44. Wildcard Matching

Total Accepted: **60636** Total Submissions: **340929** Difficulty: **Hard**

Implement wildcard pattern matching with support for '?' and '*'.

'?' Matches any single character.

'*' Matches any sequence of characters (including the empty sequence).

The matching should cover the <b style="box-sizing: border-box; font-weight: 700;">entire</b> input string (not partial).

The function prototype should be:

bool isMatch(const char *s, const char *p)

Some examples:

isMatch("aa","a") → false

isMatch("aa","aa") → true

isMatch("aaa","aa") → false

isMatch("aa", "*") → true

isMatch("aa", "a*") → true

isMatch("ab", "?*") → true

isMatch("aab", "c*a*b") → false

```
class Solution {
//author:zzw
public:
    bool isMatch(string s, string p) {
        int n1 = s.size();
        int n2 = p.size();
        vector<vector<bool> >res(n1+1,vector<bool>(n2+1,false));
        res[0][0]=true;
        for(int i=1;i<=n2;i++)
        {
            if(p[i-1] == '*')
                res[0][i] = res[0][i-1];
        }
        for(int i=1;i<=n1;i++)
        {
            for(int j=1;j<=n2;j++)
            {
                if(p[j-1] != '*')
                {
                    res[i][j] = res[i-1][j-1] && (s[i-1] == p[j-1] || p[j-1]=='?');
                }else{
                    res[i][j] = res[i][j-1] || res[i-1][j];
                }
            }
        }
        return res[n1][n2];
    }
};
```