

《数据结构与算法》第二周作业反馈

助教-陈孙一硕

先上我写的这几道题的代码：

1. 冰雹猜想

这道题的测试点 example1 有一些问题. 题目的描述中, 输入是满足 $n > 1$ 的. 但是有同学的代码无法通过这个测试点. 我在这份代码的基础上加了一个 $n = 1$ 情况的特判就可以通过, 因此可以得出是测试点存在一些问题, example1 测试的就是 $n = 1$ 的情况, 而这是不合法的输入.

```
#include <bits/stdc++.h>
using namespace std;
#define int long long
int n, cnt = 1;

signed main() {
    ios::sync_with_stdio(false);
    cin.tie(nullptr);
    cin >> n;
    while (n != 1) {
        cout << n << ' ';
        cnt++;
        if (n % 2 == 0) {
            n /= 2;
        } else {
            n = n * 3 + 1;
        }
    }
    cout << 1 << endl;
    cout << "count = " << cnt << endl;
    return 0;
}
```

2. 打印图形

应该是属于《程序设计基础》课程中的内容了吧，不多说了，上代码。

```
#include <bits/stdc++.h>
using namespace std;
#define int long long
int n;

signed main() {
    ios::sync_with_stdio(false);
    cin.tie(nullptr);
    cin >> n;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            cout << '*';
        }
        for (int j = 1; j <= 2 * n - 2 * i; j++) {
            cout << ' ';
        }
        for (int j = 1; j <= i; j++) {
            cout << '*';
        }
        cout << endl;
    }
    return 0;
}
```

3. 最大子段和

经典板子题。这里我使用的是基于动态规划的做法， $dp[i]$ 表示的是以 $a[i]$ 为结尾的最大子段和。转移的想法就是，要么延续前一个子段，要么以当前位置为首作为一个新的子段。

当然还有一种经典做法，就是基于前缀和的。预处理一个前缀和数组，再用一个数组去记录，对于每个位置 i ，在此之前出现过的前缀和的最小值，也是从前往后扫一遍，把两个数组每个位置上的值作差，取 \max 即可。在线性表这个知识点中，有关“前缀”的应用是很广泛且经典的，大家可以好好研究，后面要考。

```
#include<bits/stdc++.h>
using namespace std;
#define int long long
```

```
const int N=2e6+101;
int a[N],dp[N];

signed main(){
    int n; cin>>n;
    for (int i=1;i<=n;i++) cin>>a[i];
    int ans=0;
    for (int i=1;i<=n;i++){
        dp[i]=max(dp[i-1]+a[i],a[i]);
        ans=max(ans,dp[i]);
    }
    cout<<ans<<endl;
    return 0;
}
```