Daily Challenge **Happy Coding from necse** **SkillRack**

Print Directions - Visited Cities

The program must accept a matrix of size **R*C** representing R*C cities. A person visits the cities from the top-left position and he marks the visited cities with the integers starting from **1**. The integers in the matrix represent the cities he visited. The asterisks in the matrix represent the cities he has not visited. The program must print the directions (N-North, S-South, E-East and W-West) in which he visited the cities as the output.

Boundary Condition(s):
2 <= R, C <= 25

Input Format:
The first line contains R and C separated by a space.
The next R lines, each contains C values separated by a space.

Output Format:
The first line contains a list of characters separated by a space representing the directions in which the person visited the cities.

Example Input/Output 1:
Input:
5 5
1 2 * 8 9
* 3 6 7 10
* 4 5 * 11
* * 16 * 12
* * 15 14 13

Output:
E S S E N E N E S S S S W W N

Explanation:
Here **R = 5** and **C = 5**.
The person visits **16 cities**(1 - 16) starting from the top-left position.
The directions in which he visited the 16 cities are given below.
E S S E N E N E S S S S W W N

Example Input/Output 2:
Input:
7 5
1 * 5 6 7
2 3 4 * 8
* * * * 9
20 * * 11 10
19 * 13 12 *
18 15 14 * *
17 16 * * *

Output:
S E E N E E S S S W S W S W S W N N N

Max Execution Time Limit: 50 millisecs

Ambiance

Java (12.0)

```java
import java.util.*;
public class Hello {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int r =sc.nextInt(),c=sc.nextInt();
        Hashtable<Integer,String> ht = new Hashtable<>();
        int max = 0;

        for(int i=0;i<r;i++){
            for(int j=0;j<c;j++){

                String temp1=sc.next();
                if(temp1.equals("*")) continue;
                int temp=Integer.parseInt(temp1);
                max= Math.max(temp,max);
                if(temp>0) ht.put(temp,i+"-"+j);
            }
        }

        int prev_r=-1,prev_c=-1;
        boolean first=true;

        for(int i=1;i<=max;i++){
            String[] temp = ht.get(i).split("-");
            int curr_r=Integer.parseInt(temp[0]),curr_c=Integer
                .parseInt(temp[1]);
            if(first) {
                first=false;
                prev_r=curr_r;
                prev_c=curr_c;
                continue;
            }


            if(prev_r<curr_r){
                System.out.print("S ");
            }else if(prev_r>curr_r){
                System.out.print("N ");
            }else if(prev_c<curr_c){
                System.out.print("E ");
            }else{
                System.out.print("W ");
            }
            prev_c=curr_c;prev_r=curr_r;
        }

    }
}
```

1912067@nec

Code did not pass the execution

Input:

```
5 5
1 2 * 8 9
* 3 6 7 10
* 4 5 * 11
* * 16 * 12
* * 15 14 13
```

Expected Output:

```
E S S E N E N E S S S S W W N
```

Your Program Output:

```
S S E N E N E S S S S W W N
```

Save Run