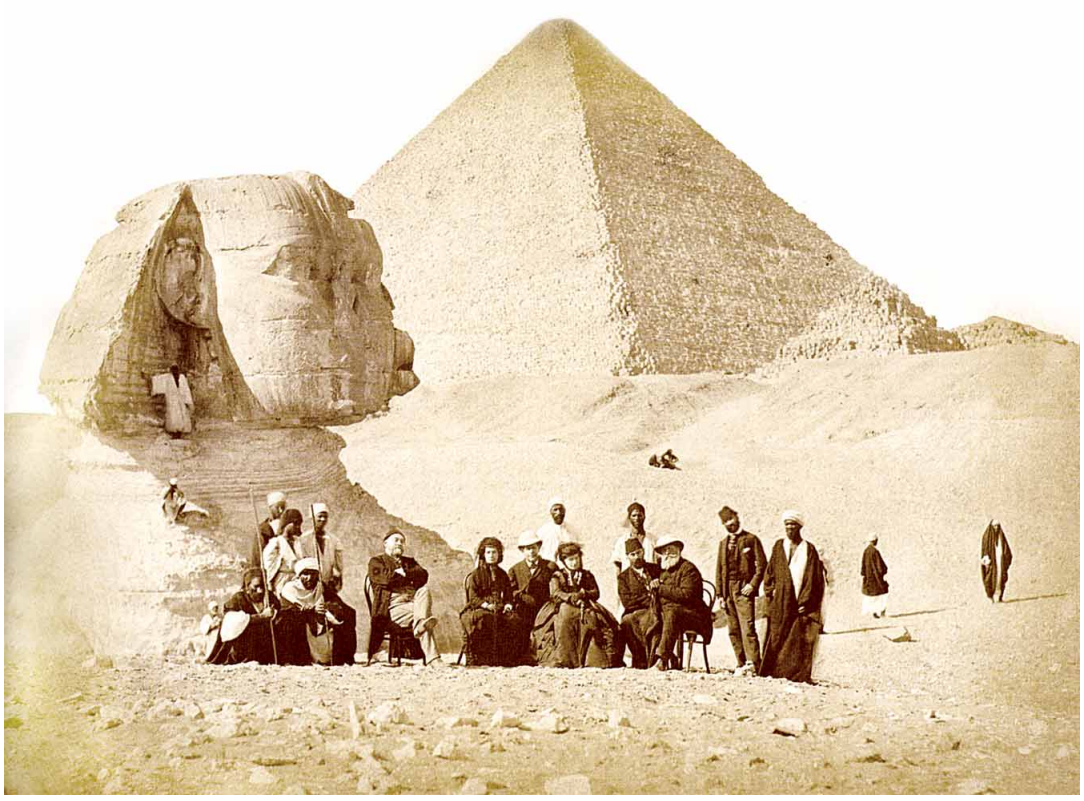
Problem A. Ancient Myth

- Input File: `stdin`
- Output File: `stdout`
- Time Limit: `1000 MS`
- Memory Limit: `4 MB`



Archaeologists recently unearthed a broken clay pot at the Luxor Temple on the east bank of the Nile. In fact, they considered it to be the earliest computer in human history. In the mythological system of ancient Egypt, there was a Sphinx who stopped passers-by and asked them with the riddle, anyone guessed it wrong would be eaten. Archaeologists believed that the prophets on the ancient Egypt had built this computer to solve the riddle given by Sphinx.

Rumors said that Sphinx once asked the great prophets a complicated question and the prophets used an artifact to answer that easily. The problem could be translated into modern language:

Given a collection of length n $(n \leq 3000000)$**, find out** k $(k \leq 500)$ **non-negative integers** a $(0 \leq a_i \leq 10^{10})$ **that appear odd times.**

Now, archaeologists believe that it is impossible to complete this computing task with an ancient computer that relies on only four megabyte of memory. But, can you give a second opinion?

Input

The first line contains two integers n, k. Each of the next n lines contains a non-negative integer, representing the elements in the set.

Output

k lines, each of k lines contains a non-negative integer that appear odd times, from small to large.

Sample

Sample Input 1	Sample Output 1
3 1 2 2 2	2

Note

It is guaranteed that all testcases have exactly k non-negative integers that appear odd times.
It is also guaranteed that those k non-negative integers are obeying uniform distribution in $[0, 10^{10}]$.

Problem B. Sdl and interview questions

- Input File: `stdin`
- Output File: `stdout`
- Time Limit: `1000 MS`
- Memory Limit: `256 MB`

There are new players entering the pit! SDL is responsible for interviewing new players this time. I remember that the last time SDL made a cute sister give up because of the problem of network flow directly during the interview. So this time, SDL only asked a simple question:

There are n points in the coordinate system, you need to divide these points into several groups (**each group is independent**), do the following operations for **each group**:

1. For each point (x_i, y_i) in the group, draw two lines $x = x_i$ and $y = y_i$.
2. When all the points have completed this operation, extract the intersection of all the lines(**If there are multiple lines that coincide, then only one is processed**).
3. The core point we define for this group is the one of the intersections that is furthest from the origin $(0, 0)$ of the coordinate axis.
4. Put the core points of this group after you have been grouped into a collection called "SdlSet".
5. Remove all lines.

After all the groups have completed the above operations, we have several core points in the SdlSet (the number of core points is derived from the number of groups you have divided). Rest assured, SDL does not let you find the core points of each group. He asked you to make a reasonable division plan to minimize:

$$\sum_{(x,y)\in \text{SdlSet}} xy$$

Your task is to give this minimum so you can pass the Sdl interview.

Input

The first line of the input contains an integer$n(1 \le n \le 2 \cdot 10^5)$, denoting the number of points.

In the second line, there are n pairs (x_1, y_1),$(x_2, y_2) \cdots$,(x_n, y_n) $(1 \le x_i, y_i \le 10^9)$,denoting coordinates of each point.

Output

Print a single line containing an integer, denoting the minimum value that sdl want.

Sample

Sample Input 1	Sample Output 1
4 1 10 3 3 4 2 10 1	32

Note

If the group is assigned as follows:

- Group 1: (1,10)
- Group 2: (10,1)
- Group 3: (3,3), (4,2)
- SdlSet: (1,10), (10,1), (4,3)

So the answer is $1 \times 10 + 10 \times 1 + 4 \times 3 = 32$.

Problem C. Sld and network-flows

- Input File: `stdin`
- Output File: `stdout`
- Time Limit: `1000 MS`
- Memory Limit: `256 MB`

Welcome to join the sdl national fan support club! (Link: https://acm.njupt.edu.cn/group/sdlnb)

Sdl is very enthusiastic about the fans of his support club. He is always willing to teach the members of the support club a high-level network flow algorithm, so that they can happily ak network flow topics. Today, a new batch of fans came in the club. After they queued up to receive the sdl signature photo, sdl found that these new fans could not be as happy as the old fans to ak network flow topics, so he let everyone line up in a row, and began to teach skills.

The number of new fans is n, each person has a network flow ability a_i, the ability is no limit, but through his own teaching, sdl wants the fans to have the same final ability value.

However, sdl has its own perfect habit. He only chooses two adjacent personal skills with the same ability value, and each time he can only increase the ability value of these two individuals by one. In other words, each time sdl can choose number $i(1 \leq i < n, a_i = a_{i+1})$, then $a_i = a_i + 1, a_{i+1} = a_{i+1} + 1$.

Sdl wants to ask you if he can reach his own wishes.

If you solve his problem, he will not teach you network flow skills, because you are not a member of the support club! However, as a reward, he will ask you to eat McDonald's.

Note that in the eyes of sdl, when $n > 1$, a_1 and a_n are not adjacent.

Input

The first line of the input contains an integer$n(1 \leq n \leq 2 \cdot 10^6)$, denoting the number of new fans.

In the second line, there are n intergers $a_1, a_2, \cdots, a_n(1 \leq a_i \leq 10^9)$,denoting the abilty value of each new fans.

Output

Print "`Yes`" if Sdl can reach his own wishes. Print "`No`" otherwise.

Sample

Sample Input 1	Sample Output 1
5 7 6 6 7 8	Yes

Note

First, Sdl selects (2, 3) and then selects (1, 2) (3, 4).

Problem D. Suggest to transfer to Africa (version 2)

- Input File: `stdin`
- Output File: `stdout`
- Time Limit: `2000 MS`
- Memory Limit: `256 MB`

Where is version 1? Go to https://acm.njupt.edu.cn/problem/NOJ2382 to view the details.

One day SDL secretly ate popcorn chicken in the computing center and was discovered by the boss. The boss was very angry and wanted to transfer him to Africa, so he came up with a math problem. If SDL can't answer this question, he will be sent to Africa.

The description of the problem is as follows.

We define the function $f(x,u)$ as the sum of x's factors when the sum of x's factors is not less than u, otherwise $f(x)$ is 0.

In short:

$$f(x,u)=\sigma(x)[\sigma(x)\geq u]=\begin{cases}\sigma(x), & \sigma(x)\geq u\\ 0, & \sigma(x)<u\end{cases}$$

where $\sigma(x)$ represents the sum of factors of x.

The boss has T questions. The format of each question is a triple (n,u), where u is the parameters mentioned above. For each question, SDL must answer the value of the following formula:

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}f(\gcd(i,j,k),u)\mod(10^9+7)$$

Can you solve this problem for SDL?

Input

The first line contains one integer $T(1\leq T\leq 2\times 10^4)$.

For the next T lines, each line contains a triple (n,u) represents a question$(1\leq n\leq 10^5, 1\leq u\leq 10^5)$.

Output

For each question, output your answer.

Sample

Sample Input 1	Sample Output 1
3 8 4 5 3 10 9	118 38 58

Problem E. SDL's Number

- Input File: `stdin`
- Output File: `stdout`
- Time Limit: `1000 MS`
- Memory Limit: `256 MB`

One day SDL had a strange idea and wrote a bunch of numbers $l, l+1, l+2, \cdots, r-1, r$, he hoped to know if he stitched these numbers together in order and formed a new number x, what is the value of x mod 9?

For example, $l = 1, r = 3$, $x = 123$,$x \bmod 9 = 6$.

Input

The first line contains an integer $T(T \leq 5000)$, which means the number of queries.

For the next T lines, each line contains two integer $l, r(0 < l \leq r \leq 10^{12})$.

Output

For each query, output an integer representing x mod 9.

Sample

Sample Input 1	Sample Output 1
1 3	6

Problem F. Game

- Input File: `stdin`
- Output File: `stdout`
- Time Limit: `1000 MS`
- Memory Limit: `64 MB`

Little A and little B are playing stone games.There are N bottles,labeled 1 to N,each with a capacity of A_i.In each round,the bottles are all full.

In each turn,the player takes a certain amount of stones from every bottle. Different from the past, each person takes up at most r_i stones from each bottle, ($r_i <= a_i$), and at least takes l_i stones ($l_i >= 0$),and put them into the stone library(initially empty).

When r_i is greater than $a_i, r_i = min(r_i, a_i)$, when l_i is greater than a_i, $l_i = min(l_i, a_i)$. You can also choose to take some of the stones from the stone library (independent of the stone pile) and put them back into the original bottle.The first person who takes all the stones wins.

You can assume that everyone takes the best strategy.

Input

First line, $N, K(1 <= K <= N <= 1e6)$.
The second line $a_1, a_2, ..., a_n(0 <= ai <= 1e9)$.
The third line $l_1, l_2, ..., l_n$.
The fourth line $r_1, r_2, ..., r_n(0 <= l_i <= r_i <= a_i)$.

Output

If A wins,print one line "`A`", else if B wins,print one line "`B`", else print one line "`Draw`".

Sample

Sample Input 1	Sample Output 1
3 0 1 2 0 0 0 0 1 2	A

Problem G. SDL and popcorn chicken

- Input File: `stdin`
- Output File: `stdout`
- Time Limit: `1000 MS`
- Memory Limit: `256 MB`

As a new participation of SDL' s fan club, you have been told that SDL has successfully cut down his weight, and take fancy to popcorn chicken again.

It is a lovely Thursday, SDL goes to K*C to enjoy half-price popcorn chicken as usual. However, the boss do not want SDL to have too much popcorn chicken (maybe it is because popcorn chicken is unhealthy), he gives a easy problem to SDL and only when SDL solves this problem can he enjoy popcorn chicken happily. As a super fan of SDL, can you help him solve this problem?

There are totally N boxes of popcorn chicken(shorten as boxes in following description) distribute on the coordinate system, each box have a coordinate (x_i, y_i) (it is guaranteed that there are no two boxes share one coordinate). SDL have to collect as much boxes as possible. Initially, he begins at the box which y_i is lowest, and go to another box. But strict boss give SDL a restriction: He can not turn right when heading to next box. That is to say, the next box SDL wants to collect ought to be on the left side(or on the extension line) of the vector formed by the previous movement. Your task is to calculate how many boxes SDL can collect.

Input

The fist line contains one integers N. Describing the number of popcorn chicken boxes.
In next N lines, every line contain two integers x_i, y_i describing the coordinate of the box.
Note that: $1 < N < 1000$,$0 < x_i, y_i \leq 10000$.

Output

In first line, output how many boxes SDL can collect.
In the next line, out put the order that SDL collect all possible boxes.

Sample

Sample Input 1	Sample Output 1
5 2 3 3 4 3 6 5 7 4 1	5 5 4 3 1 2

Problem H. Sdl all kill problems

- Input File: `stdin`
- Output File: `stdout`
- Time Limit: `5000 MS`
- Memory Limit: `256 MB`

Sdl has to be completed in the multi-school competition, but he feels very bored after ak,he feels that there may be some connection between the problems so he can do it so fast. If the description of each problem can be thought of as a string, the longest common prefix between each two strings is the link between the two problems.But the greedy sdl wants the connection between any two problems. Can you help him? That is to say, give you n strings, m queries, and ask for the longest common prefix between the two strings at a time.

Input

Input consists of a number of test cases. The first line of input contains the number of test cases T ($T \leq 20$). Immediately following this line are T cases. Each case starts with the number of strings $N(1 \leq N \leq 1 \times 10^5)$ on a line by itself. The following N lines each contain a single non-empty string made up entirely of lower case English characters('`a`' to '`z`'), whose length L is guaranteed to be less than or equal to $10,0000$. In every case it is guaranteed that $N \times L \leq \times 10^6$. The line following the last string contains a single integer $Q(1 \leq Q \leq 1 \times 10^6)$, the number of queries.

Each of the Q lines following contain a query made up of two integers i and j separated by whitespace$(1 \leq i, j \leq N)$.

Output

The output consists of T cases, each starting with a single line with '`Case X:`', where X indicates the X-th case. There should be exactly Q lines after that for each case. Each of those Q lines should contain an integer that is the answer to the corresponding query in the input.

Sample

Sample Input 1	Sample Output 1
2 5 asdwqdasda asdasdqwdqw asdsvasdadwq casacas qweaasdas 4 1 2 1 3 1 4 1 5 2 bb bb 1 1 2	Case 1: 3 3 0 0 Case 2: 2

Problem I. Sdl and math problem

- Input File: `stdin`
- Output File: `stdout`
- Time Limit: `1000 MS`
- Memory Limit: `256 MB`

In order to improve the ACM level of Sdl National Fan Support Association (https://acm.njupt.edu.cn/group/sdlnb) members, Sdl decided to conduct a series of ACM training for members.

This time it is the turn of the math project, and because it has only just begun, Sdl do not want to crack down on Mengxin's self-confidence, so he thinks of a simpler question: What is the K-th incomplete cubic number in a positive integer?

An incomplete cubic number is the cubic root of a number that is not an integer, such as 2, 3, 4, 5, 6, 7, 9... In addition, if you can't answer this question, he will sent you to Africa.

Input

Input starts with an integer $T(T \leq 10^5)$, denoting the number of test cases.
Each input line contains a single integer $K(1 \leq K \leq 10^{18})$.

Output

For each case, print one number per line, the K-th incomplete cubic number.
See the samples for details.

Sample

Sample Input 1	Sample Output 1
3 1 2 7	2 3 9

Problem J. Sdl and McDonald's

- Input File: `stdin`
- Output File: `stdout`
- Time Limit: `1000 MS`
- Memory Limit: `256 MB`

"I love three things:the sun, the moon and you. The sun is for the day, the moon is for the night, and you forever."

"But you are so huge..."

"I saw you. You smiled warmly. The world began waking up."

"But you are so huge..."

"Without you, however beautiful the city is, it is just null."

"But you are so huge..."

Sdl recently hates McDonald's very much. Because the girl that Sdl loves rejected him because of his weight, the chief culprit that causes Sdl's huge weight is McDonald's. So Sdl decided to lose weight.

One day, Sdl came to calorie kingdom. There was a magical medicine in calorie kingdom. After eating one medicine you will lose 1 weight. After Sdl's continous pleas, the king of calorie kingdom allowed Sdl to go to the place where the magical medicine were stored. There are n pillars (numbered 1 to n), each of which has a_i magical medicine above, and the height of each pillar is h_i.

In order to eat the magical medicine, Sdl needs to jump to the pillar. Sdl's jumping ability is b (Jumping ability is the ability to jump from a low position to a high position. Jumping from a high position to a low position is unaffected). However, the king told Sdl that some of the magical medicine may have expired due to the long time of not opening the warehouse, and taking expired medicine will increase 1 weight (represented by a negative value -1). And the king ordered Sdl that he just can only jump forward, not backward, meaning Sdl can only jump from i to j, where $i \leq j$. And Sdl must to eat all the magical medicine when he gets. In order to regain the girl's heart, Sdl must keep his weight to a minimum number.

Now Sdl is on the first pillar. What is the maximum weight that Sdl can lose after this journey?

Input

Input starts with tow integer $n(1 \leq n \leq 2 \cdot 10^5)$ and $b(0 \leq b \leq 2 \cdot 10^6)$, denoting the number of the pillar and Sdl's jumping ability.

In the second line, there are n integers $a_i(a_i \leq 2 \cdot 10^6)$, denoting the number of the magical medicine on i_{th} pillar. If magical medicines are not expired, a_i is a non-negative integer. If magical medicines are expired, a_i is a negative integer.

In the third line, there are n integers $h_i(1 \leq h_i \leq 2 \cdot 10^6)$, denoting the height of the i_{th} pillar. It guarantees that a_1 equals to zero.

Output

Print a single line containing an integer, denoting the minimum weight of Sdl after this journey.

Sample

Sample Input 1	Sample Output 1
4 100 0 1 100 -2 3 100 900 3	1

Sample Input 2	Sample Output 2
4 1000 0 1 100 -2 3 100 900 3	101