Problem A: Paper Route

As a poor, tuition-ridden student, you've decided to take up a part time job as a paperboy/papergirl. You've just been handed your paper route: a set of addresses (conveniently labelled 1 to N).

Every morning, you start at the newspaper office (which happens to be address number 0). You have to plan a route to deliver a newspaper to every address - and you also want to get to class right after you're done.
Conveniently, there are only N roads in your area connecting the addresses, and each of them takes a known time to traverse.
Also, you've precalculated the time it takes to get to Waterloo campus from each address, including the newspaper office (through some combination of biking, busing, or hitching a ride).
How soon can you be done delivering papers and be in your seat at school?

Input Specification

First, there will be a single integer N (the number of addresses, $1 \leq N \leq 100{,}000$).
Next, there will be N+1 lines, each with an integer c_i (starting with $i = 0$, $0 \leq c_i \leq 1{,}000{,}000{,}000$), the time it takes to get from location i to campus.
Finally, the input will contain N lines, each with three integers a, b, c ($0 \leq a, b \leq N$, a != b, $0 \leq c \leq$ 1,000). Each of these lines describes a road between locations a and b taking c minutes to traverse. It is guaranteed that you will be able to reach all the addresses. (Remember that location 0 is the newspaper office.)

Sample Input

```
2
1
3
4
0 1 1
0 2 2
```

Output Specification

Output the minimum time it will take to deliver all the papers and get to class.

Output for Sample Input

```
7
```

It's actually better to visit all the addresses, go back to the office, and get to school from there.
An example route: 0 -> 1 -> 0 -> 2 -> 0 -> school = 1 + 1 + 2 + 2 + 1 = 7

Hanson Wang

Problem B: Octagons

Below is a picture of an infinite hyperbolic tessellation of octagons. If we think of this as a graph of vertices (of degree three), then there exists an isomorphism of the graph which maps any vertex x onto any other vertex y. Every edge is given a label from the set $\{a,b,c\}$ in such a way that every vertex has all three types of edges incident on it, and the labels alternate around each octagon. Part of this labeling is illustrated in the diagram.

So a path in this graph (starting from any vertex) can be specified by a sequence of edge labels. Your job is to write a program which, given a sequence of labels such as "abcbcbcabcaccabb", returns "closed" if the path ends on the same vertex where it starts, and returns "open" otherwise.

Input Specification

The input will begin with a number $Z \leq 200$ on a line by itself. This is followed by Z lines, each of which is a sequence of length at least 1 and at most 40 of 'a's 'b's and 'c's.

Sample Input

```
2
abababab
abcbcbcbcba
```

Output Specification

For each input instance, the output will be the words "closed" or "open", each on a single line.

Output for Sample Input

```
closed
open
```

Danny Sleator

Problem C: Party Location

After the programming contest, all of the contestants would like to throw a party. After the party, however, it will be late, and the contestants will be too tired to walk a long way home. In particular, each contestant refuses to come to the party if it is more than 2.5 km from his or her house.

The solution is to hold the party as close to as many of the contestants' houses as possible. This is where you come in: your job is to determine the optimal location for the party, so that as many contestants as possible will be willing to attend it.

We consider the city to be a flat square, 50 km on each side. A contestant can walk directly from the party in a straight line to his or her house (there are no obstacles).

Input Specification

Standard input consists of a number of lines, each containing two floating point numbers indicating the (x,y) coordinates of the house of one of the contestants. Each coordinate is between 0.0 and 50.0 (km). Each house is at a distinct location. There are at most 200 contestants.

Sample Input

```
4.0 4.0
4.0 5.0
5.0 6.0
1.0 20.0
1.0 21.0
1.0 22.0
1.0 25.0
1.0 26.0
```

Output Specification

Standard output consists of a single integer: the maximum number of contestants that can attend the party.

Output for Sample Input

```
4
```

Gordon V. Cormack, Ondřej Lhoták

Problem D: Numbersrebmun

Anna and Bob are starting up a new high-tech company. Of course, one of their key considerations is choosing a good name for the company. Palindromes are cool. (A palindrome is a word that is the same when reversed, like the names of our two entrepreneurs.) They would really the name of their company to be a palindrome. Unfortunately, they cannot think of a nifty company name that is also a palindrome.

Maybe at least the telephone number of their company could be a palindrome. However, they really want their customers to be able to call them, so they want to choose the company name so that, when it is typed using the letters printed on a phone keypad, the result is also their phone number. (On a standard phone keypad, the following keys contain the corresponding letters: 2: ABC, 3: DEF, 4: GHI, 5: JKL, 6: MNO, 7: PQRS, 8: TUV, 9: WXYZ.)

Input Specification

The first line of input contains a single integer, the number of lines to follow. Each following line contains a company name, which is a string of at most 20 letters, which may be either uppercase or lowercase.

Sample Input

```
2
ANBOBNA
iAmACoolCompany
```

Output Specification

For each company name, print a single line of output, containing the word YES if the phone number is a palindrome, or NO if it is not.

Output for Sample Input

```
YES
NO
```

Ondřej Lhoták

Problem E: Class Schedule

At Fred Hacker's school, there are $T \times C$ classes, divided into C catagories of T classes each. The day begins with all the category 1 classes being taught simultaneously. These all end at the same time, and then all the category 2 classes are taught, etc. Fred has to take exactly one class in each category. His goal is to choose the set of classes that will minimize the amount of ``energy'' required to carry out his daily schedule.

The energy requirement of a schedule is the sum of the energy requirement of the classes themselves, and energy consumed by moving from one class to the next through the schedule.

More specifically, taking the jth class in the ith category uses E_{ij} units of energy. The rooms where classes take place are located at integer positions (ranging from 0 to L) along a single hallway. The jth class in the ith category is located at position P_{ij}. Fred starts the day at position 0, moves from class to class, according to his chosen schedule, and finally exits at location L. Moving a distance d uses d units of energy.

Input Specification

The first line of the input is $Z \leq 20$ the number of test cases. This is followed by Z test cases. Each test case begins with three space-separated integers: C, T, and L. Each of the following C× T lines gives, respectively, the location and energy consumption of a class. The first T lines represent the classes of category 1, the next T lines represent the classes of category 2, and so on. No two classes in the same category will have the same location.

Bounds

$1 \leq C \leq 25$
$1 \leq T \leq 1000$
$1 \leq L \leq 1{,}000{,}000$
$1 \leq E_{ij} \leq 1{,}000{,}000$
$0 \leq P_{ii} \leq L$

Output Specification

For each input instance, the output will be a single integer on a line by itself which is the minimum possible energy of a schedule satisfying the constraints.

<table>
<tr><td>Sample Input<pre>1
3 2 5
2 1
3 1
4 1
1 3
1 4
3 2</pre></td><td>Fred must take 3 classes every day, and for each he has 2 choices. The hall has length 5. His first possible class is located at position 2 and will take 1 unit of energy each day, etc.</td></tr>
<tr><td>Output for Sample Input<pre>11</pre></td><td>Here is one way to obtain the minimum energy:
Go to the class at location 2. Energy used: 3
Next, go to the class at location 4. Energy used: 6
Then go to the class at location 3. Energy used: 9
Finally, leave the school at location 5. Energy used: 11</td></tr>
</table>

Neal Wu