DATA STRUCTURES AND ALGORITHMS

ASSIGNMENT-3
Language allowed: C

January 23, 2020

A. Majority Element

In this task you are supposed to find the *majority element* of a given array. A *majority element* of a given array of size N is defined as the element that occurs *atleast* N/2 times in it. If N is odd, apply the ceiling function on N/2. So, given an array, compute its majority element.

Input

The first line contains a single integer N ($1 \leq N \leq 10^5$) denoting the size of the given array A. The next line contains N space-separated integers ($-10^9 \leq A_i \leq 10^9$) denoting the array A.

Output

Print a single integer X, which is the majority element of the given array. If more than one majority element exist, print the greatest one. If no majority element is present, print "NO MAJORITY ELEMENT".

```
input
11
1 2 2 2 8 8 1 1 1 1 1

output
1
```

```
input
12
1 2 -7 -1 -1 7 -23 -23 -12 -67 12 78

output
NO MAJORITY ELEMENT
```

B. Longest Series

You have recently studied about the *Collatz series* and are quite enthralled by it. It goes as follows: for some number X_i (i^{th} element in the series) X_{i+1} is defined as:
a. $X_i/2$ if X_i is even, or
b. $3X_i + 1$ if X_i is odd
This sequence is continued till the series converges to 1. Seeing this series you begin to wonder which seed or starting number for this sequence produces the longest sequence (see sample case for clarity) and decide to write a program to do the same.

Input

The only line of input contains a single integer N ($1 \leq N \leq 5.10^5$) denoting the upper limit under which a seed that produces the longest sequence has to be found.

Output

Print two space-separated integers Z, L where
(a) Z ($1 \leq$ Z $\leq$ N) denotes the seed or starting number which produces the longest collatz sequence under N; and
(b) L denotes the length of the sequence generated by Z. If there are multiple answers for the seed that generates the longest sequence, output the greatest one.

```
input
10

output
9 20
```

```
input
100000

output
77031 351
```

C. Drone Flying

You and Harsh made a drone which can move in the 3-dimensional space. It is restricted to move only along the axes, one inch at a time. To test the drone, you want to move it from a point A to another point B. Harsh, being a numberphile, wants to find out the number of minimum-length paths the drone can take between those points. A minimum-length path is a path which has the least possible number of steps. However, he cannot crunch such huge numbers in his head, so he asks you for help. Given the coordinates of point A and B, find the number of minimum-length paths from A to B. Since this number can be very large, print it modulo 998244353.

Input

The first line of input contains three space-separated integers ($0 \leq X_1, Y_1, Z_1 \leq 2000$) denoting the initial position (Point A) of the drone. The next line also contains three space-separated integers ($0 \leq X_2, Y_2, Z_2 \leq 2000$) denoting the final position (Point B) the drone has to reach.

Output

Print the number of minimum-length paths modulo 998244353.

---

input
```
0 0 0
0 1 2
```

output
```
3
```

explanation
```
All possible minimum-length paths from (0, 0, 0) to (0, 1, 2) are
(0, 0, 0) → (0, 0, 1) → (0, 0, 2) → (0, 1, 2)
(0, 0, 0) → (0, 0, 1) → (0, 1, 1) → (0, 1, 2)
(0, 0, 0) → (0, 1, 0) → (0, 1, 1) → (0, 1, 2)
```

---

input
```
1553 664 1922
19 20 34
```

output
```
928421248
```

D. Silly Mistake

Your final English exam paper has a weird question on string matching. You were given a long string S and a word W. You were required to find the number of occurrences of the word in the string. In hurry, you made a silly mistake. If a substring of the given string matched the given word (letter by letter) with *atmost one letter off (one letter incorrectly substituted)*, you counted it as a valid occurrence of the word in the string. You realized your mistake after coming home and want to know how many times you counted the word in the sentence to know how much you'd be penalized and decide to write a program to do the same for you to ease the task.

Input

The first line of input contains a string S ($1 \leq |S| \leq 10^3$) denoting the string given to you in your exam. The second line contains the word W ($1 \leq |W| \leq |S|$) denoting the word whose occurrences had to be found in the string S. Both S and W will consist of lowercase letters.

Output

Output a single integer X, denoting the number of times you counted the word in the string (both correctly and incorrectly)

---

input
```
thisissometeststringimadeup
et
```

output
```
5
```

explanation
In this given testcase, you match the input word five times, first with *et*, second with *es* and third with *st*, fourth with *st* again and finally with *eu*.

---

input
```
aaaaaaaa
ab
```

output
```
7
```

E. Connect it!

In this task you are given a *non-circular singly linked list*. You need to make pointer manipulations in the following way:
a. the first node's next will point to the last node,
b. the last node's next will point to the second node,
c. the second node's next will point to the penultimate node
.
.
and so on till all the nodes are connected in this spiral fashion. The next of the last node of the resulting list is NULL.

Input

The only line of input contains a sequence of integers A ($-10^9 \leq A_i \leq 10^9$) denoting the data field of the linked list nodes. It is guaranteed that the length of the sequence will be less than 500.

Output

Print a sequence of integers B denoting the list after all manipulations. Note that just printing the list in the spiral form without making any manipulations should not be done and no use of arrays must be made to solve any part of the problem. Also, the pointer manipulations should be done on the given linked list itself and no new list should be made.

```
input
1234 -1 234 -6555 -4531 12 988 8970 16783 -2181 -74467 -463 87

output
1234 87 -1 -463 234 -74467 -6555 -2181 -4531 16783 12 8970 988
Note the next of 988 is NULL.
```

F. Gym Battles V2

Ash has just landed in the new region Galar and wants to part take in Gym Battles as soon as possible. This region has various cities and roads (undirected) between them such that a unique roadway exists between any two cities. The Gyms in this region are located in the *leaf cities*. A leaf city is a city that is connected to a single city. As Ash is really excited about his first gym match, help him find the nearest gym. Ash is currently in city 0 and all the cities are numbered from 0 to N - 1, N being the total number of cities.

Input

The first line of input contains a single integer N ($1 \leq N \leq 10^5$) denoting the number of cities in Galar. The next N-1 lines contain two space separated integers U_i and V_i ($0 \leq U_i, V_i \leq N-1$) denoting that a roadway exists between cities U_i and V_i.

Output

In the first line, print a single integer L denoting of number of cities in the region that have a gym and are closest to Ash. In the next line print a single integer X, denoting the least numbered city Ash can visit for a Gym Battle.

---

input

```
13
0 2
0 7
12 0
8 0
8 9
11 12
12 5
4 6
6 12
3 1
1 7
2 10
```

output

```
4
5
```

---

explanation

The leaf nodes in this case are : 10, 11, 5, 9, 4, 3. But the cities 3, 4 are a distance of 3 units from 0 and 10, 11, 5, 9 are at a distance of 2 units. Hence the cities that have gyms and that are closest to Ash are only 4, i.e. 10, 11, 5, 9. And out of those the minimum numbered city is 5.

G. Server Farm

Evil computer hacker Elzo D'Plaza is back to attack Foodle's server farm and planted a virus into the *main server*. But this time, the engineers at Foodle were alert and figured out the system in which virus was initially planted by Elzo and are ready with a remedy. As they cannot discard the main server, they plan to disconnect one smaller server from the farm so that the network is *broken* (into atleast two components) so that they can curb the damage. Help the engineers at Foodle figure out which server needs to be disconnected (see sample case for better clarity). The connections between the servers at Foodle are undirected.

Input

The first line of input contains two space-separated integers N ($1 \leq N \leq 10^3$) and M ($1 \leq M \leq \frac{N(N-1)}{2}$) denoting the number of servers in Foodle and the number of connections between those servers. The following M lines each have two space-separated integers U_i, V_i ($0 \leq U_i, V_i \leq N-1$) denoting a connection between servers U_i and V_i. The last line of input contains a single integer X, denoting the ID of the main server.

Output

Print a single integer Y, denoting the ID of the server which when removed (along with all its connections) will break the farm into atleast two components. If there are multiple answers, print the ID of that server which when disconnected leaves the main server connected to minimum number of other smaller servers. If no such server is present, print "BLACK OUT".

```
input
12 15
0 1
0 2
1 2
2 3
2 4
1 4
3 4
4 7
5 7
5 6
6 7
7 8
8 11
9 10
9 11
6

output
7
```

H. Gray Codes

In this task, you have to convert a given number in decimal to Gray Code and a given Gray Code to binary. Write a program to do the same.

Input

The first line contains a single integer Q ($1 \leq Q \leq 10^6$) denoting the number of queries for this problem. Each of the following Q lines contain queries in one of the following way:
A. *G* X: ($0 \leq X \leq 2^{32}$). This query is to convert a given integer (in decimal) to its corresponding gray code.
B. *D* S: ($1 \leq |S| \leq 32$). This query is to convert a given gray code (bit string) to its corresponding decimal equivalent.

Output

Print Q lines where the i^{th} line corresponds to the output of the i^{th} query. For query type A, print the gray code bit string of the given integer X and for query type B, print the a single integer which the decimal equivalent of the given gray code bit string (see sample case for clarity).

```
input
4
G 56
G 0
D 10010011
D 001011

output
100100
0
226
13
```

I. Detective Kudo

Shinichi Kudo is investigating the Black Organization and has found a major clue, a pile of digits. He knows that the boss's address is hidden in this pile of digits and is the *largest number can be formed by using some non-empty subset of those digits and is divisible by 30.* As Shinichi is quite tensed and is unable to figure out the address, it is now up to you, Haibara, to get that number.

Input

The first line of input contains a single integer N ($1 \leq N \leq 10^3$) denoting the number of digits in the pile. The next line contains a sequence of N space-separated digits $A_1, A_2, \ldots, A_N$ ($0 \leq A_i \leq 9$) denoting the pile of integers.

Output

Print a single integer X, denoting the largest number that can be formed from some non-empty subset of the given integers which is divisible by 30. If no number can be formed, print "NO NUMBER".

---

input
```
4
7 7 7 6
```

output
```
NO NUMBER
```

---

input
```
8
6 0 1 8 0 5 3 6
```

output
```
8663100
```

J. Tom and Jerry

The michevious cat, Tom, has tossed the little mouse Jerry into a large rectangular pool of water. Luckily, there are a few rubber ducks floating in the pool and Jerry can move only on them without sinking. Each rubber duck covers one sq. unit. Help Jerry mouse figure out a path on the rubber ducks from where he is, to a point from where he can safely jump out of the pool. (see input for better clarity).

Input

The first line of input contains two space-separated integers N, M ($2 \leq N, M \leq 500$) denoting the dimensions of the pool. Each of the next M lines contain N space-separated bits A_{ij} ($A_{ij} \in \{0, 1\}$) denoting the pool and the rubber ducks in it. If a cell is 1, it represents a rubber duck and if it is 0, it represents water. The next line contains two space-separated integers P_1 and Q_1 ($0 \leq P_1 \leq$ M-1, $0 \leq Q_1 \leq$ N-1) denoting the cell on which Jerry is present ($A_{P_1Q_1}$ is guaranteed to be 1). The last line of input contains two more space-separated integers P_2 and Q_2 ($0 \leq P_2 \leq$ M-1, $0 \leq Q_2 \leq$ N-1) denoting the cell Jerry wants to reach ($A_{P_2Q_2}$ is guaranteed to be 1). *Jerry can only move on the rubber ducks one unit up, down, left or right.*

Output

If Jerry can reach the cell he desires (P_2, Q_2) from the cell he is currently on (P_1, Q_1), print "YES", else print "NO".

```
input
4 4
1 1 1 0
0 1 1 0
0 0 1 1
1 0 0 1
3 3
0 0

output
YES
```

```
input
4 5
1 1 1 0 1
0 1 1 0 0
1 0 0 1 0
1 0 1 1 0
3 2
0 1

output
NO
```
