

Задача А. Тренировка о двух концах

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Вместо предисловия

Так уж сложилось исторически, что самой простой программой на любом языке программирования считается “Hello, world!”, а самой простой олимпиадной задачей — задача $A+B$. Сегодняшние задачи будут про двух товарищей, которых мы условно назовём A и B . Любые совпадения с реальными событиями и реальными товарищами следует считать случайными.

Приближается очередная олимпиада по программированию, и товарищи A и B решили устроить совместную тренировку. Они выбрали n задач и договорились, что тренировка продлится t минут.

Для каждой задачи известно время s_i ($i = 1, 2, \dots, n$), которое потребуется на её решение. Уровень подготовки у товарищей A и B одинаковый, и если товарищ A решает некоторую задачу за время s , то и товарищ B решает эту задачу за это же время s .

Товарищ A начал решать задачи с первой и будет решать их подряд, двигаясь от первой к последней и не пропуская ни одной. Товарищ B начал решать задачи с последней и тоже будет решать их подряд, двигаясь от последней к первой и не пропуская ни одной.

Ваша же задача — определить, сколько задач решат в течение тренировки товарищи A и B . В качестве ответа выведите два целых числа: количество задач, которые решит товарищ A , и через пробел — количество задач, которые решит товарищ B .

Формат входного файла

В первой строке содержатся через пробел два целых числа — n ($1 \leq n \leq 100$) и t ($1 \leq t \leq 1000$).

Во второй строке содержатся n целых чисел s_1, s_2, \dots, s_n ($1 \leq s_i \leq 1000, i = 1, 2, \dots, n$), также отделённые друг от друга пробелом.

Формат выходного файла

В единственной строке выходного файла содержатся два целых числа через пробел — количество задач, которые решит товарищ A , и количество задач, которые решит товарищ B соответственно.

Примеры

| <code>input.txt</code> | <code>output.txt</code> |
|-------------------------|-------------------------|
| 5 150 10 30 80 50 25 | 3 2 |
| 3 100 30 32 31 | 3 3 |

Задача В. Зефир и мороженое

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Товарищ A очень любит мороженое, а товарищ B очень любит зефир в шоколаде. Съедая одну порцию любимого лакомства, каждый из них получает $+1$ к удовольствию.

Однако товарищи A и B обладают достаточной вредностью и не просто хотят получить удовольствие сами, но и хотят, чтобы другой товарищ при этом получил как можно меньшее удовольствие.

Назовем истинным наслаждением разницу между полученным удовольствием некоторого товарища и полученным удовольствием другого товарища. Таким образом, товарищи A и B стремятся получить как можно больше истинного наслаждения. Поэтому каждый из них может съесть не только своё любимое лакомство, но и любимое лакомство другого товарища — чтобы тому меньше досталось.

На поедание порции мороженого каждый из товарищей тратит a минут, а на поедание порции зефира — b минут. Как только какой-либо товарищ доедает порцию какого-либо лакомства, он сразу же принимается за поедание следующей порции того же или другого лакомства — на его выбор. Если товарищам приходится делать выбор одновременно, то первым всегда выбирает товарищ A .

У товарищей A и B есть m порций мороженого и z порций зефира, и они намерены съесть их все. Определите, кто из них получит большее истинное наслаждение и каким оно будет, если каждый из них будет действовать оптимально.

Формат входного файла

В первой строке через пробел записаны числа m и z ($0 \leq m, z \leq 100$) — количество порций мороженого и количество порций зефира.

Во второй строке через пробел записаны числа a и b ($1 \leq a, b \leq 100$) — время, которое необходимо для поедания порции мороженого и порции зефира соответственно.

Формат выходного файла

В первой строке выведите через пробел символ латинского алфавита A или B — имя товарища, который получит большее истинное наслаждение, и целое число — истинное наслаждение, которое он получит.

Если товарищи получают одинаковое истинное наслаждение, выведите перед целым числом символ A .

Примеры

| <code>input.txt</code> | <code>output.txt</code> |
|------------------------|-------------------------|
| 2 3 5 2 | A 0 |
| 1 3 5 2 | B 1 |

Задача С. Предчувствие континуального интеграла

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Улицы города C образуют прямоугольную сетку с единичным шагом. В некоторых узлах этой прямоугольной сетки есть остановки автобусов. Все автобусные маршруты в городе являются кольцевыми, и каждый из них может быть описан последовательностью пар целых чисел (каждая пара определяет остановку). В состав одного маршрута любая остановка входит не более одного раза. Одна и та же остановка может входить в состав разных маршрутов.

Товарищ A любит ходить в гости к товарищу B . Поскольку товарищ A живет рядом с остановкой, имеющей координаты (ax, ay) , то он выходит на эту остановку, чтобы отправиться к товарищу B . Место жительства товарища B имеет координаты (bx, by) .

Товарищ A совершенно уверен, что он доберётся быстрее, если хотя бы часть пути проедет на автобусе. Когда он приходит на свою остановку, то садится в первый же пришедший автобус и выходит на той остановке, которая ближе всего расположена к месту жительства товарища B и при этом не совпадает с остановкой (ax, ay) . После этого он идет к месту жительства товарища B пешком вдоль улиц кратчайшим путём.

Известно, что остановка с координатами (ax, ay) входит в состав n автобусных маршрутов. Товарищ A заметил, что последние n раз, когда он отправлялся в гости к товарищу B , он ровно единожды ездил каждым маршрутом автобуса. Теперь он хочет узнать, какое суммарное расстояние он прошёл за эти n походов в гости. Ваша задача — определить это расстояние.

Формат входного файла

В первой строке через пробел содержатся числа n ($1 \leq n \leq 1000$), ax , ay , bx , by ($0 \leq ax, ay, bx, by \leq 1000$).

В каждой из следующих n строк содержится описание очередного автобусного маршрута вида

$r_j, sx_1, sy_1, sx_2, sy_2, \dots, sx_{r_j}, sy_{r_j}$,

где r_j ($2 \leq r_j \leq 1000, j = 1, 2, \dots, n$) — количество остановок, составляющих маршрут автобуса, а (sx_i, sy_i) ($0 \leq sx_i, sy_i \leq 1000, i = 1, 2, \dots, r_j$) — координаты остановок в том порядке, в котором автобус их проезжает.

Гарантируется, что для каждого маршрута существует такое k_j , что $sx_{k_j} = ax$ и $sy_{k_j} = ay$.

Формат выходного файла

В первой строке выведите единственное целое число — суммарное расстояние, которое прошёл товарищ A за n походов в гости к товарищу B .

Примеры

| input.txt | output.txt |
|--|------------|
| 3 5 2 10 12 5 6 4 5 2 3 7 14 8 12 15 4 7 2 8 14 5 2 12 12 6 4 3 3 2 2 5 2 8 8 3 5 2 | 18 |
| 2 5 2 10 12 4 3 2 4 10 5 2 11 11 6 14 13 10 12 2 5 2 8 8 4 5 2 | 2 |

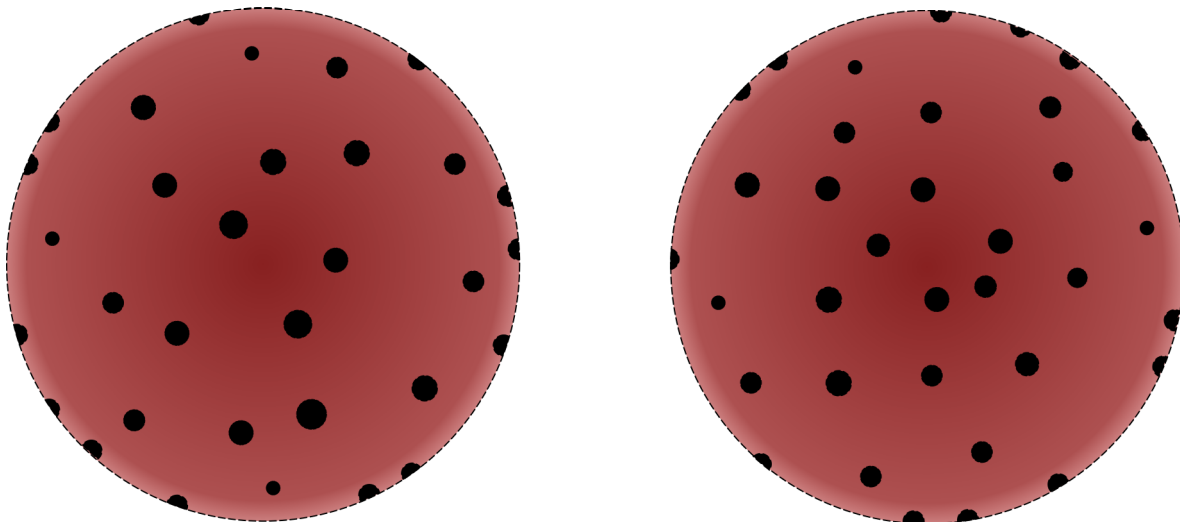
Задача D. Сон о сферической корове

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Иногда товарищи A и B видят один сон на двоих. В последнем таком сне им приснилась огромная Сферическая Божья Коровка. Она очень грустила от того, что никто никогда не хотел нарисовать её портрет. Товарищи A и B пообещали ей, что, как только проснутся, сразу же нарисуют её портрет. А Сферическая Божья Коровка пообещала им, что в следующем сне она научит их летать.

Обещания нужно выполнять, и с самого утра товарищи A и B собрались рисовать портрет. Однако смотрели они на Сферическую Божью Коровку с противоположных сторон, и поэтому каждый мог наблюдать только одну полусферу. Они решили для начала построить 3D-модель Сферической Божьей Коровки на компьютере, и каждый нарисовал карту той полусферы, которую видел.

Будем считать, что карта полусферы — это круг с центром в начале координат и радиусом r . Все пятна Сферической Божьей Коровки будем считать точками, каждая из которых на карте задана своими координатами (конечно, на самом деле это круги, только очень-очень маленькие). Получилось так, что некоторые пятна были видны и товарищу A , и товарищу B , и потому оказались как на карте одной полусферы, так и на карте другой полусферы. На рисунке можно видеть примеры карт, нарисованных товарищами A (слева) и B (справа).



Чтобы получить модель, нужно совместить нарисованные карты. Для этого достаточно указать одну пару точек (на границе), которые соответствуют одному и тому же пятну Сферической Божьей Коровки. Ваша задача — найти такую (любую) пару точек.

Формат входного файла

В первой строке записаны целые числа r ($10 \leq r \leq 10^6$), n и m ($1 \leq n, m \leq 1000$) — количество пятен, которые увидели товарищ A и товарищ B соответственно.

Далее следуют $n + m$ строк, содержащих описания пятен.

В каждой из следующих n строк содержатся по два целых числа x_j, y_j ($j = 1, 2, \dots, n$) — координаты пятен на карте товарища A .

Далее в каждой из следующих m строк содержатся по два целых числа x_k, y_k ($k = 1, 2, \dots, m$) — координаты пятен на карте товарища B .

Все координаты пятен корректны, карты допускают возможность совмещения. На каждой карте никакие два пятна не совпадают. Гарантируется, что по крайней мере одно пятно лежит на границе обеих карт.

Формат выходного файла

В первой строке выведите два целых числа — номер точки на карте товарища *A* и номер точки на карте товарища *B*, которые представляют одно и то же (любое) пятно Сферической Божьей Коровки.

Примеры

| input.txt | output.txt |
|--|------------|
| 13 2 2 12 5 5 12 12 5 5 12 | 1 2 |
| 17 4 6 -1 14 0 17 2 12 5 -11 3 8 -6 12 5 12 -8 -15 -5 -11 0 16 | 2 4 |

Задача Е. Подъёмная сила

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

В следующем сне товарищи *A* и *B* узнали от Сферической Божьей Коровки, что летать очень легко. Для этого достаточно владеть заклинанием, состоящим всего из одного магического слова. А поскольку Сферической Божьей Коровке понравился ее портрет, и она пребывала в чрезвычайно хорошем расположении духа, то она сообщила товарищам *A* и *B* это магическое слово.

И, действительно, как только товарищи *A* и *B* узнали это заклинание, они сразу же смогли подняться на пару десятков сантиметров над землей. Товарищи *A* и *B* спросили у Сферической Божьей Коровки, нельзя ли как-нибудь подняться повыше. Она ответила, что можно подниматься очень высоко, но для этого нужно использовать более сильное заклинание. Сила заклинания определяется тем, сколько раз в него входит магическое слово. Вот только Сферическая Божья Коровка не может запоминать длинные заклинания, и поэтому ей приходится летать достаточно низко.

Товарищи *A* и *B* могут запомнить n символов, и теперь они хотят придумать заклинание, состоящее из не более чем n символов и содержащее как можно больше вхождений магического слова. Ваша задача — найти такое заклинание.

Если возможно составить несколько таких заклинаний, содержащих максимально возможное количество вхождений магического слова, выведите в качестве ответа самое короткое из них. Если же их длины равны, выведите любое.

Формат входного файла

В первой строке содержится магическое слово — непустая последовательность из не более чем 1000 строчных латинских букв.

Во второй строке содержится целое число n ($n \leq 10^5$) — максимально возможная длина заклинания. Гарантируется, что максимально возможная длина заклинания не меньше длины магического слова.

Формат выходного файла

В первой строке выведите целое число — фактическую длину заклинания.

Во второй строке выведите само заклинание.

Примеры

| <code>input.txt</code> | <code>output.txt</code> |
|------------------------|-------------------------|
| abacaba 13 | 11 abacabacaba |
| baa 9 | 9 baabaabaa |