题意：道路上有n个显示器，让你选择三个显示器，每个显示器都有显示字体的大小，和费用，你需要租三个显示器，每个显示器显示字体的大小，逐渐递增，然后让他们花费最小。输入n,然后下面一行每个显示器的能够显示字体的大小，然后n个显示器的花费。

题解：暴力 或者dp 暴力就先选择一个显示器，找前面尺寸比它小的，后面尺寸比他大的一个显示器。然后记录费用，取最小值。dp的话，dp学的不好，看别人博客：因为只用选出三个物品，dp[i,j]代表第个物品作为选出的三个物品中的第

个物品的最小花费，显然有：dp[i,j]=min(dp[i,j−1]+c[i]，dp[i][j])，其中：j<ij<i且s[j]<s[i]s[j]<s[i]。

怎么就想不到暴力呢

#include<stdio.h>

#include<iostream>

#include<cstring>

#include<set>

#include<map>

#include<cmath>

#define INF 1000000007

typedef long long ll;

using namespace std;

int a[3010];

int f[3010];

int c[3010];

short b[3010];

int main()

{

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

int n;

cin>>n;

for(int i=0;i<n;i++)

cin>>a[i];

for(int i=0;i<n;i++)

cin>>c[i];

int ans(INF);

for(int i=0;i<n;i++)

{

int min1(INF),min2(INF);

for(int j=0;j<i;j++)

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

min1=min(min1,c[j]);

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

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

min2=min(min2,c[j]);

ans=min(ans,min1+min2+c[i]);

}

if(ans==INF)

cout<<"-1\n";

else cout<<ans<<"\n";

return 0;

}

另一种姿势的暴力：

对于pre[j]来说，i就相当于j之前的

对于til[i]来说，j就相当于i之后的

这样两个数组就把i之前和i之后的存了，但i存了两边，需要减去

Dp解法

#include<stdio.h>

#include<iostream>

#include<cstring>

#include<set>

#include<map>

#include<cmath>

#define INF 0x3f3f3f3f

typedef long long ll;

using namespace std;

int a[3010];

int c[3010];

int dp[3010][5];

int main()

{

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

int n;

cin>>n;

for(int i=0;i<n;i++)

cin>>a[i];

for(int i=0;i<n;i++)

cin>>c[i];

memset(dp,INF,sizeof(dp));

for(int i=0;i<n;i++)

{

dp[i][1]=c[i];

for(int j=2;j<=3;j++)

{

for(int k=0;k<i;k++)

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

dp[i][j]=min(dp[i][j],dp[k][j-1]+c[i]);

}

}

int ans(INF);

for(int i=0;i<n;i++)

ans=min(ans,dp[i][3]);

if(ans==INF)cout<<"-1\n";

else cout<<ans<<"\n";

return 0;

}