

SMS Dictionary

(Indian National Olympiad in Informatics, INOI, 2007)

In the pre-smartphone era, most mobile phones with numeric keypads had a private dictionary of words to allow users to type messages quicker. On a typical phone of this type, each number key is assigned a subset of the alphabet $\{a,b,\dots,z\}$: 2 is assigned the subset $\{a,b,c\}$, 3 is assigned $\{d,e,f\}$, 4 is assigned $\{g,h,i\}$, 5 is assigned $\{j,k,l\}$, 6 is assigned $\{m,n,o\}$, 7 is assigned $\{p,q,r,s\}$, 8 is assigned $\{t,u,v\}$ and 9 is assigned $\{w,x,y,z\}$.

When the user types a sequence of numbers, this sequence is mapped to all possible words that can be constructed from the key assignment. For instance, if the user types 66, this could correspond to any one of the letter sequences "mm", "mn", "mo", "nm", "nn", "no", "om", "on" or "oo". These letter sequences are looked up in the dictionary stored in the phone and all matches are reported. For instance, if the phone's dictionary contains only "on" and "no" from this set of sequences, the user will be offered a choice of "on" or "no" to insert in the message. Similarly, the input 4663 might be interpreted as either "good" or "home". An input sequence may have a unique interpretation---for example, the only word in the dictionary matching the input 28 may be "at". Other sequences may not match any word in the dictionary—for instance, 99999.

Your task is the following. Given the typical assignment from number keys to letters of the alphabet given above and given a dictionary of words, report the input sequence that matches the largest number of words in the dictionary. For example, if the dictionary consists of the words $\{at,on,good,no\}$ then the answer is 66, because 66 matches both "on" and "no" while no other input matches more than one word in the dictionary. On the other hand, with the dictionary $\{at,on,good,no,home,gone\}$, the answer is 4663, because 4663 matches three words, "good", "home" and "gone" in the dictionary.

Solution hint

For each word in the input, compute the number key sequence that creates it by inverting the mapping $2 \rightarrow \{a,b,c\}$, $3 \rightarrow \{d,e,f\}$ etc. Store the number corresponding to the word in an array.

After reading all the input, sort the numbers in the array.

Input format

The first line of input is an integer M , the number of words in the dictionary. This is followed by M lines of input. Each line contain one word from the dictionary, where a word is sequence of characters from the lowercase alphabet $\{a,b,c,\dots,z\}$.

Note: Each word in the dictionary is, in general, an arbitrary sequence of letters from $\{a,b,c,\dots,z\}$. In particular, it is not assumed that the words stored in the dictionary are valid words in English or any other language.

Output format

A single line containing the input sequence that matches the maximum number of words in the dictionary. If multiple input sequences qualify for the maximum number of matches, it suffices to report any one.

Test data

For all inputs, $1 \leq M \leq 100000$. Each word in the dictionary is at most 8 characters long. In 50% of the inputs, $1 \leq M \leq 1000$.

Example

Here is the sample input and output corresponding to the example discussed above.

Sample input 1

```
4
at
on
good
no
```

Sample output 1

```
66
```

Sample input 2

```
6
at
on
good
no
home
gone
```

Sample output 2

```
4663
```

The Great Escape

(IARCS OPC Archive, K Narayan Kumar, CMI)

Heroes in Indian movies are capable of superhuman feats. For example, they can jump between buildings, jump onto and from running trains, catch bullets with their hands and teeth and so on. A perceptive follower of such movies would have noticed that there are limits to what even the superheroes can do. For example, if the hero could directly jump to his ultimate destination, that would reduce the action sequence to nothing and thus make the movie quite boring. So he typically labours through a series of superhuman steps to reach his ultimate destination.

In this problem, our hero has to save his wife/mother/child/dog/... held captive by the nasty villain on the top floor of a tall building in the centre of Bombay/Bangkok/Kuala Lumpur/.... Our hero is on top of a (different) building. In order to make the action "interesting" the director has decided that the hero can only jump between buildings that are "close" to each other. The director decides which pairs of buildings are close enough and which are not.

Given the list of buildings, the identity of the building where the hero begins his search, the identity of the building where the captive (wife/mother/child/dog...) is held, and the set of pairs of buildings that the hero can jump across, your aim is determine whether it is possible for the hero to reach the captive. And, if he can reach the captive he would like to do so with minimum number of jumps.

Here is an example. There are 5 buildings, numbered $1, 2, \dots, 5$, the hero stands on building 1 and the captive is on building 4 . The director has decided that buildings 1 and 3 , 2 and 3 , 1 and 2 , 3 and 5 and 4 and 5 are close enough for the hero to jump across. The hero can save the captive by jumping from 1 to 3 and then from 3 to 5 and finally from 5 to 4 . (Note that if i and j are close then the hero can jump from i to j as well as from j to i .) In this example, the hero could have also reached 4 by jumping from 1 to 2 , 2 to 3 , 3 to 5 and finally from 5 to 4 . The first route uses 3 jumps while the second one uses 4 jumps. You can verify that 3 jumps is the best possible.

If the director decides that the only pairs of buildings that are close enough are 1 and 3 , 1 and 2 and 4 and 5 , then the hero would not be able to reach building 4 to save the captive.

Solution hint

Build a graph and use breadth-first search (bfs) or depth-first search (dfs).

[Full solution](#)

.

Input format

The first line of the input contains two integers N and M . N is the number of buildings: we assume that our buildings are numbered $1, 2, \dots, N$. M is the number of pairs of buildings that the director lists as being close enough to jump from one to the other. Each of the next M lines, lines $2, \dots, M+1$,

contains a pair of integers representing a pair of buildings that are close. Line $i+1$ contains integers A_i and B_i , $1 \leq A_i \leq N$ and $1 \leq B_i \leq N$, indicating that buildings A_i and B_i are close enough. The last line, line $M+2$ contains a pair of integers S and T , where S is the building from which the Hero starts his search and T is the building where the captive is held.

Output format

If the hero cannot reach the captive print 0. If the hero can reach the captive print out a single integer indicating the number of jumps in the shortest route (in terms of the number of jumps) to reach the captive.

Test Data:

You may assume that $1 \leq N \leq 3500$ and $1 \leq M \leq 1000000$. Further, in at least 50% of the inputs $1 \leq N \leq 1000$ and $1 \leq M \leq 200000$.

Example:

Here are the inputs and outputs corresponding to the two examples discussed above.

Sample Input 1:

```
5 5
1 3
2 3
1 2
3 5
4 5
1 4
```

Sample Output 1:

```
3
```

Sample Input 2:

```
5 3
1 3
1 2
4 5
1 4
```

Sample Output 2:

```
0
```

Number Triples

(IARCS OPC Archive, K Narayan Kumar, CMI)

In this problem you will be given a sequence of triples of positive integers. For example:

```

1  2  5
5  2  6
1  3  8
8  1  11
1  1  6
10 1  6
11 3  6
10 4  8
10 1  11

```

Given a pair of numbers A and B , a chain connecting A and B is a sequence of triples

$$A_0 \ W_0 \ A_1, \ A_1 \ W_1 \ A_2, \ A_2 \ W_2 \ A_3, \ \dots \ A_{k-2} \ W_{k-2} \ A_{k-1}, \ A_{k-1} \ W_{k-1} \ A_k$$

such that

- $A_0 = A$
- $A_k = B$
- For each i , $0 \leq i < k$, either the triple $A_i \ W_i \ A_{i+1}$ or the triple $A_{i+1} \ W_i \ A_i$ is in the given set of triples.

The value of such a chain is the sum of the W_i s in the chain. For example, here is a chain connecting 1 and 11 using the triples listed above:

```
1  1  6,  6  3  11
```

The value of this chain is $1+3 = 4$.

Here is another chain connecting 1 and 11.

```
1  1  6,  6  1  10,  10  1  11
```

The value of this chain is $1+1+1 = 3$. You can verify that among all chains connecting 1 and 11 this is the one with least value.

Sometimes there may be no chains connecting the given pair of numbers. For example, there is no chain connecting 1 and 2.

You will be given a sequence of triples and a pair of numbers. Your task is to find the value of the least value chain connecting the two numbers.

Solution hint

Construct a graph from the triples and use Dijkstra's algorithm for single-source shortest paths.

[Full solution](#)

Input format

The first line of the input has three numbers M , A and B . M is the number of triples. The next M lines (lines 2,3,..., $M+1$) describe the triples. Line $i+1$ contains the three positive integers X_i , Y_i and Z_i that make up the i th triple. Your task is to find the value of the least value chain connecting A and B .

Output format

If there is at least one chain connecting A and B the first line of the output must consist of a single word YES. In that case the second line must contain a single integer value indicating the value of the least valued chain connecting A and B . If there are no chains connecting A and B the output should contain a single line with the word no on it.

Test Data:

You may assume that $1 \leq X_i, Y_i, Z_i \leq 2000$ and $M \leq 4000000$.

Example:

We illustrate the input and output format using the above example:

Sample Input 1:

```
9 1 11
1 2 5
5 2 6
1 3 8
8 1 11
1 1 6
10 1 6
11 3 6
10 4 8
10 1 11
```

Sample Output 1:

```
YES
3
```

Sample Input 2:

```
9 1 2
1 2 5
5 2 6
1 3 8
8 1 11
1 1 6
10 1 6
11 3 6
10 4 8
10 1 11
```

Sample Output 2:

NO

Advertising hoardings

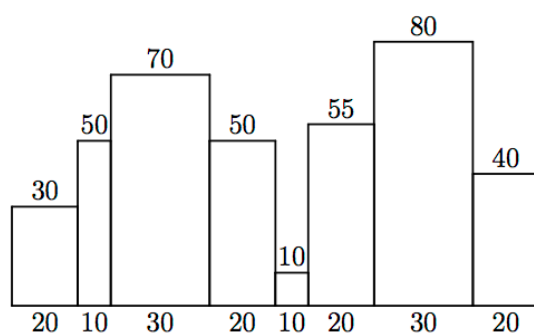
(IARCS IOI Training Camp 2006, Preyas Popat)

Downtown Siruseri is booming and the main road is now lined with a continuous row of skyscrapers. For aesthetic reasons, the Siruseri Municipal Corporation had banned advertising hoardings along the main road.

The High Court has squashed the Municipal Corporation's ban as unconstitutional and ruled that rectangular hoardings may be put up parallel to the main road, along the walls of the skyscrapers lining the road. The only restriction is that a hoarding may not extend above the height of any building along its length.

Advertising agencies across the city are buzzing with excitement at the news and accounts executives are busy calculating how large they can make their hoardings, in terms of total area, to collect maximum revenue from advertisers.

For instance suppose the Siruseri skyline appears as follows, where the number at the top of each building specifies its height while the number at the bottom of each building specifies its width, both quantities measured in metres.



In this situation, the biggest legal hoarding spans the second, third and fourth buildings with an area of $60 \times 50 = 3000$ square metres. The next best possibility is a hoarding of $70 \times 40 = 2800$ square metres, spanning the last three buildings.

Your task is to compute the maximum area that a hoarding can have for a given arrangement of skyscrapers.

Solution

The largest rectangle will be as tall as the shortest building it covers. For each building height, we need to find the widest rectangle of that height.

Partition the buildings into segments of length 1. Process the buildings in decreasing order of height as follows. For each building, if it is shorter than the building on its left, merge it with the segment on its left. Likewise, if it is shorter than the building on its right, merge it with the segment on its right.

Maintain the size of the maximum rectangle seen so far.

Use the union-find data structure to keep track of segments efficiently.

[Detailed solution](#)

Input format

The first line of input is an integer N , the number of skyscrapers. This is followed by N lines of input describing the skyscrapers in left to right order. Each of these lines consists of two integers W and H , specifying the width and height of the skyscraper, respectively.

Output format

A single integer denoting the maximum area of a legal hoarding given the arrangement of skyscrapers.

Test Data

In 10% of the inputs, $N = 10^5$. In the remaining 90% of the inputs, you may assume $1 \leq N \leq 5000$.
In 10% of the inputs, $1 \leq N \leq 500$.

Sample input

```
8
20 30
10 50
30 70
20 50
10 10
20 55
30 80
20 40
```

Sample output

```
3000
```

Dividing Sequences

(IARCS OPC Archive, K Narayan Kumar, CMI)

This problem is about sequences of positive integers a_1, a_2, \dots, a_N . A subsequence of a sequence is anything obtained by dropping some of the elements. For example, $3, 7, 11, 3$ is a subsequence of $6, 3, 11, 5, 7, 4, 3, 11, 5, 3$, but $3, 3, 7$ is not a subsequence of $6, 3, 11, 5, 7, 4, 3, 11, 5, 3$.

A *fully dividing sequence* is a sequence a_1, a_2, \dots, a_N where a_i divides a_j whenever $i < j$. For example, $3, 15, 60, 720$ is a fully dividing sequence.

Given a sequence of integers your aim is to find the length of the longest fully dividing subsequence of this sequence.

Consider the sequence $2, 3, 7, 8, 14, 39, 145, 76, 320$

It has a fully dividing sequence of length 3, namely $2, 8, 320$, but none of length 4 or greater.

Consider the sequence $2, 11, 16, 12, 36, 60, 71, 17, 29, 144, 288, 129, 432, 993$.

It has two fully dividing subsequences of length 5,

- $2, 11, 16, 12, 36, 60, 71, 17, 29, 144, 288, 129, 432, 993$ and
- $2, 11, 16, 12, 36, 60, 71, 17, 29, 144, 288, 129, 432, 993$

and none of length 6 or greater.

Solution hint

Let the input be a_1, a_2, \dots, a_N . Let us define $Best(i)$ to be the length of longest dividing sequence in a_1, a_2, \dots, a_i that includes a_i .

Write an expression for $Best(i)$ in terms of $Best(j)$ with $j < i$, with base case $Best(1) = 1$. Solve this recurrence using dynamic programming.

[Full solution](#)

.

Input format

The first line of input contains a single positive integer N indicating the length of the input sequence. Lines $2, \dots, N+1$ contain one integer each. The integer on line $i+1$ is a_i .

Output format

Your output should consist of a single integer indicating the length of the longest fully dividing

subsequence of the input sequence.

Test Data

You may assume that $N \leq 10000$.

Example:

Here are the inputs and outputs corresponding to the two examples discussed above.

Sample input 1:

```
9
2
3
7
8
14
39
145
76
320
```

Sample output 1:

```
3
```

Sample input 2:

```
14
2
11
16
12
36
60
71
17
29
144
288
129
432
993
```

Sample output 2:

```
5
```

Test data:

-

