данных

В первой строке входа содержится 4 целых числа $(x_1, y_1), (x_2, y_2)$ — координаты двух противоположных вершин первого прямоугольника.

Во второй строке входа содержится 4 целых числа $(x_3, y_3), (x_4, y_4)$ — координаты двух противоположных вершин второго прямоугольника.

Все числа не превосходят 10^4 по модулю.

Формат выходных данных

Ответ на задачу — одну заглавную букву латинского алфавита.

R — если пересечение является прямоугольником.

S — если пересечение является отрезком.

P — если пересечение является точкой.

N — если пересечение пусто.

Пример

<i>Вход</i>	<i>Выход</i>
0 0 5 5 10 10 20 20	N
0 0 10 10 3 3 5 5	R

Задача 4. Будильник

Ограничение по времени 2 с
 Ограничение по памяти 64 Мб
 Баллов за задачу 20
 Пример названия программы r99_a4.cpp

Для многих людей встать рано утром является очень большой проблемой. И Андрей не является исключением. Для того, чтобы он никогда не просыпал очень важные занятия в Университете родители подарили ему будильник. Да не простой, а японский!

Будильник является очень продвинутым и может быть запрограммирован на включение в разное время в зависимости от дня недели. Но, к сожалению, японские инженеры не успели дописать очень важную функциональность. А именно, будильник должен по заданному набору правил и текущему времени показывать сколько осталось часов и минут до следующего срабатывания.

Вам поручили ответственное задание — написать эту функциональность.

Формат входных данных

Первая строка входа содержит текущее время в формате

HH:MM DD

- DD — день недели. Одно из следующих значений: **Mon, Tue, Wed, Thu, Fri, Sat, Sun**, обозначающие понедельник, вторник, среду, четверг, пятницу, субботу и воскресенье соответственно.
- HH — час (число от 0 до 23).
- MM — минута (число от 0 до 59).

Вторая строка содержит целое число $1 \leq n \leq 10$ — количество правил для будильника.

Следующие n строк содержат по одному правилу в строке, описывающие время, когда должен работать будильник. Формат правила соответствует формату текущего времени, за исключением того, что может быть указано несколько дней недели, разделенных пробелами. Гарантируется, что текущее время не совпадает ни с одним из правил будильника.

Формат выходных данных

Вывести два целых числа, разделенные двоеточием — количество часов и минут до следующего срабатывания будильника. Если количество минут меньше 10, то выведите ведущий ноль.

Пример

<i>Вход</i>	<i>Выход</i>
09:20 Sun 3 03:34 Sun 10:20 Sun Mon Tue Wed Thu Fri Sat 00:00 Tue	1:00
12:34 Sat 1 12:33 Fri Sat	143:59

Задача 5. Выбор

Ограничение по времени 2 с
 Ограничение по памяти 64 Мб
 Баллов за задачу 20
 Пример названия программы p99_a5.cpp

Андрей целый год экономил на обеде в Университете и накопил денег на новый компьютер. Осталось дело за малым — выбрать наилучший вариант. Современные компьютеры очень сложны по своему устройству, поэтому выбор будет очень трудным.

У Андрея есть список из n ($2 \leq n \leq 10^5$) моделей компьютеров каждый из которых характеризуется k ($1 \leq k \leq 5$) различными параметрами. Каждый параметр представляет собой целое число от 1 до 50. Чем параметр больше, тем компьютер лучше. Андрей хочет существенно сократить список компьютеров для рассмотрения путем отбрасывания заранее проигрышных вариантов.

Компьютер x является заранее *проигрышным* по отношению к компьютеру y , если все кроме одного параметры x не лучше, а как минимум один хуже чем параметр y . Например, пусть x характеризуется пятеркой (1, 4, 5, 2, 9), а y описывается (1, 5, 5, 2, 9). В этом случае x является заранее проигрышным по отношению к y , так как второй параметр x меньше чем у y , а все остальные не превосходят параметров y . Если x задано пятеркой (1, 1, 1, 1, 2), а y — (2, 1, 1, 1, 1), то ни одно из них не является проигрышной по отношению к другой. Необходимо вывести количество компьютеров оставшихся после отбрасывания заведомо проигрышных моделей.

Формат входных данных

Первая строка входа содержит два целых числа n и k , разделенные пробелом. Далее идут n строк, каждая из которых содержит по k целых чисел от 1 до 50 — параметры очередного компьютера.

Формат выходных данных

Одно целое число — ответ на задачу.

Пример

Вход	Выход
4 1 1 2 3 4	1
5 3 5 2 4 6 4 3 5 5 5 4 3 2 6 3 3	2

Во втором примере ответ 2, потому что (4, 3, 2) и (5, 2, 4) являются проигрышными по отношению к (5, 5, 5), а (6, 3, 3) является проигрышной по отношению к (6, 4, 3). Поэтому Вася будет рассматривать только два варианта: (5, 5, 5) и (6, 4, 3).

Задача 6. Зеленый коридор

Ограничение по времени	2 с
Ограничение по памяти	64 Мб
Баллов за задачу	25
Пример названия программы	r99_a6.cpp

Владельцы автомобилей конечно же догадываются о чем пойдет речь в этой задаче. *Зеленым коридором* называется участок дороги, которую автомобилист может проехать, не изменяя скорость движения, при этом, не остановившись ни на одном светофоре. Вам предстоит помочь Андрею определить максимально возможную скорость или сообщить, что зеленого коридора не существует.

Дорога от дома Андрея до Университета представляет собой прямой участок длиной l ($1 \leq l \leq 1000$). На нем находятся n ($0 \leq n \leq 100$) светофоров. На каждом светофоре одновременно может гореть либо зеленый, либо красный свет. Переключение с зеленого на красный цвет и наоборот происходит мгновенно. Автомобиль не может проехать светофор, если на нем в момент проезда под светофором горит красный цвет. Однако, автомобилист может проехать светофор точно в момент переключения с красного на зеленый, и наоборот (т.е. в момент переключения с зеленого на красный).

Для каждого светофора известны следующие величины:

x_i — координата светофора ($1 \leq x_i < l$);

g_i — интервал в течение которого горит зеленый цвет ($2 \leq g_i \leq 1000$);

r_i — интервал в течение которого горит красный цвет ($2 \leq r_i \leq 1000$);

c_i — цвет светофора в момент времени $t = 0$. c_i равно "r" (без кавычек), если в момент времени $t = 0$ на светофоре горит красный цвет и "g" (без кавычек) — если зеленый;

t_i — промежуток в течение которого уже горит текущий цвет на момент времени $t = 0$. Гарантируется, что величина t_i не меньше 1 и строго меньше интервала в течение которого горит соответствующий цвет.

Автомобилист в момент времени $t = 0$ находится в точке $x = 0$.

Все расстояния и координаты заданы в метрах; временные интервалы в секундах. Все величины целые.

Формат входных данных

В первой строке находится длина дороги l . Во второй строке находится количество светофоров n . В последующих n строках находится описание очередного светофора. Сначала идет буква обозначающая цвет светофора в момент времени $t = 0$, затем через пробел указаны величины x_i , g_i , r_i и t_i .

Гарантируется, что никакие два светофора не находятся в одной точке.

Формат выходных данных

Вывести максимально возможную скорость автомобиля с точностью не менее 10^{-3} , чтобы автомобиль, стартовав с этой скоростью в момент времени $t = 0$, смог проехать дорогу без остановки. Причем известно, что на всей дороге действует ограничение по скорости и автомобиль не может двигаться со скоростью более 100 метров в секунду. Но и для того, чтобы не тормозить все движение, скорость автомобиля должна быть не менее 1 метра в секунду.

Если зеленого коридора не существует, то вывести строку "Impossible" (без кавычек).

Пример

Вход	Выход
100 1 r 20 10 5 2	6.667
100 2 r 10 5 5 4 g 40 5 4 4	8.000