

[Quay lui-Nhánh cận]. Bài 21. Generate Word

Cho một bảng HCN cỡ N hàng, M cột, mỗi ô trên HCN chứa 1 chữ cái in hoa hoặc chữ số. Một từ có thể được xây dựng từ các chữ cái của các ô liên kề theo thứ tự, trong đó các ô liên kề nằm cạnh nhau theo chiều ngang hoặc chiều dọc. Không được sử dụng cùng một ô chữ cái nhiều lần. Bạn hãy xác định xem có thể tạo thành từ S cho trước hay không ?

A	C	A	C
2	8	B	5
1	T	E	5
A	B	C	H

```
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;

int n, m , used[1001][1001], check;
string s;
char a[1001][1001];
int dx[]={0, 0, -1, 1};
int dy[]={-1, 1, 0, 0};

void nhap(){
    cin >> n >> m;
    for(int i =1 ;i <= n ;i++){
        for(int j =1 ; j <= m ; j++) cin >> a[i][j];
    }
    cin >> s;
}

void backtrack(int i,int j, int pos){
    if(s.size() == pos){
        check = 1;
        return;
    }
    for(int k = 0;k < 4 ;k++){
        int x = i + dx[k];
        int y = j + dy[k];
        if(x >= 1 && x <= n && y >= 1 && y <= m && used[x][y]== 0 && a[x][y] == s[pos]){
            used[x][y]=1;
            backtrack(x, y, pos + 1);
            used[x][y]=0;
        }
    }
}
```

```

int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    nhap();
    for(int i = 1; i <= n ;i++){
        for(int j = 1; j <= m ;j++){
            if(a[i][j]== s[0]){
                used[i][j]=1;
                backtrack(i,j,1);
                used[i][j]=0;
            }
        }
    }
    if(check ) cout <<"YES";
    else cout <<"NO";

    return 0;
}

```

Input Format

- Dòng đầu tiên là N và M
- N dòng tiếp theo mô tả bảng HCN
- Dòng cuối cùng là từ S

Constraints

- $1 \leq N, M \leq 5$
- $1 \leq \text{len}(S) \leq 25$

Output Format

- In ra YES nếu có thể tạo được từ S, ngược lại in NO

Sample Input 0

```

4 4
ACAC
28B5
1TE5
ABCH
28TECH

```

Sample Output 0

```

YES

```