DP-S013 (E040)

Solved Challenges 0/1

Back To Challenges List

Wildcard Pattern Matching

ID:11136 **Solved By 517 Users**

The program must accept a **text** and a **wildcard pattern** as the input. The program must print "**Matching**" if the wildcard is matched with text. Else the program must print "**Not matching**" as the output.

The wildcard pattern can include the characters '**?**' and '*****'
'**?**' – Matches any single character
'*****' – Matches any sequence of characters (including the empty sequence)

Boundary Condition(s):
1 <= Length of text <= 100
1 <= Length of wildcard pattern <= 50

Input Format:
The first line contains the text.
The second line contains the wildcard pattern.

Output Format:
The first line contains either "Matching" or "Not matching".

Example Input/Output 1:
Input:
abbbbbbbbccbbbbbed
a*b?d

Output:
Matching

Explanation:
The wildcard pattern is "a*b?d".
'*****' can be replaced by "bbbbbbbbccbbbb".
'**?**' can be replaced by "e".
Hence the output is Matching

Example Input/Output 2:
Input:
abbbbbbbbccbhd
a*b??b?d

Output:
Matching

Example Input/Output 3:
Input:
abbbbbbbbccbhd

*c??b?d

Output:
Not matching

Max Execution Time Limit: 500 millisecs

Ambiance

Java (12.0)

Reset

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

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

        String pattern = sc.next();

        boolean[][] matrix = new boolean[str.length()+1][patte
            .length()+1];

        matrix[0][0] =true;

        if(pattern.charAt(0) == '*')
        {
        matrix[0][1] = true;
        }

        for(int row=1;row<=str.length();row++)
        {
            for(int col=1;col<=pattern.length();col++)
            {
                if(str.charAt(row-1) == pattern.charAt(col-1)
                    pattern.charAt(col-1)=='?')
                {

                    if(matrix[row-1][col-1] == true)
                    matrix[row][col] = true;
                }

                else if(pattern.charAt(col-1) == '*')
                {
                    if(row==0)
                    matrix[row][col] = true;
                    if(matrix[row][col-1]==true || matrix[row
                        -1][col]==true)
                    matrix[row][col] = true;
                }
            }
        }

        if(matrix[str.length()][pattern.length()] == true)
        System.out.println("Matching");
        else
        System.out.println("Not matching");

    }
}
```

Code did not pass the execution

— ×

```
Hello.java:40: error: incomparable types: boolean and int
 if(matrix[str.length()-1][pattern.length()-1] == 1)
                                                ^
1 error
```

Save Run

□ Run with a custom test case (Input/Output)