

## Задача 1. Путь к Пьяной Берлоге

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда на тест  
Ограничение по памяти: 64 Мб

Румата Эсторский изрядно поиздержался и теперь направляется в сторону Пьяной Берлоги, где размещён аппарат по изготовлению золотых монет. По дороге от столицы Арканара к Пьяной Берлоге есть несколько постоянных дворов, в которых можно пополнить запас корма для хамахарского жеребца. Но так как арканарская экономика находится на очень низкой стадии развития, то цена на корм изменяется в довольно широких пределах.

Вы — программист экспедиции Института Экспериментальной истории, который осуществляет масштабное изучение ситуации в Арканаре и в других феодальных государствах этой планеты. И вам поручили написать программу, которая позволит составить маршрут для Руматы так, чтобы он преодолел расстояние от столицы до Пьяной Берлоги, затратив как можно меньше денег.

### Входные данные

В первой строке входного файла заданы четыре числа  $S$ ,  $M$ ,  $V$  и  $N$ , разделенные пробелам, где  $M$  — вещественное число, остальные целые:  $S$  — это расстояние в километрах по дороге между столицей и Пьяной Берлогой ( $100 \leq S \leq 2000$ ),  $M$  — количество корма в килограммах, требуемое хамахарскому жеребцу на один километр, заданное не более чем с двумя знаками после десятичной точки ( $0.01 \leq M \leq 10$ ),  $V$  — максимальный запас корма в килограммах, который может взять с собой благородный дон, чтобы не стать похожим на торговца сеном, ( $10 \leq V \leq 50$ ),  $N$  — количество постоянных дворов по дороге ( $1 \leq N \leq 10^5$ ).

В следующих  $N$  строках для каждого из постоянных дворов указано два целых числа  $S_i$  и  $C_i$ , где  $S_i$  — расстояние по дороге от столицы до него в целых километрах, а  $C_i$  — цена корма на этом дворе в золотых за килограмм ( $0 \leq S_i \leq S$ ,  $1 \leq C_i \leq 100$ ,  $1 \leq i \leq N$ ). Ближайший к столице постоянный двор находится на расстоянии 0 километров от неё. Гарантируется, что поездка всегда возможна, то есть максимального запаса корма хватит на то, чтобы преодолеть расстояние между любыми двумя соседними постоянными дворами и между последним перед лесом постоянным двором и Пьяной Берлогой. На постоянных дворах могут продать корм с точностью до 10 граммов, цена корма всегда исчисляется с точностью до сотых долей золотого.

### Выходные данные

В выходной файл необходимо вывести одно вещественное число не более чем с двумя знаками после десятичной точки — минимально возможную стоимость корма, требуемого для того, чтобы добраться до Пьяной Берлоги, в золотых.

### Примеры

<i>input.txt</i>	<i>output.txt</i>
100 0.1 10 2 0 50 50 20	350
100 0.12 10 3 78 91 0 45 33 34	451.56



## Задача 2. Финский язык

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда на тест  
Ограничение по памяти: 64 МБ

— Что же, Вы считаете, что несколько провалов опытных прогрессоров в этом мире вызваны неосторожностью при передаче информации?

— Да, я так считаю, — ответил прибывший на корабль эксперт по структурной лингвистике. — Вы используете для общения между собой русский и английский языки, не понимая того, что для этого мира само построение фраз на этих языках странно. И когда знатный купец из одной страны и вельможа из другой общаются на вообще неведомом в этом мире языке — случайно услышавший разговор тут же заподозрит в них книжечеев. Или того хуже, исчадий ада.

— И что делать?

— Использовать для общения между прогрессорами финский язык. Он чем-то напоминает наречие варваров, и его легко принять за незнакомый диалект. Поверьте, опасность для тех, кто работает на Арканаре, существенно снизится.

С учётом мнения экспертов руководителем программы было принято решение о форсированном обучении прогрессоров на Арканаре финскому языку по специальной методике. Вам поручено написать один из модулей ПО для обучающей системы. В техническом задании были те сведения по финскому языку, которые нужны для Вашей части работы.

В финском языке слог может быть образован одним из следующих способов:

- краткой гласной **a, e, i, o, u, y, ä, ö**;
- долгой гласной **aa, ee, ii, oo, uu, yy, ää, öö**;
- дифтонгом **ai, au, ei, eu, ie, iu, oi, ou, ui, uo, yi, yö, äi, äy, öi, öy**.

Рядом с дифтонгом не может стоять никакая гласная, два дифтонга подряд тоже не встречаются. После долгой гласной может стоять одна краткая гласная, в этом случае они образуют два слога. В слове могут встречаться подряд две разные краткие гласные. Если их сочетание не является дифтонгом, тогда это тоже два слога.

Множественное число слов, стоящих в некоторых косвенных падежах<sup>1</sup>, образуется так:

**<слабая\_основа\_единственного\_числа> + i + <падежное\_окончание>**

Косвенные падежи, образуемые по этому правилу, имеют следующие окончания:

Аблатив ( <b>Ab</b> l)	<b>-lta/-ltä</b>
Адессив ( <b>Ade</b> )	<b>-lla/-llä</b>
Аллатив ( <b>All</b> )	<b>-lle</b>
Инессив ( <b>Ine</b> )	<b>-ssa/-ssä</b>
Транслатив ( <b>Tra</b> )	<b>-ksi</b>
Элатив ( <b>Ela</b> )	<b>-sta/-stä</b>

Если в исходном слове присутствует хотя бы одна из гласных **ä, ö** или **y**, либо все гласные в слове — это только **i** и **e** в любых сочетаниях, то в падежном окончании используется диэрезис (т.н. *правило гармонии гласных*, которому подчиняются все гласные в слове).

Диэрезис — это надстрочный знак, выглядит, как две точки, расположенные над буквой горизонтально, например, в русском языке диэрезис используется в букве **ë**.

Гласное окончание основы (им может быть любая гласная или дифтонг) может изменяться. Изменения подчиняются следующим правилам (всегда применяется только одно правило):

- 1) краткие гласные **u, y, o, ö** не изменяются;
- 2) краткая гласная **e** исчезает;
- 3) краткая гласная **i** превращается в **e**;

<sup>1</sup> В финском языке 15 падежей. Из них некоторые имеют аналоги среди падежей русского языка (ср. номинатив – именительный, генитив – родительный, аккузатив – винительный), другие – нет.

- 4) долгие гласные сокращаются до кратких;  
 5) в дифтонгах **uo, yö, ie** первая гласная исчезает, остальные дифтонги не изменяются;  
 6) краткие гласные **a, ä** либо исчезают, либо изменяются на **o, ö** соответственно. Замена происходит в следующих случаях:
- если первый слог двусложного слова, оканчивающегося на краткую гласную **a**, образован краткой или долгой гласной **a, e, i, aa, ee, ii**;
  - если многосложное слово (три и более слогов) оканчивается на **-kka/kkä, -la/lä, -na/nä, -ija/ijä**. Нужно помнить, что перед согласной буквой, на которую начинается какое-либо из указанных окончаний, должна идти гласная. Таким образом, данное правило не относится к словам, оканчивающимся, например, на **-lla/llä**.

Напишите программу, которая по заданной основе единственного числа будет строить слово в указанном косвенном падеже, во множественном числе (в соответствии с описанным правилом).

### Входные данные

В первой строке входного файла записано одно число  $N$  ( $1 \leq N \leq 20$ ). Далее в каждой из  $N$  строк записано по два слова: трехбуквенное обозначение падежа и слабая основа единственного числа, длина которой не превышает 16 букв.

### Выходные данные

В выходной файл нужно вывести  $N$  строк, каждая из которых является соответствующим словом в указанном падеже, во множественном числе.

Правило записи диэрезисов: Во входном и выходном файлах буквы **ä** и **ö** должны быть представлены парой символов: **a:** и **o:** соответственно.

### Пример

<i>input.txt</i>	<i>output.txt</i>	<i>Примечание</i>
12	taloissa	<i>talo</i> – дом, <i>taloissa</i> – в домах
Ine talo	ja:rvilla:	<i>järvi</i> – озеро, <i>järvillä</i> – на озерах
Ade ja:rve	tuoleilla	<i>tuoli</i> – стул, <i>tuoleilla</i> – на стульях
Ade tuoli	huoneista	<i>huone</i> – комната, <i>huoneista</i> – из комнат
Ela huonee	mailta	<i>maa</i> – земля, <i>mailta</i> – с земель
Abl maa	soista	<i>suo</i> – болото, <i>soista</i> – из болот
Ela suo	to:issa:	<i>työ</i> – работа, <i>töissä</i> – на работах
Ine tyo:	teilla:	<i>tie</i> – дорога, <i>teillä</i> – на дорогах
Ade tie	koirille	<i>koira</i> – собака, <i>koirille</i> – собакам
All koira	teemoilla	<i>teema</i> – тема, <i>teemoilla</i> – по темам
Ade teema	ikkunoista	<i>ikkuna</i> – окно, <i>ikkunoista</i> – из окон
Ela ikkuna	vena:la:isilta:	<i>venäläinen</i> – русский, <i>venäläisiltä</i> – от русских
Abl vena:la:ise		

### Задача 3. Антропологические исследования

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 4 секунды на тест  
Ограничение по памяти: 64 МБ

Для антропологических изысканий в рамках исследовательской программы Института Экспериментальной Истории было необходимо собрать статистическую информацию по разным удалённым племенам, проживающим на территории герцогства Ируканского.

Задействовать для этого квалифицированных прогрессоров было бы излишним риском. Был найден другой способ. Через дону Гуга герцогу Ируканскому была подброшена идея о переписи всех потенциальных солдат и составлении реестра их по росту (чтобы в случае войны знать, куда можно будет распределить ополченцев).

И вот в отдалённое племя, в котором было  $N$  боеспособных мужчин, были посланы писари из штаба с указанием провести сравнительный анализ роста. Увы, как свойственно военным, они взялись за выполнение этой задачи буквально. Они или сами измеряли разницу в росте двух человек, иногда на глазок, или записывали со слов самих аборигенов, которые информацию давали в виде: «Вася ниже Пети, когда Петя в кепке».

В результате были заполнены таблички в форме

«Vasja shorter Petja with C» ( «Вася ниже Пети с добавкой C см») или

«Vasja taller Petja at least C» ( «Вася выше Пети не менее чем на C см»).

Во избежание разглашения секретных военных сведений герцогства Ируканского все имена заменены.

Возможно, попарно измерить всех они не успели. И вот такой вот ворох записей они принесли в штаб. Получив информацию, координатор проекта (представляя, что ему скажут по поводу таких вот “результатов” антропологи), решил уменьшить количество записей, не искажая картину соотношений роста между людьми.

Ваша задача — составить описание соотношения роста исследуемого населения с использованием минимального количества записей. При этом можно либо выбирать уже существующую запись, либо дописывать новую, если она не противоречит уже имеющимся записям и не искажает картину.

#### Входные данные

В первой строке входного файла записаны через пробел два целых числа  $N$  и  $M$  — количество исследуемых людей и количество собранных записей, соответственно ( $2 \leq N \leq 300$ ,  $1 \leq M \leq 90000$ ).

В следующих  $M$  строках идут сравнительные записи, по одной на строку, в следующем формате:

`name1 shorter name2 with C`

`name1 taller name2 at least C`

Здесь `name2`, `name1` — различные имена, написанные латинскими буквами, длина имени не более 10 символов, большие и маленькие буквы не различаются;  $C$  — целое число ( $0 \leq C \leq 100$ ).

#### Выходные данные

В первую строку выходного файла необходимо вывести количество записей в минимальном наборе. В следующих строках должны находиться сами записи из минимального набора, по одной на строку в том же формате, что и во входном файле. Числа должны быть неотрицательными. В именах большие и маленькие буквы не различаются.

Если собранная информация оказалась противоречивой, то нужно вывести число `-1`.

### Примеры

<i>input.txt</i>	<i>output.txt</i>
3 3 Petja taller Vasja at least 2 Ivan shorter Vasja with 4 Ivan shorter Petja with 8	2 Ivan shorter Vasja with 4 Petja taller Vasja at least 2
2 2 Petja taller Vasja at least 3 Petja shorter Vasja with 2	-1

## Задача 4. Король Арканара и шахматы

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда на тест  
Ограничение по памяти: 64 Мб

Пиц Шестой, король Арканарский, посмотрел на гостя с интересом.

“А, Румата Эсторский! Давненько я вас не видел! Мне тут про вас много чего рассказывали...”

“Интересно, что ему могли про меня нашептывать?” - подумал Антон. “Скорее всего, дон Рэба постарался представить меня в худшем свете...”

“Но всё это чепуха. Я всего и не запомнил, но вот сказали мне, что вы в шахматы хорошо играете. Тут вот два моих министра как-то раз играли партию, и у них вышел спор. Один говорит, что тут есть мат не более, чем в два хода. А второй – что ничего подобного. С тех пор они уже дрались три раза, но вопрос так и не решили. Я слышал, Вы просите за одного грамотея?”

“Видите ли, Ваше Величество, этот шельмец превосходно разбирается в шахматах. И я бы хотел – до того, как сам вздёрну негодея – сыграть с ним несколько партий...”

“Вот и отлично. Разрешите спор – получите этого пройдоху. Но если ошибётесь... я могу и рассердиться!” - подытожил король.

Центр решил подстраховать Антона и предложил Вам написать программу, которая по заданной позиции и очередности хода определяет, можно ли форсированно (т.е. независимо от игры соперника) поставить мат в 1 ход или мат в 2 хода.

Правила игры в шахматы, принятые в Арканаре:

- игра происходит на доске размером 8x8 клеток; поля (клетки) нумеруются по горизонтали латинскими буквами от **a** до **h** слева направо, по вертикали цифрами от 1 до 8 снизу вверх;
- играют двое, один белыми фигурами, другой – черными;
- ходы делаются игроками поочередно, пропускать ходы не разрешается;
- за один ход двигать можно только одну свою фигуру;
- ни одна фигура, кроме коня, не может перепрыгивать через свои или чужие фигуры, конь может это делать;
- нельзя делать ход на поле, занятое своей фигурой; при ходе на поле, занятое чужой фигурой, она снимается с доски (взятие);
- в начальном положении каждый игрок имеет 1 короля, 1 ферзя, 2 слона, 2 коня, 2 ладьи и 8 пешек своего цвета;
- в начальном положении всю 2-ую горизонталь занимают белые пешки, а 7-ую – соответственно, черные;
- фигуры ходят следующим образом:
  - 1) ферзь – на любое расстояние по вертикали, горизонтали или диагонали;
  - 2) ладья – на любое расстояние по вертикали или горизонтали;
  - 3) слон – на любое расстояние по диагонали;
  - 4) конь – ходит буквой «Г»: на 1 клетку по горизонтали и на 2 по вертикали, или наоборот, на 1 клетку по вертикали и 2 по горизонтали;
  - 5) король – на одну клетку по вертикали, горизонтали или диагонали;
  - 6) пешка – ходит со взятием и без взятия по-разному:
    - без взятия пешка может сделать первый ход на 1 или 2 клетки вперед (для белых вперед значит в направлении возрастания номера строки, для черных – убывания), а последующие ходы – только на 1 клетку вперед;
    - со взятием – на одну клетку левее или правее той, что впереди текущей позиции пешки (т.е. на одну клетку вперед по диагонали);
    - если пешка доходит до противоположной стороны доски (для белых – до 8-ой горизонтали, для черных – до 1-ой горизонтали), ходящий должен заменить её на любую другую фигуру того же цвета (по его желанию), кроме короля, что считается частью этого же хода;

НГУ, 12 ноября 2006 года

- правило "**взятие пешкой на проходе**": если пешка совершила свой первый ход на две клетки и при этом непосредственно слева или справа от нее находится вражеская пешка, то на следующем и только на следующем ходу эта вражеская пешка может взять ходившую пешку, при этом переместившись на клетку между старой и новой позицией ходившей пешки (например, белые ходят Pe2-e4, на что черные могут ответить Pd4-e3 или Pf4-e3, удаляя белую пешку e4), в начальной позиции взятие на проходе невозможно;
- поле называется **находящимся под ударом** противника, если при своём ходе противник мог бы пойти на это поле;
- король называется **стоящим под шахом**, если при своём ходе противник мог бы взять его;
- запрещены ходы, ставящие своего короля под шах или оставляющие его под шахом;
- если игрок не может сделать ход, поскольку после любого хода его король будет стоять под шахом, партия на этом заканчивается. При этом:
  - если король игрока, который не может ходить, стоит под шахом, этот игрок объявляется проигравшим. Такая ситуация называется мат.
  - если король игрока, который не может ходить, не стоит под шахом, и при этом другие фигуры или пешки также не имеют ходов, партия считается закончившейся ничью. Такая ситуация называется пат.

Для тех, кто знает что такое "рокировка": в Арканаре её нет, так что за такое здесь считают шулером и бьют канделябрами.

### Входные данные

Первые две строки входного файла задают позиции белых и черных фигур соответственно. Фигуры описываются через запятую без лишних символов и пробелов. В третьей строке задается текущая очередь хода: **W** – ход белых, **B** – ход черных. Для описания фигур используются следующие буквы: пешка – **P** (Pawn), конь – **N** (kNight), слон – **B** (Bishop), ладья – **R** (Rook), ферзь – **Q** (Queen), король – **K** (King). Например, **Kd6** обозначает короля на поле **d6**.

### Выходные данные

Если ход белых (черных), и они могут поставить мат противнику следующим ходом, вывести в первой строке: **White (Black) wins in 1 move(s)**.

Если же они не могут поставить мат следующим ходом, но после своего первого хода на любой ответный ход противника существует матующий ход, вывести в первой строке: **White (Black) wins in 2 move(s)**. Если же таких ходов не существует вывести в первой и единственной строке: **No solution**. Во второй строке через пробел нужно вывести в лексикографическом порядке все начальные ходы, форсированно ведущие к выигрышу начинающего игрока. Ход обозначается буквой фигуры, начальное поле, дефис, конечное поле. Если пешка дошла до своей последней горизонтали, к ходу приписывается буква новой фигуры (например, Ph7-h8Q – пешка белых "превращается" в ферзя по вертикали **h**).

### Примеры

<i>input.txt</i>	<i>output.txt</i>
Kd6, Qd5 Kd8 W	White wins in 1 move(s). Qd5-a8 Qd5-g8
Kd6, Qd5 Kd8 B	No solution.
Kd6, Ph7 Kd8 W	White wins in 1 move(s). Ph7-h8Q Ph7-h8R
Kd6 Kd8 W	No solution.
Kb6, Ph6 Ka8 W	White wins in 2 move(s). Ph6-h7





### Входные данные

В первой строке входного файла заданы два целых числа  $N$  и  $K$ , где  $N$  — общее количество благородных донов в трактире,  $K$  — количество бутылок перед каждым доном ( $1 \leq N \leq 1000$ ,  $1 \leq K \leq 20$ ).

В следующих  $N$  строках приводятся через пробел по  $K$  наборов следующего вида:  $+/-i$ , где, соответственно "+", если бутылка была открыта и выпита, и "-", если не была открыта,  $i$  — количество сломанных штопоров ( $0 \leq i \leq 200$ , если число сломанных штопоров равно нулю, то оно может быть опущено). Если бутылка была открыта, то затем через пробел указывается время ее открытия в формате *ч.мм.сс*, начиная с момента начала соревнования. Никакие два благородных дона не открывали бутылки в один и тот же момент времени.

### Выходные данные

В выходной файл нужно вывести  $N$  строк по 2 целых числа в строке, разделенных пробелом. В каждой строке указывается для соответствующего благородного дона его наивысшее место в текущем моментальном рейтинге и суммарное время в секундах, которое он находился на этом месте. Секунда, на которой благородный дон откупорил бутылку, учитывается.

### Пример

<i>input.txt</i>	<i>output.txt</i>
2 3	1 12001
-10 +2 0.00.05 + 1.59.59	1 16797
-0 +1 0.20.04 +3 1.00.00	

## Задача 6. Таинственный замок

Имя входного файла: *input.txt*  
Имя выходного файла: *output.txt*  
Ограничение по времени: 10 секунд на тест  
Ограничение по памяти: 64 МБ

Дон Рипат, верный агент дона Руматы, сообщил ему про предстоящую тайную встречу дона Рэбы и предводителя разбойников Ваги Колеса. Судя по всему, обсуждаться должно было что-то достаточно важное.

“Вероятно, стоит попробовать подслушать разговор” — решил Антон.

Но на двери, ведущей в покои дона Рэбы, висел замок странной конструкции. Это был один из знаменитых ируканских замков. Один такой замок стоил целого замка. На замке была нарисована полоска из  $N$  равных квадратиков, в открытом состоянии последовательно меняющих свой цвет от белого к чёрному. При закрывании квадратики случайно расставлялись. Можно было взять несколько подряд идущих квадратиков и развернуть весь этот прямоугольник на 180 градусов — механизм замка позволял это сделать для любого числа квадратиков. Замок открывался приведением полоски в исходное состояние.

Сразу же после того, как информация поступила на базу, Вам была поставлена задача написать программу, которая бы открывала подобный замок за минимальное число разворотов.

### Входные данные

Во входном файле в первой строке задано число  $N$  ( $1 \leq N \leq 12$ ) — количество квадратиков в полоске замка. В следующей строке задана начальная конфигурация замка.  $N$  различных чисел от 1 до  $N$ , соответствующие определенному оттенку серого цвета, записаны через пробел друг за другом.

### Выходные данные

В первую строку выходного файла вывести число  $M$  — минимальное количество разворотов, с помощью которых можно открыть замок. Во вторую строку вывести исходную конфигурацию замка. Далее должны следовать  $M$  строк, каждая из которых должна содержать очередную конфигурацию замка, полученную из предыдущей с помощью одного разворота. Таким образом, в последней строке выходного файла всегда должна быть конфигурация, соответствующая исходному состоянию замка.

### Пример

<i>input.txt</i>	<i>output.txt</i>
5	3
3 5 2 4 1	3 5 2 4 1
	3 2 5 4 1
	3 2 1 4 5
	1 2 3 4 5



## Задача 7. Рыцарский турнир

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 4 секунды на тест  
Ограничение по памяти: 64 Мб

Барон Пампа решил принять участие в рыцарском турнире, проводимом в Торговой республике Соан. Турнир был необычайно популярен из-за необычных правил соревнований и богатых призов победителям.

Необычность правил была в том, что турнир проходил с отбором, в котором претенденты должны были показать искусство владения мечом. И только те, кто прошёл этот отбор, попадали в основные состязания. Таким образом, зрителям было гарантировано настоящее сражение за победу, а победитель по праву мог считать себя лучшим из лучших.

Антон, которому надо было проконсультироваться по поводу ситуации в Арканаре с доном Кондором, работавшим под легендой одного из высших чиновников торговой республики Соан, согласился сопровождать барона. Так что тренером барона Пампы был объявлен непревзойдённый мастер битвы на мечах дон Румата Эсторский.

По дороге барон заявил, что не имеет ни малейшего представления о том, какие элементы боя он должен показать судьям. “Я воин, а не писарь!” — кричал он, распугивая встречных прохожих. Уже на территории Республики Соан Пампа успокоился и решил всё же посоветоваться со своим тренером по поводу тактики.

Румата тут же составил список умений барона:

- количество элементов, которые он должен выполнить в течение программы;
- список номеров элементов, которые он умеет выполнять.

Пронумеровав все элементы, начиная с 1, Антон обнаружил, что существует всего  $N$  элементов. Кроме того, в таблицу были занесены следующие сведения о каждом  $i$ -м элементе:

- стоимость в баллах за правильное выполнение  $G_i$ ;
- штраф за ошибку  $W_i$ , не превышающий  $G_i$  (считаем, что при выполнении элемента возможна только одна ошибка, штраф за любую ошибку одинаков);
- если возможно исполнение связки двух последовательных элементов с номерами  $i$  и  $j$ , то известна вероятность  $P_{ij}$  ошибки во втором элементе. И тогда ожидаемая оценка за исполнение  $j$ -го элемента считается как  $G_j \cdot (1 - P_{ij}) + (G_j - W_j) \cdot P_{ij}$

Ожидаемая оценка за программу получается суммированием ожидаемых оценок за каждый элемент. Считается, что первый элемент программы всегда исполняется без ошибок.

При первом взгляде на схему стало понятно, что без компьютерных расчётов тут не обойтись. Так что во время встречи двух резидентов с Земли схема была передана на борт.

Руководство решило пойти навстречу, тем более, что победа барона на турнире несколько отвлекла бы слишком пристальное в последнее время внимание высокопоставленных арканарцев от персоны дона Руматы. Так что Вам поручили написать программу для выбора последовательность элементов, при которой ожидается максимальное количество баллов для барона Пампы.

### Входные данные

В первой строке входного файла указаны целые числа  $N$  и  $K$ , где  $N$  — количество элементов, которые барон умеет выполнять,  $K$  — количество элементов, которые он должен выполнить в течение программы ( $1 \leq K \leq N \leq 300$ ). В следующих  $N$  строках файла для каждого элемента указано по два целых числа: стоимость в баллах от 1 до 10 за правильное исполнение и штраф за ошибку.

Далее в  $N$  строках, в порядке возрастания  $i$ , на отдельной строке для каждого  $i$  заданы либо вещественные числа  $P_{ij}$  вероятности ошибки при исполнении  $j$ -го элемента после  $i$ -го, либо  $-1$ ,

если исполнение  $j$ -го элемента после  $i$ -го невозможно. Все числа заданы не более чем с двумя знаками после десятичной точки.

### Выходные данные

Выходной файл должен содержать одно вещественное число – максимальную ожидаемую оценку. Если выполнить  $K$  элементов невозможно, то оценка должна быть равна 0.

### Примеры

<i>input.txt</i>	<i>output.txt</i>
3 2 10 9 8 1 7 1 1 0.5 0.51 1 1 0.3 1 0.4 1	17.5
2 2 10 1 10 2 0.5 -1 0.3 0.5	19.70

## Задача 8. Фильтрация пакетов

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 5 секунд на тест  
Ограничение по памяти: 64 Мб

Расширение исследовательской программы экспедиции привело к увеличению объёма информации, передаваемой по локальной сети внутри базы. Из-за этого сигналы, идущие от передатчиков с поверхности планеты (в том числе и от тех, которые передают информацию от прогрессоров, находящихся на задании), часто запаздывают. Так что руководитель экспедиции поручил вам заняться оптимизацией алгоритмов, используемых для передачи информации по локальной сети.

Для начала Вы решили разработать алгоритм фильтрации пакетов, основным достоинством которого будет СКОРОСТЬ работы. Требуется написать программную реализацию такого фильтра. Пакет представляется в виде совокупности полей длины, кратной 8 битам. Для простоты предполагается, что правила пропускания пакетов имеют точечный характер: если все поля пакета совпадают с одной из масок пропускания, то пакет пропускается в сеть, иначе – блокируется.

### Входные данные

В первой строке входного файла записано одно целое число  $N$  — количество полей пакетов ( $1 \leq N \leq 6$ ). В следующей строке следуют  $N$  чисел, кратных 8, обозначающих размер каждого из полей пакета, размер каждого поля не превышает 32 битов. Далее на новой строке записано целое число  $K$  – количество правил фильтрации ( $1 \leq K \leq 10000$ ). На следующих  $K$  строках дано описание масок. Каждая маска описывается на отдельной строке, в виде последовательности символов ‘0’, ‘1’, ‘x’. Символ ‘x’ означает, что в пропускаемом пакете в указанном бите может стоять любое значение. Символы ‘0’ и ‘1’ означают, что в пропускаемом пакете в указанном бите должно стоять значение ‘0’ или ‘1’ соответственно. Символы ‘x’ обязаны либо заполнять все биты какого-то поля пакета, либо не встречаться в описании маски вовсе.

После описания масок на новой строке следует число  $M$  — количество пакетов ( $1 \leq M \leq 10000$ ), которые необходимо отфильтровать. В следующих  $M$  строках идет описание пакетов в виде последовательности символов ‘0’, ‘1’, задающих значения соответствующих бит в пакете. Количество символов ‘0’, ‘1’ в строке точно совпадает с размером пакета, равным сумме длин всех полей.

### Выходные данные

В выходном файле, для каждого пакета, на новой строке необходимо вывести ‘1’, если мы его пропускаем, и ‘0’, если его запрещаем.

### Пример

<i>input.txt</i>	<i>output.txt</i>
3	1
8 8 8	0
4	1
10001100xxxxxxxxxxxxxxxx	0
11101000xxxxxxxx11010010	1
11101000xxxxxxxx11111111	1
xxxxxxxx11111111xxxxxxxx	
6	
111111101111111111111111	
111111101111011111111111	
111010001111011111111111	
111010001000101011011111	
111010000000000011010010	
100011001000010111001010	

## Задача 9. Погоня

Имя входного файла: *input.txt*  
Имя выходного файла: *output.txt*  
Ограничение по времени: 2 секунды на тест  
Ограничение по памяти: 64 МБ

Предводитель восставших Арата Горбатый убежал по лесу от погони. Стрелы свистели справа и слева. Чтобы быть в безопасности от стрел преследователей, Арата должен рассчитать свою траекторию так, чтобы пройти на минимально возможном расстоянии от максимального числа деревьев. При этом бежать он должен по параболе – такова особенность леса.

Вам заданы координаты деревьев в том порядке, в котором мимо них бежит Арата. Необходимо провести параболическую кривую  $FL(x)$  на как можно более длинном начальном участке  $x_0 \dots x_m$ , так, чтобы сумма квадратов отклонений  $y$ -координат этой кривой от  $y$ -координат деревьев была не больше заданной погрешности  $E$ :

$$\sum_{i=0}^m (FL(x_i) - y_i)^2 \leq E$$

### Входные данные

Первая строка входного файла содержит числа  $N$  и  $E$ , записанные через пробел ( $0 < N \leq 50000$ ,  $0.1 \leq E \leq 1$ ).  $E$  задается с точностью до двух знаков после запятой. Далее в  $N$  строках задаются координаты деревьев  $(x_i, y_i)$ ,  $3 \leq i \leq N$ . Координаты деревьев отсортированы по  $x$ , т.е. гарантируется, что если  $i < k$ , то  $x_i < x_k$ . Все координаты не превосходят 10 по модулю, и заданы не более чем с 5 знаками после запятой.

### Выходные данные

В выходной файл нужно записать одно целое число  $m$  — максимально возможное количество деревьев, мимо которых Арата может пробежать по параболе.

### Пример

<i>input.txt</i>	<i>output.txt</i>
3 0.1 0.0 10.0 5.6 8.1 7.2 4.2	3



## Задача 10. Эксперимент продолжается

Имя входного файла: *input.txt*  
Имя выходного файла: *output.txt*  
Ограничение по времени: 2 секунды на тест  
Ограничение по памяти: 64 МБ

За какие-то два года дону Рэбе удалось известить под корень практически всех учёных и просто мало-мальски знающих людей. Это не могло не сказаться на состоянии армии. После того, как Святой Орден захватил власть, дон Рэба объявил чуде: бог в ответ на молитвы арканарцев ниспослал им новое оружие — "рогатку святого Мики".

Оружие это представляло собой обыкновенную баллисту, и было придумано отцом Кабани ещё до того, как дон Рэба появился при арканарском дворе. К баллисте прилагалась таблица синусов для вычисления параметров стрельбы. Но проблема была в том, что осталось мало тех, кто умел этой таблицей пользоваться.

Именно на это и решили сделать ставку руководители экспедиции, отправив в Арканар нового резидента взамен не справившегося с заданием Антона. Резидент должен начать военную карьеру с низов, и, благодаря своим умениям, подняться до высших военных чинов. И вот новобранец предстал перед военным представителем Святого Ордена.

— Говоришь, стрелять умеешь и в числах поднаторел? А ну-ка, во имя Господа, сложи все эти числа в табличке!

Задача была не из сложных, но и не из приятных. К счастью, новая система связи, введённая после провала Руматы Эсторского, позволяла получать ответ с базы. Так что вас попросили написать простейшую программу для того, чтобы сэкономить время нового резидента.

### Входные данные

Во входном файле в одной строке записано через пробел четыре числа  $K$ ,  $L$ ,  $x_0$  и  $h$ .  $K$  и  $L$  — целые,  $K$  — количество десятичных знаков после запятой в таблице ( $1 \leq K \leq 5$ ),  $L$  — количество чисел в этой таблице ( $1 \leq L \leq 10^9$ ).  $x_0$  — начальное значение аргументов в этой таблице, заданное не более чем с 9 знаками после десятичной точки ( $0 \leq x_0 \leq \pi/2$ ),  $h$  — шаг, с которым эти значения приводятся в таблице ( $10^{-9} \leq h$ ). При этом все аргументы  $x_i = x_0 + h \cdot i$  находятся в интервале между 0 и  $\pi/2$  ( $0 \leq i \leq L-1$ ).

Имейте в виду, что при указанных в задаче ограничениях **относительная** погрешность вычислений тригонометрических функций в типе `double` не превосходит  $10^{-14}$ , а реально ещё меньше.

### Выходные данные

В выходной файл нужно выдать одно число, равное  $\sum_{i=0}^{L-1} \sin(x_0 + h \cdot i)$ , причем значения синусов

округляются до  $K$  десятичных знаков по всем правилам, то есть до ближайшего  $K$ -разрядного числа. Если число равноудалено от двух  $K$ -разрядных чисел, то оно округляется до большего из них. Например, при округлении до двух знаков после запятой 0.446 округляется до 0.45, 0.4434 — до 0.44, 0.445 — до 0.45.

### Примеры

<i>input.txt</i>	<i>output.txt</i>
4 10 0.0 0.01	0.4497
1 10 0.0 0.01	0.4