

任务调度

2014年4月25日

507

0

来源：<http://nnsznoi.openjudge.cn/greedy/0014/>

Description

一个单位时间任务是一个作业,如要在计算机上运行一个程序,它恰覆盖一个单位的运行时间。给定一个单位时间任务的集合 S ,对 S 的一个调度即 S 的一个排列,其中规定了这些任务的执行顺序。该调度中的第一个任务开始于时间0,结束于时1;第二个任务开始于时间1,结束于时间2;.....。单处理器上具有期限和罚款的单位时间任务调度问题的输入如下:

- 1.包含 n 个单位时间任务的集合 $S=\{1,2,\dots,n\}$;
 2. n 个取整的期限 $d_1,\dots,d_n,(1\leq d_i\leq n)$,任务 i 要求在 d_i 前完成;
 3. n 个非负的权(或罚款) w_1,\dots,w_n 。如果任务 i 没在时间 d_i 之前结束,则导致罚款 w_i ;
- 要求找出 S 的一个调度,使之最小化总的罚款。

Input

输入第一行为 $N\leq 1000$,后面 N 行每行两个数,即为对应的 d_i 与 w_i 。($w_i\leq 1000$)

Output

输出最小总罚款

Sample Input

```
5
1 4
2 4
2 8
3 8
5 2
```

C++

Sample Output

```
4
```

C++

跟任务安排一样还不用堆,因为有人问。。

```
#include<iostream>
#include<cstdio>
#include<algorithm>
#define inf 0x7fffffff
using namespace std;
struct data{int d,w;}a[1001];
int n,ans,b[1001],now;
bool cmp(data a,data b)
{return a.d<b.d;}
int findmn()
{
    int mn=inf,tmp;
```

C++

```
13     for(int i=1;i<=now;i++)
14         if(b[i]<mn){mn=b[i];tmp=i;}
15     return tmp;
16 }
17 int main()
18 {
19     scanf("%d",&n);
20     for(int i=1;i<=n;i++)
21         scanf("%d%d",&a[i].d,&a[i].w);
22     sort(a+1,a+n+1,cmp);
23     for(int i=1;i<=n;i++)
24     {
25         if(now<a[i].d){now++;b[now]=a[i].w;}
26         else
27         {
28             int t=findmn();
29             if(b[t]<a[i].w){ans+=b[t];b[t]=a[i].w;}
30             else ans+=a[i].w;
31         }
32     }
33     printf("%d",ans);
34     return 0;
35 }
```