PROBLEM A

ABNORMAL 89'S

A palindrome is a word that can be read the same way in either direction. More formally if a string is d (d > 0) characters length and the i^{th} character is a_i, the string is palindrome if and only if a_i equals $a_{(d-i+1)}$ for $1 \le i \le d$. For example "abcba" is palindrome while "aaab" is not.

It is known that everyone who gets to know palindromes, begin an emotional relationship with these beautiful strings. The harmony between the letters makes them artistic. But the 89's (those who entered AUT at 1389) claim they love another kind of strings. It is called *alindrome*. Actually an alindrome is the result of concatenation of two palindromes. For example "abacc"="aba"+"cc" is alindrome.

Now you should write a program to distinguish alindromes, palindromes and other kind of strings.

Input (Standard Input)

The first line contains *T* ($T \le 100$), the number of tests. Each test that comes in a separate line contains a string to be checked. Input strings contain only lower case letters ('a' to 'z') and their length will be at most 100.

Output (Standard Output)

For each test output a single word in a single line. If the input string can be made by concatenating two palindromes, output "`alindrome`". Otherwise if it's a palindrome output "`palindrome`". In any other case print "`simple`". (Quotes for clarity)

Sample Input and Output

Sample Input	Sample Output
4 aaa aabaa aabaaa abc	alindrome palindrome alindrome simple

PROBLEM B

BENEFIT

Recently *Yaghoub* is playing a new trick to sell some more. When somebody gives him A Tomans, he who never has appropriate changes, asks for B Tomans such that lowest common multiple of A and B equals to C and he will pay back a round bill. Or otherwise take some snack instead of the remaining of his money. He believes that finding such a number is hard enough that dissuades students from paying that.

You should write a program that help poor students giving the appropriate amount of money to Yaghoub. Of course if there are several answers you go for students' benefit which is the lowest of them.

Input (Standard Input)

The first line begin with an integer *T* (T ≤ 100), the number of tests. Each test that comes in a separate line contains two integers *A* and *C* (1 ≤ A, C ≤ 70000).

Output (Standard Output)

Print the lowest integer B such that LCM(A,B)=C in a single line. If no such integer exists, print "NO SOLUTION" instead. (Quotes for clarity)

Sample Input and Output

Sample Input	Sample Output
3 2 6 32 1760 7 16	3 55 NO SOLUTION

PROBLEM C

CALCULUS SIMPLIFIED

We define an expression as below:

```
<expression> := <number> | <expression>+<expression> |
      <expression>-<expression> | (<expression>)
```

where number is defined to be an integer.
In this problem you are given an expression with all its numbers replaced with character 'x'. Then you are given a set of numbers that were actually in the expression. We know that numbers were placed in the expression in such a way that the expression evaluates to maximum possible value among all other placements. Write a program that calculates this maximum value.

Input (Standard Input)

In the first line there is an integer *T* ($T \le 100$), the number of tests. Each test begins with the expression itself. Next line is an integer *N*, the number of numbers in the expression. In the final line of each test there are N integers a_i ($|a_i| \le 3000$). Each of these numbers should be used in the expression exactly once. It is guaranteed that the expression can be parsed by the definition in the problem statement and its length will not exceed 5000. There are no whitespaces in the expression and all numbers are replaced with a single 'x'. The number of 'x's in the expression is N.

Output (Standard Output)

For each test output the maximum possible value of the expression in a single line.

Sample Input and Output

Sample Input	Sample Output
3 x 1 2 x-x 2 -1 1 (x)+(x)-(x) 3 1 1 1	2 2 1

PROBLEM D

DEEP IN JUNGLES

The *Army of Computing Men* (ACM) is one the strongest armies in the world. They are well-equipped and have trained and armed programmers. They fight to spread creativity, teamwork and innovation among people.

The last victory of the ACM against knowledge monopolies brought them wealthy spoils. The supply arms have located position of N crates along a straight line in the jungle. Some trucks are to carry the crates to the army base. Since the jungle is so dense, it is costly for a truck to cut through and bring the crates back. Therefore commanders have decided to group the crates together in points and use the minimum number of trucks. For each such gathering point we should use a truck. Each truck has a maximum capacity of K crates and has required fuel to bring only one shipment.

The army men want to obey their commanders with minimum effort. Each crate has a weight. The effort needed to handle a crate along the line can be formulized by the weight of the crate multiplied by the distance it is carried. It is important not to change the order of crates while there are carried.

You are appointed to manage the shipping procedures. Higher ranked commanders asked you for a report on minimum needed resources. You should first consider minimizing the number of trucks and then the army effort. Write your report as soon as possible.

Input (Standard Input)

In the first line there will be *T* ($T \le 50$), number of tests. Each test begins with an integer *N* and *K* ($1 \le N \le 10^5$, $1 \le K \le 100$) . It is followed by N pairs of integers p_i and w_i ($1 \le p_i, w_i \le 10^6$), position and weight of i^{th} crate. Crates appear in the input in increasing order of their position. No two crates occupy the same position in the beginning.

Output (Standard Output)

For each test print the minimum number of trucks and the minimum effort needed to collect all crates in a single line separated by a single space.

Sample Input and Output

Sample Input	Sample Output
2	3 10

5 2 1 1 4 2 5 10 6 10 10 2 5 5 1 1 4 2 5 10 6 10 10 2	1 26

Hint

In the first sample test, we need at least 3 trucks to carry the crates. the optimal solution is to carry crate number (1), (2, 3), (4, 5) in separate trucks with costs 0, 2 (for carrying crate 2 to the position of crate 3) and 8 (for carrying crate 5 to the position of crate 4) respectively.

PROBLEM E

ENimEN

In deterministic games no chance is involved, meaning that the final result can be predicted from the initial arrangement assuming players play optimal. These games are so boring.

piloop and *poopi* are professional gamers. They play games only to study their algorithmic properties. Their field of expertise is boring games. One of the boring games they often play is Nim. Nim is a two-player game which is played using distinct heaps, each containing a number of objects (e.g. stones). Players take turns removing non-zero number of objects from a heap of their choice. The player who removes the last object will win.

They wonder if they can change the game to make it more fascinating. Would not that be more interesting if make the rules stricter? For example what if each player is obliged to take objects from the last non-empty heap as his opponent took objects from. And if there is no such heap, he can choose one heap freely and take objects from it. *ENimEN* is their new invented game based on this rule.

If you are interested in ENimEN, write a program to determine the winner given the initial arrangement assuming both players, play optimal. We believe it has also some benefits for you!

Input (Standard Input)

The first line contains *T* ($T \le 100$), the number of test cases. Each test begins with an integer *N* ($N \le 20000$) in the first line, the number of heaps followed by N integers a_i ($1 \le a_i \le 10^9$), are the number of objects in i^{th} heap.

Output (Standard Output)

If in the optimal strategy the first player is the winner print "`poopi`" (as he always plays first), otherwise print "`piloop`". (Quotes for clarity)

Sample Input and Output

Sample Input	Sample Output
2 2 1 1 4 1 2 1 1	Piloop poopi

PROBLEM F

FABULOUS DAGy

Little poopi had something strange that was called *DAGy*. poopi liked DAGy so much, but when he showed DAGy to other children they scared and ran away! DAGy is not a pet or a toy; it is a special kind of graph! DAGy is made up of a directed acyclic graph plus one additional directed edge. With this additional edge a cycle forms that goes through every vertex in the graph.

Once when poopi was playing with DAGy, it fell out of his hands and became totally deformed. He cried and cried. He denied new graphs because he wanted his own DAGy.

It is said that computer programmers are supermen, because they can solve problems that nobody else is able to even approach. You, the computer programmer! Help little poopi and dispose his DAGy again!

Input (Standard Input)

In the first line there is an integer *T* ($T \le 40$), the number of tests. You are given *N* and *M* ($1 \le N \le 400$) in the first line of each test, which are the number of vertices and the number of edges respectively. Next M pairs of integers u, v ($0 \le u, v < N$) meaning that there is an edge from vertex u to vertex v. There is at most one edge between each pair of vertices. It is guaranteed that each input graph is a directed acyclic graph with one additional edge between two distinct vertices of graph.

Output (Standard Output)

DAGy can be put back in order if you find the maximal cycle that goes through every vertex. If you found such a cycle print "`Yeah, I'm superman`" in a single line. Otherwise print "`Your DAGy was initially defected!`" (Quotes for clarity) You are superman trying to help little poopi anyway!

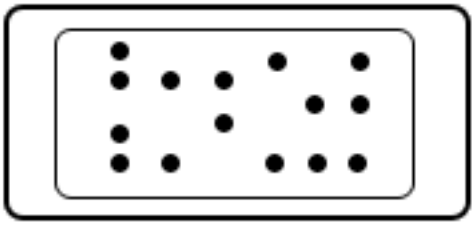
Sample Input and Output

Sample Input	Sample Output
2 3 3 0 1 1 2 2 0 4 5 0 1 1 2 2 0 0 3 3 2	Yeah, I'm superman Your DAGy was initially defected!

PROBLEM G

GENIUS MJ

After proving that many of current port types like Serial, USB and … are useless, *MJ* designed a new complicated port. His new port turned out to be extremely efficient, but it has one big problem. As the port is produced in diverse shapes for different purposes, people usually try to plug a male cable into a differing female socket, causing the pins to damage. So MJ proposed computer experts to solve this problem.



You should write a program to check whether the two parts of a port match. The pins (and holes) of the plug (socket) are given to you as a set of N distinct points in 2D plane. You can translate the points in a set altogether. You can also rotate them around the origin in multipliers of 90 degrees. (i.e. 90, 180 or 270 degrees) Two parts match each other if a one to one correspondence can be made between the points of the two sets using translation and rotation.

Input (Standard Input)

In the first line there is an integer *T* ($T \le 20$), the number of pairs of ports to check. Each test begins with an integer *N* ($1 \le N \le 10^5$), the number of pins (and holes). 2 * N lines follow. First N lines are two integers x_i and y_i ($|x_i|, |y_i| \le 1000$), coordinates of i^{th} pin of plug and next N lines are coordinates of socket in the same format as the plug. Points in each set are distinct.

Output (Standard Output)

For each set output a single word "MATCHED" if the two parts of the port match each other and "NOT MATCHED" if they do not. (Quotes for clarity)

Sample Input and Output

Sample Input	Sample Output
2 3 0 0 1 0 0 1 -2 0	MATCHED NOT MATCHED

-1 0 -1 -1 2 0 1 1 0 0 -1 0 0	

PROBLEM H

HONORARY TICKETS

Chief of AUT ICPC public relations offers an honorary visit to AUT ICPC site to lucky winners. It includes acknowledging former members *Agha Hadi*, *smile2ka10* and *Agha Reza*; meeting with current members! and an extra workshop to measure the performance of the site's computers. As it is a one time opportunity and everything is going to be renewed soon, many students registered.

Selection mechanism takes registration priority, student intellectuality and their chance into account all at once. There are M bags. Each bag contains a number of similar envelopes some of which contain a lucky ticket. Students know these amounts. They come one by one in order of registration and take one envelope from one of the bags. If it contains a lucky ticket, the student takes it. In either situation the empty envelope is put inside the bag again. The winners are announced after all students tried their chance, so nobody knows the result of the students before him.

nimA is K^{th} student to try his chance. But he is worried that no lucky tickets are left. So he asked you to compute his chance of winning a lucky ticket assuming all students preceding him are clever enough to take the best action.

Input (Standard Input)

First line begins with an integer *T* ($T \le 50$), the number of tests. In the first line of each test there are two integers *N* and *K* ($1 \le N, K \le 10^5$), the number of bags and position of nimA in the queue. Next N lines contain integers t_i and l_i, the total number of envelopes and number of envelopes which contain lucky tickets in the i^{th} bag.

Output (Standard Output)

For each test print the probability of nimA winning a lucky ticket in the form of a fraction *p/q* such that greatest common divisor of p and q equals to 1. It is guaranteed that such p and q's for nimA and all students preceding him will fit in 32-bit signed integer.

Sample Input and Output

Sample Input	Sample Output
3 2 1 1 1 2 1 2 2 2 2 6 4 1 3 6 4	1/1 2/3 25/54

PROBLEM I

INGLORIOUS GANGS

In the *Vice City* gangs are disturbing people. Gangs usually drive in streets of Vice City in groups of cars. The cops cannot resist because the number of gangs is much greater. Therefore each gang group can be chased by only one cop. But at some intersections in the city, each of the gang cars goes in one direction. The lonely cop will be confused which one to follow, and the gang group escapes.

The police officers know the fact that gang groups will only separate if they can gather together again somewhere in the city. They devised a plan and asked the mayor to block some of the roads to prevent the gang groups from separating. Now cops can chase gang groups till they catch them.

You are hired to check whether or not the city is well prepared to trap the sinister gangs. Map of the city is in the form of a directed graph where nodes indicate intersections and directed edges are roads of the city.

Input (Standard Input)

In the first line there is *T* ($T \le 100$), the number of tests. Each test begins with integers *N* and *M* ($1 \le N \le 10^4$, $M \le 10^5$), the number of intersections and roads in the Vice City. Next M lines are non-blocked roads of Vice City, each of which is a pair of integers u, v ($1 \le u, v \le N$, $u \ne v$) meaning that there is a road from intersection u to intersection v.

We know that the central police office located at intersection number 1 have routes to every other intersections in the city.

Output (Standard Output)

Print a single word "`Trapped`" if we can trap the gangs and "`Not Trapped`" otherwise. (Quotes for clarity)

Sample Input and Output

Sample Input	Sample Output
2 3 3 1 2 2 3 1 3 4 4 1 2 2 3 3 4 4 2	Not Trapped Trapped

PROBLEM J

JINGLING DOMINOES

Domino tiles are pieces separated in two parts, each having a number of dots on its surface. Yet numerous games are devised with these simple tiles. One the most amusing of these games is domino spiral. Domino spiral is a game in which tiles are put beside each other near enough that toppling the first domino topples the second, and so on.

Continuing his adventures, this time the great *Amoo Ardal* has constructed the longest domino spiral made ever. In fact he is going to improve his last record in Guinness World Records. The spiral is made up of millions of tiles and will take minutes to complete. He has arranged a magnificent ceremony. Many guests are invited from all around the world and press will record the event. One of the most famous and popular music bands will play the *Jingling Dominoes* piece during the event. To coordinate the music with dominos, the music band asked Amoo Ardal for the accurate ending time of the spiral. (i.e. the time for the last tile to fall)
Amoo is confused. As it took month to construct the spiral, he does not want to ruin his work for that purpose. So he asked you as a proficient programmer to do the job for him.
Your are given the domino structure in the form of a weighted graph where edges denote a series of continuous dominoes and the length of the edges indicate the number of dominoes on the edge. Each domino will fall in one second. Also nodes of the graph show the places where domino lines fork. There is no domino in the place of a node. When a tile falls down, all the neighboring not-fallen dominoes will fall too. Do not make any other assumptions.

Input (Standard Input)

In the first line of input there will be an integer *T* ($T \le 20$). T test sets follow. Each begins with three integers *N*, *M* and *S* ($2 \le N \le 1000$, $M \le 400000$, $0 \le S < N$), number of nodes, edges and the starting node respectively. Each of next M lines contain three integer u_i, v_i, c_i ($0 \le u_i, v_i < N$, $0 < c_i < 10^6$) which are the endpoints of i^{th} continuous domino line and the number of dominoes on it. It is guaranteed that each node has at least one edge and the starting node has exactly one edge. There are no loops or multiple edges in the graph.

Output (Standard Output)

If the domino spiral does not complete at all print "`Sorry Amoo, your domino does not complete!`" in a single line (Quotes for clarity). Otherwise print the time that the latest tile will fall rounded to two decimal digits after the decimal point.

Sample Input and Output

Sample Input	Sample Output
3 4 3 0 0 1 1 1 2 2 2 3 3 4 4 1 1 0 1 0 2 2 2 3 2 0 3 3 4 2 2 0 1 1 3 2 2	6.00 4.50 Sorry Amoo, your domino does not complete!

PROBLEM K

KILLER PROBLEM

You are given an array of *N* integers and *Q* queries. Each query is a closed interval [l, r]. You should find the minimum absolute difference between all pairs in that interval.

Input (Standard Input)

First line contains an integer *T* (T ≤ 10). T sets follow. Each set begins with an integer N (N ≤ 10^5). In the next line there are N integers a_i (1 ≤ a_i ≤ 10^4), the number in the i^{th} cell of the array. Next line will contain Q (Q ≤ 10^4). Q lines follow, each containing two integers l_i, r_i (1 ≤ l_i, r_i ≤ N, l_i < r_i) describing the beginning and ending of of i^{th} range. Total number of queries will be less than 15000.

Output (Standard Output)

For the i^{th} query of each test output the minimum $|a_j - a_k|$ for l_i ≤ j, k≤ r_i a single line.

Sample Input and Output

Sample Input	Sample Output
1 10 1 2 4 7 11 10 8 5 1 10000 4 1 10 1 2 3 5 8 10	0 1 3 4