

```

/*
1, If p.charAt(j) == s.charAt(i) : dp[i][j] = dp[i-1][j-1];
2, If p.charAt(j) == '.' : dp[i][j] = dp[i-1][j-1];
3, If p.charAt(j) == '*':
    here are two sub conditions:
        1 if p.charAt(j-1) != s.charAt(i) : dp[i][j] = dp[i][j-2] //in this case, a* only counts as empty
        2 if p.charAt(j-1) == s.charAt(i) or p.charAt(j-1) == '.':
            dp[i][j] = dp[i-1][j] //in this case, a* counts as multiple a (s.charAt(i) of course match with * since s.charAt(i)==p.charAt(j-1))
            or dp[i][j] = dp[i][j-1] // in this case, a* counts as single a
            or dp[i][j] = dp[i][j-2] // in this case, a* counts as empty
*/
bool isMatch(char* s, char* p) {
    bool** dp;
    int s_len;
    int p_len;
    int s_index;
    int p_index;

    s_len = strlen(s);
    p_len = strlen(p);

    dp = (bool**)calloc(p_len+1, sizeof(bool*));
    dp[0] = (bool*)calloc((p_len+1)*(s_len+1), sizeof(bool));

    dp[0][0] = true;

    for(p_index = 1; p_index <= p_len;p_index++)
    {
        if(dp[p_index] == NULL)
        {
            dp[p_index] = dp[0] + p_index * (s_len+1);
        }

        if(p[p_index-1] == '*')
        {
            dp[p_index][0] = dp[p_index-2][0];
        }
    }

    for(p_index = 1; p_index <= p_len;p_index++)
    {
        for(s_index = 1; s_index <= s_len;s_index++)
        {
            if(dp[p_index] == NULL)
            {
                dp[p_index] = dp[0] + p_index * (s_len+1);
            }

            if(p[p_index-1] == '*')
            {
                dp[p_index][s_index] = dp[p_index-2][s_index] | ( (s[s_index-1] == p[p_index-2] || p[p_index-2] == '.') && dp[p_index][s_index-1]);
            }else if( (s[s_index-1] == p[p_index-1] ) || ( p[p_index-1] == '.' ) )
            {
                dp[p_index][s_index] = dp[p_index - 1][s_index - 1] ;
            }
        }
    }

    return dp[p_len][s_len];
}

```