周总结：

1. BFS实现leetcode 22题括号生成:（之前用递归和深度优先实现）

class Solution {

class Node{

private String val;

private int left;

private int right;

public Node(String val , int left, int right){

this.val = val;

this.left = left;

this.right = right;

}

}

List<String> listr = new ArrayList<>();

public List<String> generateParenthesis(int n) {

Queue<Node> que = new LinkedList<>();

Node node1 = new Node("(",1,0);

que.offer(node1);

while(!que.isEmpty()){

Node node = que.poll();

if(node.left < n){

que.offer(new Node(node.val+"(", node.left+1, node.right));

}

if(node.right < node.left){

Node newNode = new Node(node.val+")", node.left, node.right+1);

if(newNode.right == n){

listr.add(newNode.val);

}else{

que.offer(newNode);

}

}

}

return listr ;

}

}

1. 广度优先:目前看到的都是队列

深度优先：递归，回溯。

贪心算法：要求局部最优一定能达到全局最优