**题目描述 Description**

    某国为了防御敌国的导弹袭击，发展出一种导弹拦截系统。但是这种导弹拦截系统有一个缺陷：虽然它的第一发炮弹能够到达任意的高度，但是以后每一发炮弹都不能高于前一发的高度。某天，雷达捕捉到敌国的导弹来袭。由于该系统还在试用阶段，所以只有一套系统，因此有可能不能拦截所有的导弹。

**输入描述 Input Description**

输入导弹依次飞来的高度（雷达给出的高度数据是不大于30000的正整数）

**输出描述 Output Description**

输出这套系统最多能拦截多少导弹，如果要拦截所有导弹最少要配备多少套这种导弹拦截系统。

**样例输入 Sample Input**

389 207 155 300 299 170 158 65

**样例输出 Sample Output**

6

2

**数据范围及提示 Data Size & Hint**

导弹的高度<=30000，导弹个数<=20

第一问是一个最长不上升子序列

被第二问卡住了

其实他就是个最长上升子序列

因为能拦截的都是不上升的序列，所以也就是说，每多一个更高的，就需要一套新的，所以找最长上升子序列

#include<bits/stdc++.h>

using namespace std;

int n(0);

int f[100],a[100];

int main()

{

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

int ans(0),x;

while(cin>>x)

{

a[++n]=x;

}

f[1]=1;

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

{

for(int j=1;j<i;j++)

if(a[j]>=a[i])

f[i]=max(f[i],f[j]);

f[i]++;

ans=max(ans,f[i]);

}

//for(int i=1;i<=n;i++)cout<<f[i]<<' ';cout<<endl;

cout<<ans<<endl;

ans=0;

memset(f,0,sizeof(f));

f[1]=1;

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

{

for(int j=1;j<i;j++)

if(a[j]<a[i])

f[i]=max(f[i],f[j]);

f[i]++;

ans=max(ans,f[i]);

}

cout<<ans<<endl;

return 0;

}