One Swap to Palindrome

You are given **T** tasks to perform. In each task, you are given a string **S** of length **N**. You are allowed to select any two indices **i** and **j (i!=j)** of the given string and perform exactly one swap between the characters at these indices.

If it is possible to make the new string a palindrome then print "Yes", else print "No".

Note:

A string is said to be palindrome if it reads same as its reverse form. For example: "madam" , "dad" etc.

INPUT

First line contain an integer **T** denoting total number of tasks to perform.

Each of the following **T** lines contains a string **S**.

OUTPUT

For each task, print a single line with "**Yes**" if it is possible to make palindrome otherwise print "**No**" without any quotes.

Constraints:

1≤T≤5
2≤N≤100000

String contains only lowercase English alphabets.

Sample Input
```
1
bbg
```

Sample Output
```
Yes
```

Explanation

We can swap indices pair (1,2) [0-based] , so that final string will be **"bgb"** which is a palindrome.

Note: Your code should be able to convert the sample input into the sample output. However, this is not enough to pass the challenge, because the code will be run on multiple test cases. Therefore, your code must solve this problem statement.

Question

Equal Array

You are given an array A of size N

Find the minimum non negative number X such that there exist an index j and when you can replace **Aj by Aj+X,** the sum of elements of array from index 1 **to j** and **j+1 to** N becomes equal, where **1≤j≤N−1**. Assume array to be **1-indexed**.

If there is no possible X print **−1** in separate line.

Input Format

The first line contains T, the number of test cases.
For each Test case :
The first line contains an integer N, size of the array.
The second line contains N space-separated integers, the ith of which is Ai.

Input Constraints

$1 \le T \le 105$
$2 \le N \le 105$
$0 \le Ai \le 106$
Sum of N all over testcases doesn't not exceed 106.

Output Format

For each test case, print the required answer in separate line.

Sample Input

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

Sample Output

```
3
```

Explanation

Add 3 to the 2ndindex(1-indexing).

Note: Your code should be able to convert the sample input into the sample output. However, this is not enough to pass the challenge, because the code will be run on multiple test cases. Therefore, your code must solve this problem statement.
Time Limit: 1.0 sec(s) for each input file
Memory Limit: 256 MB
Source Limit: 1024 KB
Marking Scheme: Marks are awarded if any testcase passes
Allowed Languages: C++, C++14, Clojure, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Julia, Kotlin, Lisp, Lisp (SBCL), Lua, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Swift

Question

Bob and cities

Bob is living in a city in which houses are arranged in **N**x**M** block.

The city is denoted by **N** strings having **M** characters such that '**.**' denotes house while '**#**' denotes forests.

Bob has to pay a certain amount **LCost, RCost, UCost, DCost** to move 1 step across left, right, up or down respectively.

Bob lives in a house having co-ordinates (Stx , Sty) (**1-Based Indexing)**.

You are given **Q** tasks contains an integer **X** each. In each task, you have to find number of unique houses (including his house) can be travelled using the amount **X.**

INPUT

First line contains two space separated integers **N** and **M**, denoting number of rows and columns respectively.

Next **N** lines contain a string each containing **M** characters. (**Note :-** Top left corner will denote {1,1})

Next line containd four space separated integers denoting **LCost, RCost, UCost, DCost** respectively.

Next line contains two space separated integers **Stx** and **Sty**, denoting position of Bob's house.

Next line contains an integer **Q** denoting number of tasks.

Next **Q** lines contain an integer **X** each, denoting the amount Bob have.

Output

For each task, output a single integer denoting the number of unique houses (including his house) Bob can visit using the amount X.

Constraint

$1 <= N, M <= 1000$

$1 <= Stx <= N$

$1 <= Sty <= M$

$1 <= LCost, RCost, UCost, DCost <= 10^9$

$1 <= Q <= 10^5$

$0 <= X <= 10^{18}$

Sample Input

```
3 4
..#.
#...
..#.
1 2 3 4
2 3
3
2
5
10
```

Sample Output

```
3
7
9
```

Explanation

As the starting point is {2, 3}.

In first query Bob has 3 units of money. The total number of unique houses that can be visited are (2,3), (2,2), (2,4). Therefore the answer is **3** .

Similarly we can check for other 2 queries.

Note: Your code should be able to convert the sample input into the sample output. However, this is not enough to pass the challenge, because the code will be run on multiple test cases. Therefore, your code must solve this problem statement.
Time Limit: 1.0 sec(s) for each input file
Memory Limit: 256 MB
Source Limit: 1024 KB
Marking Scheme: Marks are awarded if any testcase passes
Allowed Languages: C++, C++14, Clojure, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Julia, Kotlin, Lisp, Lisp (SBCL), Lua, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Swift