2024 Florida International University
High School Programming Competition

Problems – Novice Division

	Problem Name	Filename
A.	Alphabetical Anomalies!	alphabet
B.	Panther Optimization	panther
C.	Server Assignment	server
D.	A Day Off	dayoff
E.	Cow Farm	farm
F.	A Fun Trip	trip
G.	Group Project	group
H.	Texting Mania	mania
I.	Secret Password	secret
J.	Unstable Compounds	unstable

Call your program file: filename.cpp, filename.java, or filename.py

B. Panther Optimization

Geeyawn is a student at the Paul L. Cejas School of Architecture here at FIU, and he has just received his homework from one of his professors: the pathways connecting certain buildings have eroded sufficiently enough such that FIU now needs to do something about it. For the assignment, Geeyawn's professor has given him a list of the buildings of concern, as well as a list of pairs of these buildings for which the pathway connecting them needs fixing, along with the cost of fixing each pathway. In giving Geeyawn this assignment, his professor has noted the following:

- Not all pathways need to be fixed. The professor wants the students to choose a subset of the pathways such that the total cost of fixing the chosen pathways is *minimized*, and that it would be possible to reach every building from every other building using the fixed pathways.
- The professor has guaranteed the existence of a set of pathways to satisfy the above conditions. In fact, it may be that there are *multiple* valid answers, so the the students may pick one of them arbitrarily. The pathways to be fixed may be output in any order the students chooses.
- Connecting any two buildings, there is *at most* one pathway.

Geeyawn's ability in architecture is rather poor, and so he has sought your guidance in completing this homework assignment.

Input

The first line of the input will contain two space-separated integers n and m, where $1 \leq n \leq 500000$ and $n-1 \leq m \leq \min\{500000, \frac{n(n-1)}{2}\}$. Here n is the number of buildings to be considered, and m is the total number of pathways that need fixing.

The next n lines of the input will each contain a single string $s, 1 \leq |s| \leq 10$, containing uppercase English letters or digits, which represent the buildings of concern in this ordeal.

The next m lines of the input will each contain pairs of building names a and b, as well as a positive integer $c, 1 \leq c \leq 5000$, all space-separated. Thus, the cost of fixing the pathway connecting the buildings a and b will be c dollars.

Output

The first line of output should contain a single integer u, the total cost of fixing the optimal set of pathways. The following lines, one for each pathway in the optimal set of pathways to be fixed, should contain two strings, space-separated, denoting the pair of buildings connected by the pathways chosen to be fixed.

Sample Input 1

```
4 4
A
B
C
D
A B 1
B D 3
B C 2
C D 0
```

Sample Output 1

```
3
A B
B C
C D
```

Sample Input 2

```
8 8
A
B
C
D
E
F
G
H
A B 3
A D 3
A F 4
B C 1
B H 1
C G 2
E F 1
D E 1
```

Sample Output 2

```
12
B C
B H
D E
E F
C G
A B
A D
```

C. Server Assignment

Ricky has been setting up this year's high school programming competition. He has been working on setting up the judging servers. During the contest, contestants will a series of submissions which will be automatically judged by a computer. A pool of computers have been set aside to do the judging, and submissions will be assigned to computers from this pool to judge. Each submission has two pieces of information, the submission time and the time it takes to judge that submission. This being a tedious process, we would like to automate it. Since Ricky has been busy, he doesn't have time to implement this, but he does know how computer assignment should be handled. Initially, there will be n computers with IDs 1 through n, when a contestant makes a submission, if there are multiple computers available, the computer that has spent the least total amount of time judging submissions will be assigned to judge that submission; if two or more computers are tied for the least amount of time spent judging, then the one with the smallest ID will be picked. If no computers are available, then the submission will be placed in a queue to wait for the first available computer. Multiple submissions may be in the queue at any given time. If multiple submissions arrive at the same time, they will be ordered by increasing time needed to judge them.

Input

The first line of the input will contain two integers m ($1 \leq m \leq 10^5$) and n ($1 \leq n \leq 10^5$), the number of submissions and the numbers of computers, respectively. The following m lines will each contain two integers t ($1 \leq t \leq 10^9$) and v ($1 \leq v \leq 10^9$), where t is the time of the submission and v is the time it takes to judge that submission.

Output

Output n space-separated integers: the total time that each of the n computers spent on judging problems, going in ascending order of ID.

Sample Input 1

```
4 2
2 3
1 7
1 4
5 10
```

Sample Output 1

```
17 7
```

Initially two submission arrive at $t = 1$ the one that takes 4 seconds to process is given to computer 1. The second one is given to computer 2.

Sample Input 2

```
6 2
1 1
10 3
4 10
3 1
3 2
2 5
```

Sample Output 2

```
14 8
```

D. A Day Off

Aaarush and Bronit are on their day off, and they're up to their usual shenanigans and antics as they're making their way to the Green Library so they can acquire some of that sweet, sweet Starbucks. On their way to the Green Library, however, they spotted some stones of varying sizes arranged in a single file right beside the lake in front of the Green Library. Was this the work of mother nature, or did some other student have a hand in this? Since thinking is too hard for Aaarush and Bronit (evident from their scores on their Programming 1 finals last semester), they promptly dismissed such thoughts, and started playing with the stones. They learned that each stone also had an associated value. They devised the following game with the stones: In a single turn, a player must take at least one and no more than three stones from the *beginning* of the row of stones. Once they do this, the sum of the values of the stones that player took is added to their score. They repeat this until no stones remain, which is when the game ends. Whichever player ends with a higher score wins, unless both of their scores are equal, in which case they call it a tie. Aaarush will always be playing first because he *just barely* beat Bronit's score on their Programming 1 final. Since Aaarush and Bronit don't know how to play this game optimally, they would like you, an expert game theorist, to tell them which one of them would win this game under the assumption that they both play optimally.

Input

The first line of the input will contain a single integer $1 \leq n \leq 50000$, the number of stones they found along the lake. The second line of the input will contain a list $a_1, a_2, \ldots, a_n$ of the values of stones listed in a space-separated manner. Note that $1 \leq |a_i| \leq 1000$.

Output

Output the player who will end up with the greater sum and will therefore win, assuming they play optimally. Output `Aaarush` if Aaarush will win, output `Bronit` if Bronit will win, and output `Tie` if the game will end in a tie.

Sample Input 1

```
4
1 2 3 7
```

Sample Output 1

```
Bronit
```

For the above case, no matter how many stones Aaarush takes, Bronit will always be able to take *at least* the last stone, which makes him the winner, as that stone has a sum greater than the first three stones.

Sample Input 2

```
4
1 2 3 6
```

Sample Output 2

```
Tie
```

For the above case, Aaarush can force a tie by taking all three of the first stones in the row (any other move would *not* be optimal for Aaarush).

Sample Input 3

```
5
1 2 -5 1 3
```

Sample Output 3

```
Aaarush
```

For the above case, Aaarush's optimal move would be to take the first two stones. Then, Bronit makes his optimal move of taking the last three stones, but because the third stone was just so ugly (and he must take *at least* this stone), Aaarush ends up being victorious.

E. Cow Farm

John, a farmer, owns several cows in his pasture. Recently, a tornado damaged the fences enclosing his cows, and now he needs to reconstruct them. As John prefers simplicity, he plans to build a rectangular fence. The goal is to determine the perimeter of the smallest rectangular fence that encompasses all his cows, given the (x, y) coordinates of each cow. Any cows sitting *on* the fence are considered to be within the enclosure. It is guaranteed that the minimum enclosing rectangle will have a non-zero area. John would also like to construct the rectangle such that the sides are parallel or perpendicular to the x and y axes.

Input

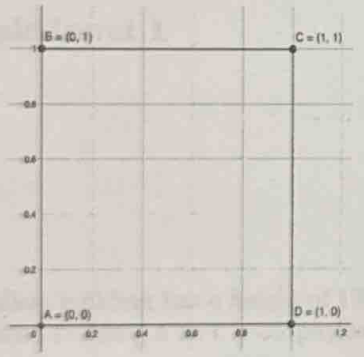
The first line of the input will contain one integers n ($3 \leq n \leq 10^5$) the number of cows in the pasture. The following n lines contain two integers x ($-10^9 \leq x \leq 10^9$), y ($-10^9 \leq y \leq 10^9$) the x and y position of the cows.

Output

Output the minimum perimeter of a XY-aligned rectangle needed to enclose all of the cows.

Sample Input 1

```
4
0 0
1 1
1 0
0 1
```

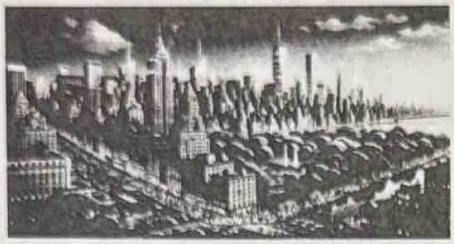


A square with perimeter 4 can include all points

Sample Output 1

```
4
```

F. A Fun Trip



Ricky is going to New York for the first time. Ricky is interested in finding the best view in the city. He thinks that the tallest building will give him the best view in town. Ricky will visit n buildings. They are numbered from 1 to n and have a specified height. You will be given an integer list of size n containing the heights of the n buildings. Write a program to output the number of the tallest building. If two buildings are of the same height, then output the one with the smaller number.

Input

The first line of the input will contain one integer n ($1 \leq n \leq 10^3$) the number of buildings. The following n lines contain one integer h ($1 \leq h \leq 10^9$) which is the height of building i with the first one is building 1.

Output

Output the number of the tallest building.

Sample Input 1

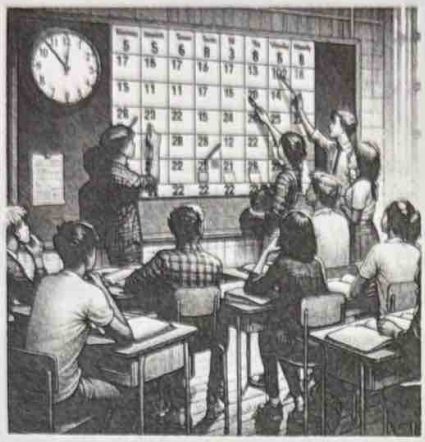
```
5
10
1
2
7
10
```

Sample Output 1

```
1
```

The tallest building has a height of 10, both building 1 and building 5 are the tallest, but building 1 is before building 5 so 1 is outputted.

G. Group Project



You and your friends have to do a group project for science class. But not everyone has the same schedule so you need to find a time when everyone is available. Your friends' group is of size n, including yourself. Each person's schedule is represented as an interval $[a, b]$ meaning that the person is available from time a to time b. Since the group project is short you only need 1 hour to finish it. Find out if there is a time when everyone is simultaneously available for at least 1 hour.

Input

The first line of input is an integer n ($2 \leq n \leq 500$) which is the number of total friends, including yourself. The following n lines provide the availability of each of the n individuals in order, one line per individual. The availability of each individual starts with an integer m ($1 \leq m \leq 20$), after which follows a line with $2m$ pairs of integers $a_1\ b_1\ a_2\ b_2\ \ldots\ a_m\ b_m$, indicating the m intervals $[a_1, b_1]$, $[a_2, b_2], \ldots, [a_m, b_m]$ during which that person is available. You can assume that $1 \leq a_i < b_i \leq 100$.

Output

Output `Yes` if there is a 1-hour time when everyone is simultaneously available, else output `No`.

Sample Input 1

```
4
1 1 5
2 1 2 4 5
1 4 9
1 2 8
```

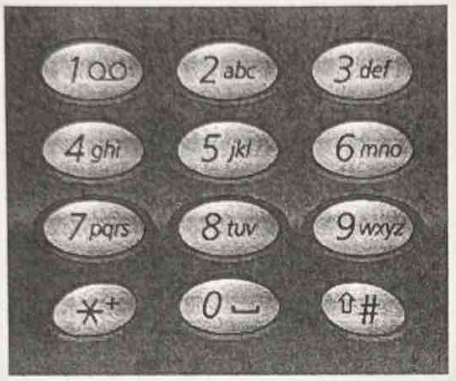
Sample Output 1

```
Yes
```

The answer for the test case is `Yes`, since everyone is free from time 4 to 5.

H. Texting Mania

Even though it's already 2024, Rick He is *still* using a flip phone, and as such, texting is an extremely tedious task for him. Texting on an old flip phone usually requires pressing a digit multiple times just to get a single letter out; for example, if you wanted to get the letter 'c', you would have to press the '2' digit three times. As an example, in order to type the string of letters abk, Rick He would need a total of five key presses: one to digit 2, two to digit 2, and two to digit 5.



Since this method of input is admittedly very slow, Rick He would like to remap certain letters to other keys (you tried to convince Rick He to get with the times, but he adores his flip phone too much, and as such, your efforts were in vain). Since Rick He is very intelligent, and already knows what he wants to type before typing it, he is redesigning the number pad with his specific input in mind: he will design the number pad such that the number of key presses needed to type this string is *minimized*. Note that after redesigning, all letters must map to *only* one digit between 2 and 9 inclusive (refer to the image). Additionally, Rick He is allowed to map as many letters to a digit as he likes. You may assume Rick He makes *no* typing mistakes and the phone *doesn't* allow copy-and-paste.

Input

The first and only line of input will contain a string $s, |s| \leq 10^5$, the string of lowercase English letters that Rick He wants to type.

Output

The first and only line of output will contain a single integer: the minimum number of key presses needed to type this string, provided that Rick He has been allowed to remap letters.

Sample Input 1	Sample Output 1
abcde	5

Sample Input 2	Sample Output 2
xyzxyzxyzxyz	12

Sample Input 3	Sample Output 3
aabbccddeeffgghhiiiiiii	24

I. Secret Password

You and your friend like to pass notes during class, but Dr. Giri will take those notes if he catches you guys. So you and your friend create a way to encode and decode the notes you pass. To encode a message the two friends first assign a binary string to each of the letters in the alphabet. Then, each letter is replaced by its respective binary string. For example, if they want to encode the message `hello` and we have the pairs (h, 10), (e, 0), (l, 111), (o, 110) the message becomes 100111111110. It is guaranteed that a letter's binary code is not the prefix of another letter. You will be given some pairs of lowercase letters with their respective binary encoding and the cipher-text. Output the original plain-text message. It is guaranteed that the cipher-text has a valid decoding.

Input

The first line of the input will contain one integer n ($1 \leq n \leq 26$) which refers to the number of letters that have a binary encoding. The following n lines contain a lowercase alphabetic character c and a binary string b, $|b| \leq 10$. All characters will be unique. The last line of input contains a binary string s, with $|s| \leq 10^5$

Output

The first and only line of output will contain the decryption of the cipher-text given in the input.

Sample Input 1

```
4
h 10
e 0
l 111
o 110
100111111110
```

Sample Output 1

```
hello
```

Explanation: The binary string can be broken down into 5 characters as follows:
10-0-111-111-110 This stands for the letters shown below.
h-e-l-l-o
final message: hello

J. Unstable Compounds

Your sibling is a chemistry nerd and loves creating new compounds. He has all his n chemicals lined up on the table. Each chemical has a value given by the values $A_1, A_2, \ldots, A_n$. When two chemicals in positions i and j are combined, they create a compound with value $|A_i - A_j|$. chemicals placed close to each other are highly reactive. Your sibling plans to combine pairs of chemicals but wishes to avoid combining two reactive chemicals. You are also given a fixed integer k. You have been informed that two chemicals i and j are reactive if $|i - j| < k$. Your sibling would like to find the pair of chemicals, which minimizes the value of the resulting compound when combined. Help your sibling by writing a program that finds the two chemicals that minimize this value, but which are at distance k or greater.

Input

The first line of the input contains two integers n ($3 \leq n \leq 10^5$), the length of the array, and k ($1 \leq k \leq n-1$), the smallest distance two chemicals can be from each other while being considered a stable pair. The following line contains the array A as a sequence of space-separated integers, where $1 \leq A_i \leq 10^9$.

Output

There should be one line of output which is the minimum value of the compound obtained by combining two stable chemicals at a distance k or greater.

Sample Input 1

```
5 2
1 1 1 2 2
```

Sample Output 1

```
0
```