

2020/9/26

许茂泽

1. 代码实践

(1) 最长无重复字符串

```
1  #include<stdio.h>
2  #include<iostream>
3  #include<string>
4  #include<map>
5  using namespace std;
6  /*
7   滑动窗口，设置左右两个边界，右边界不断向右移动，如果检测到窗口内已存在一个相同的字符，
8   则左边界跳到前一个相同字符的下一个位置，移动过程中记录最大长度。
9
10  用哈希表记录字符所在位置，使用的是自己编写的数组，加快速度
11  */
12
13  int main(){
14      int n;
15      string s;
16      cin>>n;
17      cin>>s;
18      if(s.size()==0){
19          return 0;
20      }
21      if(s.size()==1){
22          return 1;
23      }
24      int chr[128];                //Hash表记录字符出现位置
25      for(int i=0;i<128;i++){
26          chr[i]=-1;
27      }
28      int i,j,max=1;
29      chr[s[0]]=0;
30      for(i=0,j=1;j<n;j++){
31          if(chr[s[j]]!=-1){
32              chr[s[j]]=j;
33          }
34          else{
35              //已有相同的字符且在窗口内
36              if(i<=chr[s[j]]){
37                  i=chr[s[j]]+1;
38              }
39              chr[s[j]]=j;
40          }
41      }
42      int tmp=j-i+1;
43      if(tmp>max){
44          max=tmp;
45      }
```

```

46         //提前退出
47         if(n-i<=max){
48             break;
49         }
50     }
51
52     cout<<max;
53     return 0;
54 }

```

(2) LeetCode113

给定一个二叉树和一个目标和，找到所有从根节点到叶子节点路径总和等于给定目标和的路径。

是回溯算法的常见题

```

1  class Solution {
2
3  public:
4      vector<vector<int>> ans;
5      vector<int> tmp;
6
7      void dfs(TreeNode* root, int sum){
8          if(!root){
9              return;
10         }
11         sum-=root->val;
12         tmp.push_back(root->val);
13         //找到一个解
14         if(sum==0&&!root->left&&!root->right){
15             ans.push_back(tmp);
16         }
17         dfs(root->left,sum);
18         dfs(root->right,sum);
19         //回溯
20         sum+=root->val;
21         tmp.pop_back();
22     }
23
24     vector<vector<int>> pathSum(TreeNode* root, int sum) {
25         dfs(root,sum);
26         return ans;
27     }
28 };

```

2. 计算机基础知识

- 进程控制一般是由OS的内核中的原语来实现的。原语(Primitive)是由若干条指令组成的，用于完成一定功能的一个过程。它与一般过程的区别在于：它们是“原子操作(Action Operation)”。所谓原子操作，是指一个操作中的所有动作要么全做，要么全不做。换言之，它是一个不可分割的基本单位，因此，在执行过程中不允许被中断。原子操作在管态下执行，常驻内存。
- 原语的作用是为了实现进程的通信和控制，系统对进程的控制如不使用原语，就会造成其状态的不确定性，从而达不到进程控制的目的。
- 进程控制原语包括：**进程创建，进程阻塞，唤醒进程和进程终止**四个原语。

3. 开源特训营

提交每日学习笔记

合并PR

给项目提了一个Issue