Longest valid prarentheses

public int longestValidParentheses(String s) {

Stack<Integer> stack = new Stack<Integer>();

int len[] = new int[s.length()];

int max = 0;

for(int i = 0; i < s.length(); i++) {

if( s.charAt(i) == '(' )

stack.push(i);

else if( !stack.isEmpty() ) {

len[i] = i - stack.peek() + 1; // Currently,len[i] only reprensents the distance between s[i] and its pair '('

len[i] += stack.peek() > 0 ? len[stack.peek() - 1] : 0; // plus the length of longest valid substring which ends at the previous element of pair '('

max = Math.max(len[i], max);

stack.pop();

}

}

return max;

}