

P2404 自然数的拆分问题

题目描述

任何一个大于1的自然数 n ，总可以拆分成若干个小于 n 的自然数之和。现在给你一个自然数 n ，要求你求出 n 的拆分成一些数字的和。每个拆分后的序列中的数字从小到大排序。然后你需要输出这些序列，其中字典序小的序列需要优先输出。

输入格式

输入：待拆分的自然数 n 。

输出格式

输出：若干数的加法式子。

输入样例

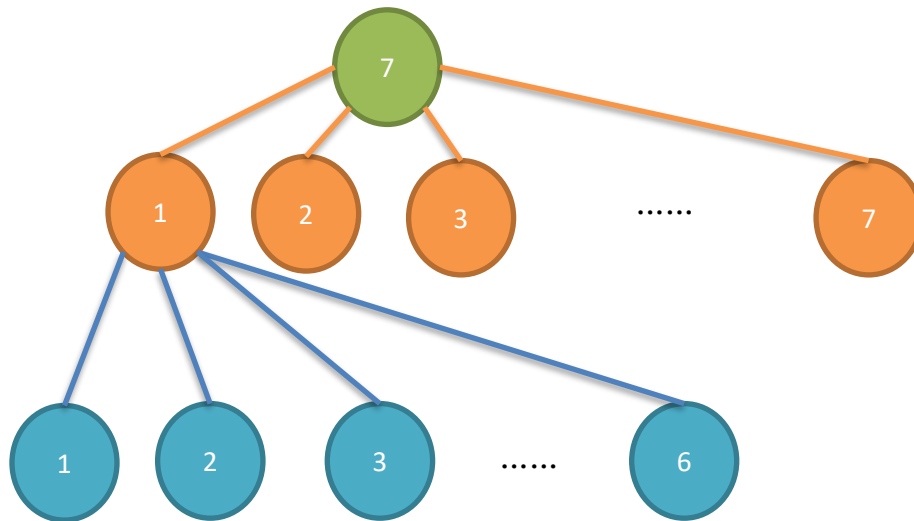
7

输出样例

1+1+1+1+1+1+1
1+1+1+1+1+2
1+1+1+1+3
1+1+1+2+2
1+1+1+4
1+1+2+3
1+1+5
1+2+2+2
1+2+4
1+3+3
1+6
2+2+3
2+5
3+4

解析

典型的深搜问题，我们需要开始构建搜索树。



我们尝试从1开始进行放置，然后逐层的拆分剩余的数字，第一层是7，第二层是6(7-1)，直到剩余的数量为0的时候，便找到了一个完整的拆分过程。

需要注意的是，作为起点的数字一定要大于前一个数字，否则就会出现2+1这样明显是重复的结果。

另外，拆分出来的加数一定要小于最初的目标值，否则就会出现 7=7这样的情况。

编码

```
#include <bits/stdc++.h>

using namespace std;

//从1开始进行拆分
int box[100] = {1};
//目标数字
int n;

//深度优先搜索
void dfs(int num, int step) {
    //达到了需要组合的数字，打印完整的运算组合
    if (num == 0) {
        //先打印独立的首个数字
        printf("%d", box[1]);
        //循环打印后面的数字
        for (int i = 2; i <= step - 1; i++) {
            printf("+%d", box[i]);
        }
        printf("\n");
        return;
    }
}
```

```

//以前一个盒子的数字为起点，在不超过当前目标数字num的范围内进行拆解
for (int i = box[step - 1]; i <= num; i++) {
    //当前的加数不能等于目标数字num，否则就会出现7=7这样的情形；
    if (i < n) {
        //用当前的目标数字num减去合法的加数
        num -= i;
        //将合法的加入放入盒子中
        box[step] = i;
        //向下一个盒子进行放置
        dfs(num, step + 1);
        //回溯：加上拆分的数，以便产生所有可能的拆分
        num += i;
    }
}

int main(int argc, char **argv) {
    cin >> n;
    dfs(n, 1);
    return 0;
}

```

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。

