

## Problem Tetris. Още едно $n$ -мерно шоколадче

Input file:           input.txt or standard input  
Output file:         output.txt or standard output  
Time limit:          2 seconds  
Memory limit:       512 megabytes

Маката на мъничкия Вася му купила  $n$ -мерно шоколадче, представляващо  $n$ -мерен куб, чиято дължина на страната е равна на 1. На шоколадчето са отбелязани линии за разделянето му на резени. По  $i$ -тото измерение то може да се раздели с хиперравнини на  $a_i$  равни части. По този начин, шоколадчето се разделя общо на  $a_1 \cdot a_2 \cdot a_3 \cdot \dots \cdot a_n$  резенчета, дължината на всяко, от които по  $i$ -тото измерение е равна на  $\frac{1}{a_i}$  и съответно обемът на всяко резенче е равен на  $\frac{1}{a_1 a_2 \dots a_n}$ .

Вася и приятелите му много искат да разрежат шоколадчето, така че да се получат поне  $k$  парченца, при това Вася иска да максимизира обемът на най-малкото от тях. Шоколадчето може да се реже само по местата на свързване на резенчетата, при това всеки разрез трябва да преминава през цялото шоколадче по някоя от хиперравнините, участващи в образуването на резенчетата. Само след като направи всички разрези, Вася може да раздели шоколадчето на парченца.

Формално, Вася иска да избере числа  $b_1, b_2, \dots, b_n$  ( $1 \leq b_i \leq a_i$ ) — броят на частите, на които да разреже шоколадчето през всяко измерение. Трябва да се изпълнява условието  $b_1 \cdot b_2 \cdot \dots \cdot b_n \geq k$ , за да се получат не по-малко от  $k$  парченца след всички разрязвания. Забележете, че при оптимално разрязване с такива параметри, минималното парченце ще съдържа  $\lfloor \frac{a_1}{b_1} \rfloor \cdot \dots \cdot \lfloor \frac{a_n}{b_n} \rfloor$  резена, а неговият обем ще бъде равен на  $\lfloor \frac{a_1}{b_1} \rfloor \cdot \dots \cdot \lfloor \frac{a_n}{b_n} \rfloor \cdot \frac{1}{a_1 a_2 \dots a_n}$ .

Вася иска да получи максималната възможна стойност на обема на минималното парченце, умножено по  $k$ , т.е. той иска да максимизира числото  $\lfloor \frac{a_1}{b_1} \rfloor \cdot \dots \cdot \lfloor \frac{a_n}{b_n} \rfloor \cdot \frac{1}{a_1 a_2 \dots a_n} \cdot k$ . Помогнете му за това.

### Input

От първия ред се въвеждат две цели числа  $n$  и  $k$  ( $1 \leq n \leq 100$ ,  $1 \leq k \leq 10^7$ ) — размерността на шоколадчето и на колко части трябва да се раздели.

На втория ред са дадени  $n$  цели числа  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^7$ ) — броят парченца, на които е разделено шоколадчето по всяко от измеренията.

### Output

Изведете едно число — максималния възможен обем на най-малкото от получените парченца, умножен по  $k$ , с абсолютна или относителна грешка не по-голяма от  $10^{-9}$ .

Ако при зададените ограничения е невъзможно шоколадчето да се разреже на поне  $k$  парченца, изведете 0.

## Examples

| input                 | output                 |
|-----------------------|------------------------|
| 1 2<br>5              | 0.8                    |
| 2 6<br>5 10           | 0.72                   |
| 2 7<br>4 4            | 0.875                  |
| 2 3<br>4 5            | 0.75                   |
| 4 444<br>57 179 239 2 | 0.97557326850704739751 |
| 2 5<br>2 2            | 0                      |

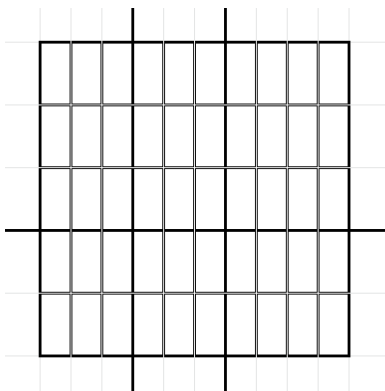
## Note

В първия пример едномерно шоколадче може да се раздели по следния начин:



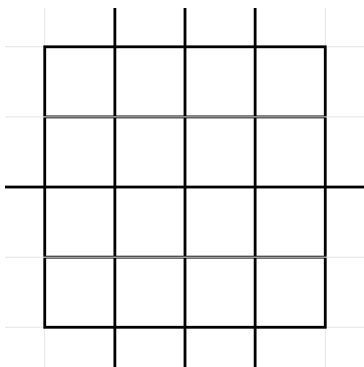
Тогава отговорът ще бъде  $\frac{2}{5} \cdot 2 = 0.8$

Във втория пример шоколадчето може да се разреже по следния начин:



Тогава отговорът ще бъде  $\frac{2}{5} \cdot \frac{3}{10} \cdot 6 = 0.72$

В третия пример шоколадчето може да се разреже по следния начин:



Тогава отговорът ще бъде  $\frac{2}{4} \cdot \frac{1}{4} \cdot 7 = 0.875$

## Scoring

Тестовые к этой задаче состоят из 8 групп. Точки за каждую группу даются только при преминаване на все тесты в группе и все тесты от некоторых из предыдущих групп. Обратите внимание, что преминаването на тесты от условия не требуется для некоторых групп. **Offline-проверка** означает, что результаты от тестирования на ваше решение на данную группу станут доступны после подключения к соревнованию.

| Группа | Точки | Доп. ограничения |                       |                 | Необх. группы | Комментарии              |
|--------|-------|------------------|-----------------------|-----------------|---------------|--------------------------|
|        |       | $n$              | $k$                   | $a_i$           |               |                          |
| 0      | 0     | –                | –                     | –               | –             | Тесты от условия.        |
| 1      | 10    | $n \leq 2$       | –                     | –               | –             |                          |
| 2      | 12    | –                | $k \leq 500$          | $a_i \leq 500$  | 0             |                          |
| 3      | 13    | –                | $k \leq 20\,000$      | $a_i \leq 2000$ | 0, 2          |                          |
| 4      | 12    | –                | $k \leq 40\,000$      | –               | 0, 2, 3       |                          |
| 5      | 10    | –                | $k \leq 200\,000$     | –               | 0, 2, 3, 4    |                          |
| 6      | 11    | –                | $k \leq 4 \cdot 10^6$ | $a_i \leq 2000$ | 0, 2, 3       |                          |
| 7      | 15    | –                | $k \leq 5 \cdot 10^6$ | –               | 0, 2 – 6      |                          |
| 8      | 17    | –                | –                     | –               | 0 – 7         | <b>Offline-проверка.</b> |

## Problem Super Mario. Цени на бензина

Input file:           input.txt or standard input  
Output file:          output.txt or standard output  
Time limit:           3.5 seconds  
Memory limit:        1024 megabytes

Берландия е огромна страна, в която има  $n$  града. Пътната мрежа на Берландия може да се представи в вид на кореново дърво, т.е. в страната има общо  $n - 1$  пътя, и от произволен град може да се достигне до произволен друг точно по един път, ако не се посещават никои градове по два пъти. За по-удобно представяне на страната, за всеки град  $i$  е фиксиран град  $p_i$  - първият град, до който трябва да се отиде от града  $i$ , по пътя за града 1. С други думи, градът  $p_i$  е предшественика на  $i$ , ако дървото се провеси за върха 1.

Във всеки град на Берландия работи по една бензиностанция. Бензиностанциите имат особено ценнообразуване, и за всяка бензиностанция е фиксиран диапазон на цените, на които там са готови да продават бензин. Бензиностанцията в града с номер  $i$  е готова да продава бензин на произволна цена от  $l_i$  до  $r_i$  включително.

Кралят на Берландия изключително много държи на увеличаване на броя на своите наследници, и в течение на  $m$  години, всяка година му се раждали по два сина. Децата на краля от ранно детство участват в държавните дела, и в края на всяка година те проверяват коректността на цените на бензина. От самото си раждане децата на краля, които се родени в година  $i$ , отговарят за проверката на цената на бензина на пътищата от града  $a_i$  до града  $b_i$  и от града  $c_i$  до града  $d_i$ .

Проверката протича по следния начин: две деца едновременно тръгват по пътищата от градовете  $a_i$  и  $c_i$ . Първият син на краля, роден в годината  $i$ , се движи по пътя от града  $a_i$  до града  $b_i$ , а втория — от града  $c_i$  до града  $d_i$ . Децата проверяват, дали цената на бензина в града  $a_i$  съвпада с цената на бензина в града  $c_i$ . След това те проверяват, дали цената на бензина във втория град на пътя от  $a_i$  до  $b_i$  съвпада с цената във втория град на пътя от  $c_i$  до  $d_i$ . След това те повтарят същото за двойката трети градове на техните пътища и т.н. Накрая, те проверяват, дали цената на бензина в града  $b_i$  съвпада с цената на бензина в града  $d_i$ . Гарантирано е, че дължината на пътя от града  $a_i$  до града  $b_i$  съвпада с дължината на пътя от града  $c_i$  до града  $d_i$ .

Бензиностанциите са длъжни да спазват строго законите и затова всички проверки на цените на бензина не би трябвало да показват нарушения. Помогнете на бензиностанциите на Берландия да разберат по колко различни начина могат да задават цените на бензина в продължение на  $m$  години. С други думи, за всяко  $i$  от 1 до  $m$  пресметнете, по колко начина могат да се зададат цените на бензина във всички бензиностанции, за да може след раждането на първите  $i$  двойки деца на краля, всички техни проверки да не откриват нарушения и на всяка бензиностанция цената да е в допустимия диапазон. Тъй като броят на тези начини може да бъде много голям, пресметнете отговора по модул  $10^9 + 7$ .

### Input

От първия ред се въвежда единствено цяло число  $n$  ( $1 \leq n \leq 200\,000$ ) — броя на градовете в Берландия.

На втория ред са дадени  $(n - 1)$  числа  $p_2, p_3, p_4, \dots, p_n$  ( $1 \leq p_i \leq n$ ), където  $p_i$  означава номера на следващия град по пътя от града  $i$  до града 1.

На всеки от следващите редове се въвеждат по две цели числа  $l_i$  и  $r_i$  ( $1 \leq l_i \leq r_i < 10^9 + 7$ ), задаващи допустимия диапазон на цените в бензиностанция номер  $i$ .

На следващия ред е дадено единствено цяло число  $m$  ( $1 \leq m \leq 200\,000$ ) — броя години, в продължение, на които на краля му се раждали по два сина.

На всеки от следващите  $m$  реда са дадени по четири цели числа  $a_i, b_i, c_i$  и  $d_i$  ( $1 \leq a_i, b_i, c_i, d_i \leq n$ ), задаващи двата пътя, на които ще проверяват цените на бензина децата на краля, родени през годината  $i$ . Гарантирано е, че дължината на пътя между градовете  $a_i$  и  $b_i$  е равна на дължината на

пътя между градовете  $c_i$  и  $d_i$ .

## Output

На  $m$  реда изведете по едно число. Числото в  $i$ -тия ред е равно на броя на начините, по които може да се зададат цените на бензина във всички градове, така че децата на краля, родени в годините до  $i$ -тата включително да не откриват нарушения при проверката. Числата да се извеждат по модул  $10^9 + 7$ .

## Examples

| input         | output |
|---------------|--------|
| 5             | 18     |
| 1 1 2 2       | 18     |
| 2 4           | 4      |
| 1 3           | 0      |
| 1 3           |        |
| 2 4           |        |
| 4 4           |        |
| 4             |        |
| 1 1 2 2       |        |
| 1 2 2 1       |        |
| 3 4 4 3       |        |
| 3 4 3 5       |        |
| 8             | 720    |
| 1 2 3 4 5 8 6 | 120    |
| 3 7           | 120    |
| 2 6           | 1      |
| 3 8           |        |
| 5 10          |        |
| 5 8           |        |
| 2 9           |        |
| 3 8           |        |
| 6 8           |        |
| 4             |        |
| 1 3 7 6       |        |
| 4 1 5 7       |        |
| 1 7 7 1       |        |
| 1 8 2 7       |        |

## Note

Да разгледаме първия пример.

След раждането на първите два сина цените в градовете 1 и 2 трябва да бъдат еднакви. Съществуват общо 2 начина да се изберат еднакви цени на бензина за градовете 1 и 2, за да може тази цена да се включи в допустимия диапазон на цените за тези два града. Това значи, че всички начини за задаване на цените на бензина са  $2 \cdot 3 \cdot 3 \cdot 1 = 18$ .

Втората двойка синове ще проверява цените на пътищата  $1 - 2$  и  $2 - 1$ . Това означава, че цените на бензина в градовете 1 и 2 трябва да съвпадат, което вече е изпълнено. Затова след раждането на втората двойка синове отговорът няма да се промени.

Третата двойка синове ще проверява цените на пътищата  $3 - 1 - 2 - 4$  и  $4 - 2 - 1 - 3$ . Тогава цената на бензина в града 3 трябва да бъде равна на цената на бензина в града 4, и цената в града 1 трябва да бъде равна на цената в града 2. Цените в градовете 1 и 2 вече са еднакви. За градовете 3 и 4 съществуват 2 начина да се изберат еднакви цени на бензина, за да може въпросната цена да е

допустима за ценовия диапазон и на двата града. Това означава, че всички различни начини да се зададат цени на бензина са:  $2 \cdot 2 \cdot 1 = 6$ .

Четвъртата двойка синове ще проверява цените на пътищата  $3 - 1 - 2 - 4$  и  $3 - 1 - 2 - 5$ . Това означава, че цените в градовете 4 и 5 трябва да бъдат равни, и тъй като цените в градовете 3 и 4 вече съвпадат, то в градовете 3, 4 и 5 трябва да има еднаква цена на бензина. Цената на бензина в града 3 трябва да бъде не повече от 3, а цената на бензина в града 5 трябва да бъде не по-малка от 4. Това означава, че след раждането на четвъртата двойка синове не съществува начин да се зададат цените на бензина така, че всички проверки да се изпълнят без грешка и цените да са в необходимите диапазони.

## Scoring

Тестовите към тази задача се състоят от 8 групи. Точките за всяка група се дават само при преминаване на всички тестове в групата и всички тестове от някои от предходните групи. Обърнете внимание, че преминаването на тестовите от условието не се изисква за някои групи. **Offline-проверка** означава, че резултатите от тестването на вашето решение на дадената група ще станат достъпни след приключване на състезанието.

| Група | Точки | Доп. ограничения  |                   | Необх. групи | Коментари   |
|-------|-------|-------------------|-------------------|--------------|---|
|       |       | $n$               | $m$               |              |   |
| 0     | 0     | –                 | –                 | –            | Тестове от условието.   |
| 1     | 12    | $n \leq 300$      | $m \leq 300$      | 0            |   |
| 2     | 10    | $n \leq 3000$     | $m \leq 3000$     | –            | $p_i = i - 1$   |
| 3     | 9     | $n \leq 3000$     | $m \leq 3000$     | 0, 1, 2      |   |
| 4     | 16    | –                 | –                 | 0 – 3        | Общата дължина на всички пътища, на които ще се прави проверка на цените, не надминава $10^8$ |
| 5     | 10    | $n \leq 100\,000$ | $m \leq 100\,000$ | 2            | $p_i = i - 1$   |
| 6     | 12    | –                 | –                 | 2, 5         | $p_i = i - 1$   |
| 7     | 13    | $n \leq 100\,000$ | $m \leq 100\,000$ | 0 – 3, 5     |   |
| 8     | 18    | –                 | –                 | 0 – 7        | <b>Offline-проверка.</b>  |

## Problem Pac-Man. Kings problem

Input file: `input.txt` or standard input  
Output file: `output.txt` or standard output  
Time limit: 3 seconds  
Memory limit: 512 megabytes

Auto translated. We will try to get Bulagrian statements as soon as possible.

Most recently, a new road network was built in Berland. There are one-way roads between some pairs of cities,  $i$ -th of which leads from the city  $u_i$  to the city  $v_i$ , and its length is  $w_i$ . The two main cities of Berland have numbers  $a$  and  $b$ .

The King of Berland loves his country very much. In particular, he loves to count all sorts of characteristics in it. It calls *beauty* of some path, the bitwise exclusive OR the lengths of all roads on that path. And he calls the beauty of his country the bitwise exclusive OR the beauty of all paths from the city  $a$  to the city  $b$ . Note that there can be infinitely many such paths, and they can pass through the same city several times.

The king wants to know what the beauty of his country is equal to, and therefore he turned to you for help and asks you to calculate this value or say that it is impossible to calculate the beauty of the country.

A bitwise exclusive OR of set of numbers is called a bitwise exclusive OR of all nonzero numbers in this set. If there are infinitely many nonzero numbers in the set, then it is impossible to calculate the bitwise exclusive OR.

### Input

Each test consists of several sets of input data. The first line contains a single integer  $t$  ( $1 \leq t \leq 40\,000$ ) — number of input data sets.

The first line of each input data set contains two integers  $n$  and  $m$  ( $1 \leq n, m \leq 200\,000$ ) — the number of cities and the number of roads in Berland.

The following  $m$  lines each contain three integers  $u_i, v_i$  and  $w_i$  ( $1 \leq u_i, v_i \leq n, 0 \leq w_i \leq 2^{30} - 1$ ) — start and end cities of  $i$ -th road and its length.

The last line of each input data set contains two integers  $a$  and  $b$  ( $1 \leq a, b \leq n$ ) — the numbers of the beginning and end of the paths that interest the king.

Let's denote for  $\sum n$  the sum of  $n$ , and for  $\sum m$  the sum of  $m$  for all sets of input data in one test. It is guaranteed that  $\sum n \leq 200\,000$  and  $\sum m \leq 200\,000$ .

### Output

For each set of input data, output one integer — the beauty of Berland. If there is no answer, then print  $-1$ .

## Example

| input | output |
|-------|--------|
| 5     | 0      |
| 1 1   | 7      |
| 1 1 0 | -1     |
| 1 1   | 0      |
| 3 5   | -1     |
| 1 2 0 |        |
| 1 2 1 |        |
| 1 2 3 |        |
| 2 3 5 |        |
| 2 3 2 |        |
| 1 3   |        |
| 2 2   |        |
| 1 2 1 |        |
| 2 1 2 |        |
| 1 2   |        |
| 3 3   |        |
| 1 2 7 |        |
| 2 3 0 |        |
| 3 1 7 |        |
| 2 3   |        |
| 4 5   |        |
| 1 1 0 |        |
| 1 2 3 |        |
| 2 2 0 |        |
| 2 3 1 |        |
| 3 4 1 |        |
| 1 4   |        |

## Scoring

The tests for this task consist of 6 groups. Points for each group are given only when passing all the tests of the group and all the tests of some of the previous groups. **Offline-check** means that the results of testing your solution on this group will be available only after the end of the competition.

| Group | Points | Constraints        |                    |                       | Required | Comment   |
|-------|--------|--------------------|--------------------|-----------------------|----------|---|
|       |        | $\sum n$           | $\sum m$           | $w_i$                 |          |   |
| 0     | 0      | -                  | -                  | -                     | -        | Samples   |
| 1     | 16     | -                  | -                  | -                     | -        | $n = m$<br>$u_i = i, v_i = i + 1$ for $i < n$<br>$u_n = n, v_n = 1$ |
| 2     | 17     | -                  | -                  | $w_i \leq 1$          | -        | $u_i < v_i$   |
| 3     | 15     | -                  | -                  | -                     | 2        | $u_i < v_i$   |
| 4     | 19     | $\sum n \leq 1000$ | $\sum m \leq 1000$ | $w_i \leq 2^{10} - 1$ | 0        |   |
| 5     | 14     | -                  | -                  | $w_i \leq 1$          | 2        |   |
| 6     | 19     | -                  | -                  | -                     | 0 - 5    | <b>Offline-check.</b>   |



## Problem DOOM. Музикален фестивал

Input file: `input.txt` or standard input  
Output file: `output.txt` or standard output  
Time limit: 1 second  
Memory limit: 512 megabytes

Малкият Витя много обича да слуша музика. Той постоянно следи новите неща около своята любима група, ето затова той знае, че в петък трябва да излязат  $n$  албума,  $i$ -я от които съдържа  $k_i$  песни. Разбира се, Витя, като най-големият фен, вече е слушал всички песни, които трябва да излязат в близко време и знае, че в  $i$ -тия албум якостта на  $j$ -тата песен е равна на  $a_{i,j}$ .

Витя има приятелка Маша, която той много иска да покани на фестивал, в който ще участва неговата любима група. За да се съгласи да отиде, тя първо трябва да оцени новите албуми. Витя знае, че ако Маша прослуша песен, която е била по-яка от всички чути преди това, тя ще получи 1 единица впечатление.

Помогнете на Витя да намери такава последователност от албуми, така че впечатлението на Маша да бъде възможно най-голямо, за да отиде с него на фестивала.

### Input

На първия ред е дадено едно цяло число  $n$  ( $1 \leq n \leq 200\,000$ ) — броя на албумите.

Следва описание на албумите. Всяко описание се състои от два реда:

На първия ред е дадено едно цяло число  $k_i$  ( $1 \leq k_i \leq 200\,000$ ) — броя песни в  $i$ -тия альбум.

На следващия ред са дадени  $k_i$  цели числа  $a_{i,1}, a_{i,2}, a_{i,3}, \dots, a_{i,k_i}$  ( $1 \leq a_{i,j} \leq 200\,000$ ) — якостта на песните в  $i$ -тия албум.

Означаваме със  $\sum k_i$  сумата на всички  $k_i$ . Гарантирано е, че  $\sum k_i \leq 200\,000$ .

### Output

Изведете едно цяло число — максималното впечатление, което може да получи Маша.

### Examples

| input   | output |
|---|--------|
| 4<br>5<br>4 9 4 6 8<br>1<br>7<br>2<br>8 6<br>1<br>1 | 4      |
| 4<br>2<br>3 4<br>2<br>1 8<br>2<br>2 8<br>2<br>7 9   | 4      |

## Note

В първия пример оптимално е албумите да се прослушат в следният ред: 4-ти, 2-ри, 3-ти и 1-ви. По този начин Маша ще прослуша песните в следната последователност: **1; 7; 8**, 6; 4, **9**, 4, 6, 8 и ще получи 4 единици впечатление.

Във втория пример трябва отначало да се прослушат 1-ви, а след това 4-ти и в произволен ред 2-ри и 3-ти. По този начин Маша ще получи максимално впечатление за всяка от песните от 1-ви и 4-ти албум и нищо от 2-ри и 3-ти.

## Scoring

Тестовите към тази задача са в 7 групи. Точките за всяка група се дават само при преминаване на всички тестове за тази група, както и всички тестове от някои предходни групи. Обърнете внимание, че преминаването на примерните тестове не е задължително за никоя група.

| Група | Точки | Доп. ограничения |                      |                        | Необх. групи | Коментари           |
|-------|-------|------------------|----------------------|------------------------|--------------|---------------------|
|       |       | $n$              | $k_i$                | $a_{i,j}$              |              |                     |
| 0     | 0     | –                | –                    | –                      | –            | Примерните тестове. |
| 1     | 14    | $n \leq 7$       | $\sum k_i \leq 1000$ | –                      | 0            |                     |
| 2     | 9     | –                | –                    | $a_{i,j} \leq 2$       | –            |                     |
| 3     | 12    | –                | –                    | $a_{i,j} \leq 10$      | 0, 2         |                     |
| 4     | 15    | –                | $k_i \leq 2$         | –                      | –            |                     |
| 5     | 13    | $n \leq 1000$    | –                    | $a_{i,j} \leq 1000$    | 0            |                     |
| 6     | 13    | $n \leq 30\,000$ | –                    | $a_{i,j} \leq 30\,000$ | 0, 5         |                     |
| 7     | 24    | –                | –                    | –                      | 0 – 6        |                     |