

week 03

1.泛型递归，树的递归

C++模板

```
1 //c++ 模板
2 void recursion (int level, int param) {
3     //recursion terminator(终止条件)
4     if(level>MAX_LEVEL)
5         // process result
6         return;
7     // process current logic (当前层逻辑)
8     process(level, param);
9
10    //drill down (向探)
11    recursion(level+1, param);
12
13    //reverse the current level status if needed (如果需要reverse)
14 }
```

注意事项：

- 不要人肉递归
- 寻找最近最简方法 和 重复子问题
- 数学归纳思维

2.分治，回溯

分治，回溯属于递归的一种， 所以代码模板可以从递归中进行修改， 一样包含四个部分
例题：

C++模板

```
1 int divide_conquer(Problem *problem, int params){
2     //recursion terminator
3     if(problem == nullptr){
4         process_result
5         return return_result;
6     }
7
8     //process current problem
```

```
9   subproblems=split_problem(problem, data)
10  subresult1=divide_conquer(subproblem[0], p1)
11  subresult2=divide_conquer(subproblem[1], p1)
12  subresult3=divide_conquer(subproblem[2], p1)
13  ...
14
15  // merge
16  result=process_result(subresult1, subresult2, subresult3)
17  //revert the current level status
18
19  return 0;
20 }
```