

022. Generate Parentheses

021 Merge Two Sorted Lists

- Backtracking+string
-

Description

Given n pairs of parentheses, write a function to generate all combinations of well-formed parentheses.

For example, given $n = 3$, a solution set is:

```
[
  "((()))",
  "(()())",
  "(())()",
  "()()()",
  "()(())"
]
```

1. Thought line

2. Backtracking+string

```
1 class Solution {
2 private:
3     void generateParenthesis_fct(int lfCount, int rtCount, int n, string str, vector<string>& result){
4         if (lfCount<rtCount || lfCount>n || rtCount>n || n<=0) return;
5         if (lfCount==n && rtCount==n) {
6             result.push_back(str);
7             return;
8         }
9         // insert "("
10        if (lfCount<n)
11            generateParenthesis_fct(lfCount+1, rtCount, n, str+"(", result);
12        // insert ")"
13        if(lfCount>rtCount)
14            generateParenthesis_fct(lfCount, rtCount+1, n, str+")", result);
15    }
16 public:
17     vector<string> generateParenthesis(int n) {
18         vector<string> result;
19         generateParenthesis_fct(0, 0, n, "", result);
20         return result;
21     }
22 };
```