

Для всех задач:

Имя входного файла:	<code>input.txt</code>
Имя выходного файла:	<code>output.txt</code>
Ограничение по памяти:	256 МБ
Максимальное количество баллов за задачу	100 баллов

Задача 1. Редактирование текста

Ограничение по времени: 1 секунда на тест

При редактировании текста некоторые распространенные ошибки можно исправить без вмешательства человека-редактора. В данной задаче участникам предлагается написать программу, исправляющую ошибки при расстановке пробелов.

При расстановке пробелов требуется использовать следующие правила:

Слова могут состоять из букв, цифр и символов апострофа и дефиса. Слово не может состоять только из дефисов и апострофов. Если символ «←» не является частью слова, он считается тире, иначе — дефисом. Тире должно иметь разделитель до и после себя.

Все разделители должны состоять из одного пробела или одного перевода строки. Разделители, состоящие более чем из одного символа пробела, заменяются на перевод строки, если содержат перевод строки, и на один пробел — иначе. В начале текста разделитель не ставится. Единственным знаком препинания, с которого может начинаться строка, является открывающая скобка. В конце текста ставится перевод строки. Если по правилам требуется вставка нового разделителя, нужно вставить один пробел.

Запятая, восклицательный знак, вопросительный знак, двоеточие, точка с запятой — обычные знаки препинания: они ставятся сразу после слова, а после них ставится разделитель, если знак препинания не стоит перед скобкой (см. ниже). Гарантируется, что в исходном тексте нет двух подряд стоящих обычных знаков препинания.

Точка считается обычным знаком препинания, если она одна. Три точки используются для обозначения многоточия и тоже считаются обычным знаком препинания. В последнем случае разделители между точками надо убирать. Гарантируется отсутствие переводов строки внутри многоточия.

Разделитель ставится до открывающей и после закрывающей круглой скобки. После открывающей скобки и перед закрывающей скобкой разделитель не ставится. Разделитель перед закрывающей скобкой не ставится, даже если перед ней стоит знак препинания. Гарантируется, что скобки образуют правильную скобочную последовательность, и что пары скобок не могут быть вложенными.

Входные данные

Исходный текст, записанный во входном файле, состоит не более чем из 20 строк, в каждой из которых не более чем 79 символов, не считая перевод строки. Текст завершается одной пустой строкой, при этом пустой является только последняя строка. Допустимыми символами являются большие и маленькие буквы латинского алфавита, цифры, символы пробела и символы, упомянутые в правилах, описанных выше.

Выходные данные

В выходной файл нужно записать результирующий текст, который должен содержать такое же количество строк, как и входной, и отличаться от исходного только добавленными или удаленными символами пробела. Строки в выходном файле могут превышать длину в 80 символов.

Пример

Для удобства прочтения символы пробела в примере ниже заменены на символы подчеркивания. В вашем решении должны использоваться символы пробела.

input.txt

```
Call_me_Ishmael.Some_years_ago_-_never_mind_how_long_precisely_-_having_little  
or_no_money_in_my_purse,and_nothing_particular_to_interest_me_on_shore,I  
thought_I_would_sail_about_a_little_and_see_the_watery_part_of_the_world._It  
is_a_way_I_have_of_driving_off_the_spleen_and_regulating_the_circulation.
```

output.txt

```
Call_me_Ishmael._Some_years_ago_-_never_mind_how_long_precisely_-_having_little  
or_no_money_in_my_purse,_and_nothing_particular_to_interest_me_on_shore,_I  
thought_I_would_sail_about_a_little_and_see_the_watery_part_of_the_world._It  
is_a_way_I_have_of_driving_off_the_spleen_and_regulating_the_circulation.
```

Задача 2. TRON

Ограничение по времени: **1 секунда на тест**

Два игрока играют в игру под названием «TRON». Происходит это следующим образом: каждый из игроков управляет точкой, двигающейся по плоскости со скоростью, равной одной единице длины в одну единицу времени. Точки могут двигаться только в четырёх направлениях, параллельных осям координат. При движении точки оставляют за собой стены. В тот момент, когда движущаяся точка врезается в стену, управлявший этой точкой игрок проигрывает.

Вам даны ломаные, представляющие собой траектории движения точек игроков. Требуется ответить на вопрос, кто из игроков проиграл, и в какой момент это произошло.

Входные данные

Первая строка входного файла содержит два целых числа N_1 и N_2 — количество вершин первой и второй ломаных соответственно ($2 \leq N_i \leq 50000$, $1 \leq i \leq 2$).

Следующие N_1 строк содержат пары целых чисел x_{1i} и y_{1i} — координаты вершин ломаной, являющейся траекторией движения точки первого игрока.

Следующие N_2 строк содержат описание траектории движения точки второго игрока в том же формате.

Гарантируется, что игроки начинают движение в разных точках.

Все координаты не превосходят 10^9 по модулю.

Выходные данные

Выведите строку **PLAYER %N% LOSES AT TIME %T%**, где **%N%** — номер проигравшего игрока (1 или 2), а **%T%** — момент времени, когда он проигрывает. Если одновременно проигрывают оба игрока, то выведите строку **ALL PLAYERS LOSE AT TIME %T%**, где **%T%** — момент времени, когда они проиграют. Если же никто не проигрывает, выведите строку **NOBODY LOSES**. Все времена выводить с одним знаком после десятичной точки, даже если это число целое.

Пример

<i>input.txt</i>	<i>output.txt</i>
3 4 1 4 2 4 3 4 3 3 3 4 3 5 3 6	PLAYER 1 LOSES AT TIME 2.0

Задача 3. Грибы

Ограничение по времени: **1 секунда на тест**

Поле состоит из квадратных клеток и имеет форму прямоугольника $N \times M$. На каждой клетке выросло некоторое количество грибов. Также в одной из клеток находится черепашка, собирающаяся совершить прогулку по полю. В каждый момент у черепашки имеется некоторое количество грибов. Изначально оно равно количеству грибов в начальной клетке. Черепашка может переползти из клетки, в которой она находится в одну из соседних по стороне клеток. Выползти за пределы поля она не может. Чтобы переползти из одной клетки в другую черепашка съедает 50 грибов. Если грибов у черепашки оказывается меньше пятидесяти, то она не может ползти дальше. Все грибы в клетке, в которую приползает черепашка, оказываются собранными ею. Таким образом, в этой клетке остается 0 грибов. Незамедлительно после того, как она покидает клетку, в этой клетке вырастает такое же количество грибов, какое в ней было раньше.

Во время прогулки черепашка хочет собрать максимальное количество грибов. При этом она желает завершить свою прогулку в некоторой заранее определённой клетке. Каждую клетку черепашка может посещать неограниченное число раз. Однако, черепашка довольно неповоротлива, в том смысле, что она не любит изменять направление своего движения. Во время прогулки она собирается совершить поворот на девяносто градусов не более чем T раз. Начальное направление своего движения черепашка может выбрать любым образом.

Вам дано описание поля, начальная и конечная клетки. Требуется найти максимально количество грибов, которое может собрать черепашка.

Входные данные

Первая строка входного файла содержит три целых числа N , M и T ($2 \leq N, M \leq 200$, $0 \leq T \leq 200$).

Вторая строка содержит пару целых чисел R_s и C_s — номер строки и номер столбца клетки, в которой черепашка находится ($1 \leq R_s \leq N$, $1 \leq C_s \leq M$).

Третья строка содержит пару целых чисел R_f и C_f — номера строки и столбца клетки, в которой черепашка планирует завершить свою прогулку ($1 \leq R_f \leq N$, $1 \leq C_f \leq M$).

Следующие N строк содержат по M неотрицательных целых чисел: i -е число j -ой строки показывает, сколько грибов растёт в клетке, находящейся на пересечении j -ой строки и i -ого столбца. Это числа не превосходят 100.

Выходные данные

В выходной файл необходимо вывести единственное число — ответ на поставленный вопрос. Если достичь желаемой клетки черепашка не может, выведите число -1 .

Пример

<i>input.txt</i>	<i>output.txt</i>
3 4 1 1 1 3 4 50 50 50 51 50 50 100 58 52 54 50 0	9

Задача 4. PVP

Ограничение по времени: **1 секунда на тест**

Два игрока играют в увлекательную компьютерную игру, сражаясь друг против друга. У каждого из них есть по герою в тяжёлой броне и с большим молотом. Требуется определить исход битвы.

Характеристики каждого из героев определяются параметрами H_0, A_0, D, p, τ . Количество пунктов здоровья изначально равно H_0 , а количество пунктов брони — A_0 . Обе этих характеристики могут уменьшаться в процессе битвы от ударов противника. Оружие характеризуется тремя неизменными параметрами: базовый урон оружия D , способность разрушать броню p , и промежуток времени между ударами Δt .

Каждый герой ударяет противника в моменты времени $0, \tau, 2\tau, 3\tau, 4\tau, \dots$. При ударе наносится реальный урон, равный $\max(D - A, 0)$, где D — базовый урон ударяющего оружия, а A — текущее количество пунктов брони противника. При этом количество пунктов здоровья противника уменьшается на величину, равную этому урону. В результате удара оружие разрушает броню противника: текущее количество пунктов его брони уменьшается на разрушающую способность оружия p . Количество пунктов брони уменьшается после нанесения урона и не может опуститься ниже нуля.

Как только количество пунктов здоровья героя становится неположительным, он погибает. Игроки могут погибнуть одновременно. Требуется определить, когда и как закончится битва.

Входные данные

В первой строке входного файла через пробел записаны параметры первого героя: H_0, A_0, D, p, τ . Во второй строке в таком же порядке записаны параметры второго героя.

Все числа целые, положительные. Ограничения: $\tau \leq 100; H_0, A_0, D, p \leq 10^{18}$.

Гарантируется, что $\max(H_0, A_0, D, p) \cdot \max(\tau) \leq 10^{18}$ (здесь максимум берётся по соответствующим характеристикам обоих героев).

Выходные данные

В первую строку выходного файла выведите момент времени, в который завершится битва.

Во вторую строку выведите два числа: количество пунктов здоровья первого и второго игроков. Если игрок погиб, требуется вывести слово **DEAD** вместо количества пунктов его здоровья.

Пример

<i>input.txt</i>	<i>output.txt</i>
15 10 10 1 3 8 8 6 5 2	9 2 DEAD

Комментарий

Ниже приведено изменение состояния героев в течение времени.

Т (время)	Н (1)	А (1)	Н (2)	А (2)	События
0	15	10	8	8	Начало
2	15	5	8	8	y2
3	15	5	6	7	y1
4	14	0	6	7	y2
6	8	0	3	6	y1 y2
8	2	0	3	6	y2
9	2	0	-1	5	y1

Система оценки:

Решение, корректно работающее при дополнительном ограничении $H_0, A_0, D, p \leq 10^6$, набирает 50 баллов.

Задача 5. Система уравнений

Ограничение по времени: **1 секунда на тест**

Дана система уравнений с параметром a :

$$\begin{cases} y = x^3; \\ y = |x - a| - 1; \end{cases}$$

Требуется найти решение этой системы для заданного значения a .

Входные данные

Во входном файле записано одно вещественное число a , заданное не более чем с двумя знаками после десятичной точки, по модулю не превосходящее 10^3 .

Выходные данные

В первую строку входного файла необходимо вывести одно целое число — количество решений заданной системы уравнений. Во вторую строку через пробел нужно вывести значения x , которые являются решениями, в порядке возрастания. Вещественные числа выводить с точностью не менее чем 10^{-5} .

Пример

<i>input.txt</i>	<i>output.txt</i>
-7	1 2.0000000