首先有一个公式:

设n[h]是高度为h的平衡二叉树的最小节点数,则n[h]=n[h-1]+n[h-2]

给定节点数为n的AVL树的最大高度为O(log2 n)

注意:

如果值是10^9 ,即1亿, 计算斐波那契到第45项就是11亿了,所以如果是int存储的话只用算到44项

找到一个不大于n的a[i] ,则最大高度就是i,  且 a[i-1]<= n <a[i]

#include <iostream>

#include <cstdio>

#include <cstring>

#include <cmath>

#include <algorithm>

using namespace std;

int main()

{

//freopen("input.txt","r",stdin);

int a[50];

int n;

a[0]=1;a[1]=2;

for(int i=2;i<=50;i++)

a[i]=a[i-1]+a[i-2]+1;

while(cin>>n && n)

{

int i=0;

while(a[i]<=n)i++;

cout<<--i<<endl;

}

return 0;

}