# 20. 有效的括号（Valid Parentheses）

## 题目：

给定一个只包括 '('，')'，'{'，'}'，'['，']' 的字符串，判断字符串是否有效。

有效字符串需满足：

1. 左括号必须用相同类型的右括号闭合。

左括号必须以正确的顺序闭合

注意空字符串可被认为是有效字符串。

**示例 1:**

输入: "()"  
输出: true

**示例 2:**

输入: "()[]{}"  
输出: true

**示例 3:**

输入: "(]"  
输出: false

**示例 4:**

输入: "([)]"  
输出: false

**示例 5:**

输入: "{[]}"  
输出: true

## 解答：

* 利用栈

|  |
| --- |
| public static bool IsValid(string s)  {  //字符串的长度不是偶数的，都不是  if (s.Length % 2 != 0)  {  return false;  }  //本题利用的是栈的知识  //主体思路就是把左边的括号放进栈中，放下一个字符时如果还是左边括号继续入栈  //如果是右边的括号判断是否和栈最上面的括号是一对，如果是一对就出栈，相当于消消乐  //如果不是一对返回错误  Stack<char> st = new Stack<char>();  for (int i = 0; i < s.Length; i++)  {  char c = s[i];  if (c == '(' || c == '[' || c == '{')  {  st.Push(c);  }  else if (c == ')' && st.Count != 0 && st.Peek() == '(')  {  st.Pop();  }  else if (c == ']' && st.Count != 0 && st.Peek() == '[')  {  st.Pop();  }  else if (c == '}' && st.Count != 0 && st.Peek() == '{')  {  st.Pop();  }  else  {  return false;  }  }  return st.Count == 0;  } |