

Задача 1. Аренда

Имя входного файла: *input.txt*
Имя выходного файла: *output.txt*
Ограничение по времени: *1 секунда*
Ограничение по памяти: *64 мегабайта*
Максимальная оценка: *100 баллов*

До Академгородка дотянулась первая компания по предоставлению услуг телепатической связи. Для построения станции телепатической связи нужно несколько «телепатических точек». Чем больше точек, тем лучше. Разведка собрала информацию обо всех телепатических точках по региону. Планируется арендовать всю территорию, которая содержит все эти точки. К сожалению, землемеры могут обмерить участок земли только если он выпуклый. Вам требуется минимизировать площадь арендуемого участка.

Входные данные

В первой строке входного файла записано целое число N – количество телепатических точек на карте ($1 \leq N \leq 300000$). Каждая из последующих N строк содержит два целых числа X_i, Y_i – координаты точки. Координаты по абсолютной величине не превосходят 10000. В силу особенностей телепатии в одной точке карты может находиться несколько телепатических точек.

Выходные данные

Выведите одно число с максимальной точностью – минимальную площадь выпуклого участка, содержащего все заданные телепатические точки.

Пример

<i>input.txt</i>	<i>output.txt</i>
5 1 1 0 0 2 2 0 2 2 0	4.0

Решение, правильно работающее для ограничения $N \leq 10000$, набирает 66 баллов.

Задача 2. Обнаружение коллизий

Имя входного файла:	<code>input.txt</code>
Имя выходного файла:	<code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта
Максимальная оценка	100 баллов

Вас просят протестировать новый модуль по определению столкновений твёрдых тел. Для простоты тестирование будет проводится на примере двух плоских тел. Граница каждого тела задана многоугольником (возможно невыпуклым). От вашей программы требуется выдавать точную характеристику взаимного расположения тел. Возможно три варианта:

- 1 Тела не пересекаются (нет общих точек тел)
- 2 Границы тел пересекаются (есть хотя бы одна общая точка границ тел)
- 3 Границы не пересекаются, но при этом одно тело находится внутри другого.

Входные данные

Входной файл содержит описания двух многоугольников. Каждый многоугольник описывается количеством вершин в нём и списком вершин. Каждая вершина описывается на отдельной строке двумя координатами. Количество вершин не превосходит 1000. Координаты целые, не превосходят 10000 по модулю.

Оба многоугольника нормальные. Многоугольник нормальный, если несмежные отрезки границы не пересекаются, а смежные имеют ровно одну общую точку. Также оба многоугольника невырожденные, то есть имеют ненулевую площадь.

Выходные данные

На первой строке выведите общий вердикт (один из следующих):

- 1 “cross” – многоугольники пересекаются
- 2 “separate” – тела не пересекаются
- 3 “first is inside” – первый многоугольник лежит внутри второго тела
- 4 “second is inside” – второй многоугольник лежит внутри первого тела

В случае если многоугольники пересекаются (cross), выведите на второй строке координаты любой общей точки многоугольников. Рекомендуется выдавать координаты с точностью порядка 15 знаков после запятой.

Пример

<i>input.txt</i>	<i>output.txt</i>
4 0 0 10 0 10 10 0 10 3 1 1 9 8 8 9	second is inside
4 0 0 1 0 1 1 0 1 4 1 0 2 0 2 1 1 1	cross 1.0000000000000000 0.0000000000000000
3 -100 0 -100 1 -99 0 3 100 0 100 1 99 0	separate
3 0 0 5 -5 5 5 3 2 0 2 10 10 1	cross 5.0000000000000000 0.3750000000000000

Задача 3. Баян

Имя входного файла: *input.txt*
Имя выходного файла: *output.txt*
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта
Максимальная оценка: 100 баллов

Дано две окружности, найти все точки пересечения.

Входные данные

Во входном файле две строки, каждая из них описывает одну окружность. Окружность задаётся числами: x , y – координаты центра, r – радиус ($-10000 \leq x, y \leq 10000$, $0 < r \leq 10000$). Все числа целые.

Выходные данные

В первой строке выведите количество общих точек окружностей. Если их бесконечно много, то выведите -1.

В последующих строках требуется вывести все общие точки окружностей в случае, если их конечное число. Каждая точка описывается координатами x , y . Точки должны быть различными, могут идти в любом порядке. Координаты выводите с точностью не менее 5 знаков после запятой.

Пример

<i>input.txt</i>	<i>output.txt</i>
0 0 1 2 0 1	1 1.0000000000000000 0.0000000000000000
0 0 1 1 0 1	2 0.50000 -0.86603 0.50000 0.86603

Задача 4. Сравнение асимптотик

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта
Максимальная оценка: 100 баллов

Вася увлекается решением олимпиадных задач по информатике. Время работы различных алгоритмов в зависимости от N он оценивает асимптотически. Вася выписал несколько асимптотических оценок вида $\Theta(a * N^b * c^N * \log^d(N))$. Он просит вас выстроить эти асимптотики в порядке возрастания времени работы.

Входные данные

Во входном файле записано целое число N – количество асимптотик ($2 \leq N \leq 100$). В следующих N строках описываются асимптотики. Каждая строка содержит четыре вещественных числа a, b, c, d – параметры асимптотики ($0 \leq b, d \leq 100, 1 \leq a, c \leq 100$). Все числа заданы с точностью не более 2 знаков после запятой.

Выходные данные

В выходной файл выведите те же асимптотики, но выстроенные по неубыванию. Каждая из N строк должна содержать параметры асимптотики.

Пример

<i>input.txt</i>	<i>output.txt</i>
3	1 0 1 1
1 2.5 1 0	2 1 1 0
1 0 1 1	1 2.5 1 0
2 1 1 0	

В примере даны асимптотики в порядке возрастания: $\Theta(\log(N))$, $\Theta(2N)$, $\Theta(N^{2.5})$

Задача 5. Простота отрезка

Имя входного файла: *input.txt*
Имя выходного файла: *output.txt*
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта
Максимальная оценка: 100 баллов

Назовём степенью простоты отрезка $[L .. R]$ количество простых чисел в диапазоне от L до R включительно. Например, степень простоты $[8..10]$ равна нулю, а степень простоты отрезка $[3..5]$ равна двум. Найдите степени простоты заданных отрезков.

Входные данные

В первой строке записано целое число N – количество отрезков ($1 \leq N \leq 10000$). Каждая из последующих N строк содержит два целых числа L и R – границы отрезка ($1 \leq L \leq R \leq 10000000$).

Выходные данные

Для каждого отрезка выведите в файл строку со степенью его простоты.

Пример

<i>input.txt</i>	<i>output.txt</i>
2	0
8 10	2
3 5	

Задача 6. Степень

Имя входного файла: *input.txt*
Имя выходного файла: *output.txt*
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта
Максимальная оценка: 100 баллов

Вася необходимо посчитать сумму геометрической прогрессии. Конечно, когда-то на уроках математики рассказывали формулу подсчёта суммы, однако Вася её уже забыл.

Напомним, что геометрической прогрессией называется последовательность вида:

$$A, Aq, Aq^2, Aq^3, \dots, Aq^i, \dots, Aq^{L-2}, Aq^{L-1}$$

Где A – первый элемент, q – степень, а L – длина последовательности. Например, последовательность длины 5 с первым элементом 2 и степенью 3 равна: 2, 6, 18, 54, 162. Сумма этой последовательности равна 242.

Васе даны параметры последовательности, и ему нужно посчитать только остаток от деления суммы последовательности на простое число P . К примеру, для примера выше и $P=5$ остаток равен 2 ($242 \bmod 5$).

Помогите Васе.

Входные данные

В единственной строке дано четыре целых числа:

A – первый элемент ($1 \leq A < P$)

q – степень ($1 \leq q < P$)

L – длина ($1 \leq L \leq 1000000000$)

P – простое число ($2 \leq P < 1000000000$)

Выходные данные

Выведите одно целое число в диапазоне от 0 до $P-1$ – остаток от деления на P суммы геометрической прогрессии длины L с первым элементом A и степенью q .

Пример

<i>input.txt</i>	<i>output.txt</i>
2 3 5 5	2

Задача 7. Диофантово уравнение

Имя входного файла: *input.txt*
Имя выходного файла: *output.txt*
Ограничение по времени: *1 секунд*
Ограничение по памяти: *64 мегабайта*
Максимальная оценка: *100 баллов*

Дано диофантово уравнение:

$$A * X + B * Y = C$$

где A , B , C – заданные натуральные числа. Нужно найти целые числа X и Y , которые удовлетворяют этому уравнению. Требуется также, чтобы числа X и Y входили в 32-битовое знаковое целое число (int на C/C++, longint на Pascal).

Входные данные

Единственная строка входного файла содержит три целых числа – коэффициенты уравнения A , B , C ($1 \leq A, B, C \leq 1000000000$).

Выходные данные

В выходном файле выведите любую пару X и Y , которая удовлетворяет уравнению. Если решения не существует, выведите единственную строку “IMPOSSIBLE” (без кавычек).

Пример

<i>input.txt</i>	<i>output.txt</i>
2 1 1	500000000 -999999999
2 2 1	IMPOSSIBLE

Задача 8. Функция Эйлера

Имя входного файла: *input.txt*
Имя выходного файла: *output.txt*
Ограничение по времени: *1 секунда*
Ограничение по памяти: *64 мегабайта*
Максимальная оценка: *100 баллов*

Числа A и B называются взаимно простыми, если их наибольший общий делитель равен единице. Количество чисел от 0 до $N-1$, взаимно простых с N равно $\varphi(N)$. Например, $\varphi(1) = 1$, $\varphi(7) = 6$, $\varphi(10) = 4$, $\varphi(16) = 8$. Функция φ называется функцией Эйлера.

Требуется найти значения функции Эйлера для заданных натуральных чисел.

Входные данные

В первой строке записано целое число N – количество чисел в файле ($1 \leq N \leq 50$). Последующие N строк содержат по одному целому числу A_i ($1 \leq A_i \leq 1000000000000$).

Выходные данные

Для каждого числа A_i выведите в файл $\varphi(A_i)$ в отдельную строку.

Пример

<i>input.txt</i>	<i>output.txt</i>
4	1
1	6
7	4
10	8
16	

Задача 9. Алгоритм для робота

Имя входного файла: *input.txt*
Имя выходного файла: *output.txt*
Ограничение по времени: *1 секунда*
Ограничение по памяти: *64 мегабайта*
Максимальная оценка: *100 баллов*

Квадратный зал со стороной N разделен на клетки размером 1×1 . Этот зал моет робот. Для эффективной мойки он движется по диагоналям. Сначала он находится в левом верхнем углу зала, потом перемещается вправо на одну клетку, затем идет по диагонали вниз до конца, потом переходит в ближайшую первую клетку следующей диагонали (смотрите пример), и продолжает до тех пор, пока весь зал не будет помыт.

Входные данные

Во входном файле записано целое число N – размер стороны зала ($2 \leq N \leq 100$).

Выходные данные

В выходной файл нужно вывести по строкам квадратную матрицу размером $N \times N$, в которой пронумерованы клетки зала в порядке обхода их роботом.

Пример

<i>input.txt</i>	<i>output.txt</i>
5	1 2 6 7 15 3 5 8 14 16 4 9 13 17 22 10 12 18 21 23 11 19 20 24 25

Задача 10. Телефонный номер

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта
Максимальная оценка: 100 баллов

Если вы обратили внимание, то клавиатура многих телефонов выглядит следующим образом:

1	2 ABC	3 DEF
4 GHI	5 JKL	6 MN
7 PRS	8 TUV	9 WXY
	0 OOZ	

Использование изображенных на клавишах букв позволяет представить номер телефона в виде легко запоминающихся слов, что бывает часто более удобным, чем традиционная запись телефона в виде последовательности цифр. Многие фирмы пользуются этим и стараются подобрать себе номер телефона так, чтобы он содержал как можно больше букв из имени фирмы.

Требуется написать программу, которая преобразует исходный цифровой номер телефона в соответствующую последовательность букв и цифр, содержащую как можно больше символов из названия фирмы. При этом буквы из названия фирмы должны быть указаны в полученном номере в той же последовательности, в которой они встречаются в названии фирмы. Например, если фирма называется IBM, а исходный номер телефона — 246, то замена его на BIM не допустима, тогда как замена его на 2IM или B4M является правильной.

Входные данные

Первая строка входного файла содержит название фирмы. Она состоит только из заглавных букв латинского алфавита, количество которых не превышает 80 символов. Вторая строка содержит номер телефона в виде последовательности цифр. Цифр в номере телефона также не более 80.

Выходные данные

В единственной строке выходного файла должно содержаться количество букв из измененного номера.

Пример

<code>input.txt</code>	<code>output.txt</code>
IBM 246	2

Задача 11. Fallout

Имя входного файла: *input.txt*
Имя выходного файла: *output.txt*
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта
Максимальная оценка: 100 баллов

Ядерный удар был нанесён успешно. Ядерная война началась и вскоре закончилась. Уцелевшее население на многие года укрылось в подземных убежищах. Вы – житель убежища номер 13. В вашем убежище сломался водоочистительный фильтр. Те, кто знал принципы работы фильтра, уже давно умерли. Совет убежища решает отправить вас на поверхность в поисках нового фильтра.

Из древних записей известно, что вход в убежище расположен в глубине пещеры. Для того, чтобы выбраться на поверхность, нужно пройти через всю пещеру. Разумеется, в пещере обитают злобные крысы-мутанты. Поэтому вам понадобится оружие. Вы решили взять пистолет Colt-6520 10мм и боевой нож.

Для того, чтобы убить крысу-мутанта при помощи пистолета, нужны патроны. Вам предложили несколько разных ящиков 10-миллиметровых патронов на выбор. Вы можете взять с собой только один из этих ящиков. Для каждого ящика известен его вес в килограммах и приблизительное количество крыс-мутантов, которые можно застрелить с его помощью.

Также при стычке с крысой вы можете воспользоваться вашим ножом. Однако при схватке рукопашную вам неизбежно будет нанесён урон. Для того чтобы восстановить уровень здоровья, нужно использовать стимуляторы. Вам предложили на выбор несколько упаковок со стимуляторами. Вам разрешили взять с собой только одну из этих упаковок. Для каждой упаковки известен вес и приблизительное количество крыс, которое можно зарезать при помощи ножа, пользуясь этой упаковкой.

Вы конечно суперкачок, но вес, который вы можете носить без ущерба для боевых навыков, ограничен. Вы не можете взять упаковку стимуляторов и ящик патронов суммарным весом больше, чем вы можете свободно нести.

За дверью убежища неизвестность. Никто не знает, насколько пещера длинная. Никто не знает, сколько злобных крыс-мутантов в ней прячется. Однако вы хотите жить. Поэтому вам нужно выбрать ящик с патронами и упаковку со стимуляторами так, чтобы максимизировать количество крыс, которое вы сможете убить.

Входные данные

В первой строке записано три целых числа: N – количество разных ящиков с патронами, M – количество разных упаковок со стимуляторами и W – максимальный вес в килограммах, который вы можете свободно нести ($1 \leq N, M \leq 100000, 1 \leq W \leq 10^9$). Следующие N строк описывают ящики с патронами. Каждая строка содержит два натуральных числа – вес ящика и расчётное количество убитых крыс. Оба числа не превосходят 10^9 . Оставшиеся M строк описывают упаковки со стимуляторами. Каждая строка содержит два натуральных числа – вес упаковки и количество потенциально мёртвых крыс. Оба числа не превосходят 10^9 .

Выходные данные

Если невозможно унести и патроны, и стимуляторы, то выведите “DIE” в единственную строку выходного файла.

Иначе выведите “SURVIVE” в первую строку, суммарный вес и максимальное суммарное количество потенциальных крысиных трупиков во вторую, а также номера ящика с патронами и упаковки со стимуляторами в третью строку. Если решений несколько, выводите любое.

Пример

<i>input.txt</i>	<i>output.txt</i>
1 1 60 40 1000 30 250	DIE

1 1 80 40 1000 30 250	SURVIVE 70 1250 1 1
2 3 80 10 100 20 150 50 50 60 50 55 60	SURVIVE 75 210 2 3
2 3 65 10 100 20 150 50 50 60 50 55 60	SURVIVE 65 160 1 3

Задача 12. Скобки

Имя входного файла: *input.txt*
Имя выходного файла: *output.txt*
Ограничение по времени: 5 секунд
Ограничение по памяти: 64 мегабайта
Максимальная оценка: 100 баллов

Дана строка из n маленьких букв латинского алфавита. Требуется сопоставить некоторым буквам открывающиеся скобки, а некоторым – закрывающиеся таким образом, чтобы получилась правильная скобочная последовательность. Одинаковым буквам должны быть сопоставлены одинаковые скобки. Тип скобок всего один – круглые.

Входные данные

Единственная строка входного файла содержит строку из маленьких латинских букв, не более 1000 символов.

Выходные данные

В выходном файле выведите любую подходящую скобочную последовательность. Если сопоставления не существует, выведите единственную строку “IMPOSSIBLE” (без кавычек).

Пример

<i>input.txt</i>	<i>output.txt</i>
abaaba	IMPOSSIBLE
abab	() ()