

字符串通配符

问题描述：在计算机中，通配符一种特殊语法，广泛应用于文件搜索、数据库、正则表达式等领域。现要求各位实现字符串通配符的算法。

要求：

实现如下 2 个通配符：

*：匹配 0 个或以上的字符（字符由英文字母和数字 0-9 组成，不区分大小写。下同）

?：匹配 1 个字符

输入：

通配符表达式；

一组字符串。

输出：

返回匹配的结果，正确输出 `true`，错误输出 `false`

本题含有多组样例输入！

输入描述:

先输入一个带有通配符的字符串，再输入一个需要匹配的字符串

输出描述:

返回匹配的结果，正确输出 `true`，错误输出 `false`

示例 1

输入

```
te?t*.*  
txt12.xls
```

输出

```
false
```

```

#include<iostream>
#include<string>
using namespace std;

bool matchstring(const char* s1,const char* s2)
{
    if(*s1=='\0'&&*s2=='\0')
        return true;
    if(*s1=='\0' || *s2=='\0')
        return false;
    if(*s1==*s2)//相同 同时后移+1
        return matchstring(s1+1,s2+1);
    else if(*s1=='?')//仅匹配一个 只能同时后移+1
        return matchstring(s1+1,s2+1);
    else if(*s1=='*')//匹配 0 个 仅通配后移+1 匹配 1 个 通配和字符串均+1 匹配多个 仅
字符串后移+1
        return matchstring(s1+1,s2) || matchstring(s1+1,s2+1) || matchstring(s1,s2+1);
    return false;
}

int main()
{
    string patten;
    string str;
    while(cin>>patten>>str)
    {
        if(matchstring(patten.c_str(),str.c_str()))
        {
            cout<<"true"<<endl;
        }
        else
            cout<<"false"<<endl;
    }
    return 0;
}

```