

32. Longest Valid Parentheses

Total Accepted: **61821** Total Submissions: **275494** Difficulty: **Hard**

Given a string containing just the characters '(' and ')', find the length of the longest valid (well-formed) parentheses substring.

For "()", the longest valid parentheses substring is "()", which has length = 2.

Another example is "()()()", where the longest valid parentheses substring is "()()()", which has length = 4.

[Subscribe](#) to see which companies asked this question

```
//swift
//author:zzw
class Solution {
    func longestValidParentheses(s: String) -> Int {
        var size:Int = s.characters.count;
        var s_arr:[Character] = [Character]();
        for c in s.characters
        {
            s_arr.append(c);
        }
        // init f
        var f:[Int]=[Int](count:size,repeatedValue:0);
        var ret:Int = 0;
        for(var i=size-2;i>=0;i--)
        {

            var match:Int = i+f[i+1]+1;
            //case ((...))
            if(match<size){
                if (s_arr[i]=="(" && s_arr[match]=="){"){

                    // add new mather "()",its length = 2
                    f[i]=f[i+1]+2;
                    //((...))()
                    if match+1<size
                    {
                        f[i]+=f[match+1];
                    }
                }
            }
        }
        return ret;
    }
}
```

```
        }
    }
}
ret = max(ret,f[i]);
}
return ret;
}
}
```