**★★★★☆**

**題組：Problem Set Archive** **with Online Judge**

**題號：11400 : Lighting System Design**

**解題者：蕭廷瑞**

**解題日期：2015年6月18日**

**題意：**  
你被要求要用數種燈泡來規劃會議廳的照明系統，但總成本越低越好。系統一開始會給定一個正整數n(1<=n<=1000)，代表接下來有n種不同的燈泡及其對應的電壓V、電源供應器成本K、每個燈泡的成本C和燈泡數量L。每一種燈泡的資訊都在同一列，也就是會有n列，每一列有4個正整數，由前至後分別為:V(1<=V<=132000)、K(1<=K<=1000)、C(1<=C<=10)、L(1<=L<=100)。而電壓大的燈泡可以取代電壓小的燈泡。題目所求為總成本最低的規劃方式的金額是多少。

**題意範例：**

**3**

**100 500 10 20**

**120 600 8 16**

**220 400 7 18 🡺 778**

**0**

**解法：**以Greedy method及Dynamic Programming(DP)的方式來解題。由於大電壓的燈泡可以取代小電壓的燈泡，因此每一種規畫方式可以用「累計至該電壓的燈泡數量」來做取代並計算成本。取代時，一定是全部替換掉某一種電壓的所有燈泡，因為只換一部分的話，還需要兩個電源供應器，這樣絕對不會是最佳解。再來還有一個要注意的地方，假設電壓已經是由小到大排列好的，最佳解的替換情形一定是連續段的替換，不會有中間跳過某一項電壓的燈泡不換的情形。因為假設中間有某一種電壓的燈泡不替換，而它的前一種卻能夠被替換(表示成本可以更少)，那麼表示那一項不能被替換的電壓也不能替換掉它的前一項，然而，這用數學式可以證明(請見下一頁)。  
第1種電壓的燈泡到第n種電壓的燈泡全部取代為第n種燈泡、扣除第1種後第2種到第n種取代為第n種燈泡、……、扣除第n-2種後第n-1種到第n種取代為第n種燈泡、不取代任何一種。每一次取代都與前一次結果做比較，取小的並紀錄。可以用陣列來實現DP並解題。

**解法範例：**

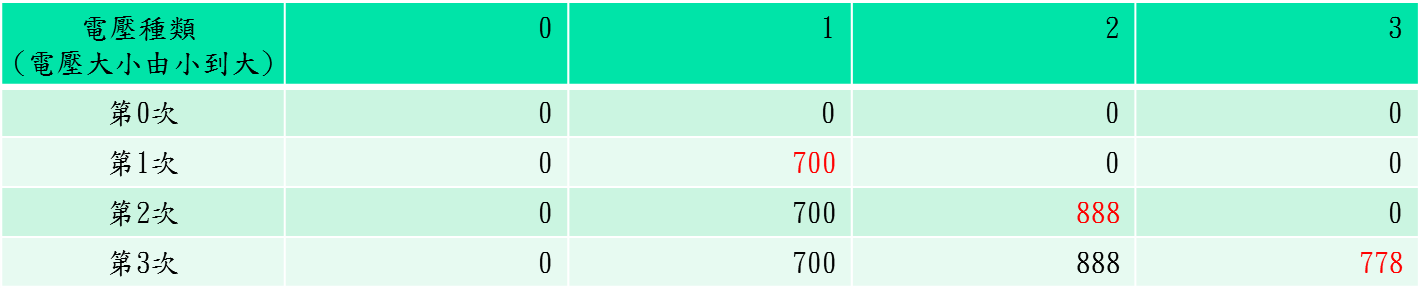
**3**

**100 500 10 20**

**120 600 8 16**

**220 400 7 18 🡺 778**

**0**



**討論：**無

**程式碼 :**

**#include <stdio.h>**

**#include <algorithm>**

**using namespace std;**

**struct light**

**{**

**int v;**

**int k;**

**int c;**

**int l;**

**};**

**bool smaller(light a,light b)**

**{**

**return a.v<b.v;**

**}**

**int main()**

**{**

**int n,i,j;**

**light la[1001];**

**int sum[1010],dp[1010];**

**while(1)**

**{**

**scanf("%d",&n);**

**if(n==0)**

**break;**

**for(i=0; i<1010; i++)**

**{**

**sum[i]=0;**

**dp[i]=0;**

**}**

**for(i=1; i<=n; i++)**

**scanf("%d%d%d%d",&la[i].v,&la[i].k,&la[i].c,&la[i].l);**

**sort(la+1,la+n+1,smaller);**

**for(i=1; i<=n; i++)**

**sum[i]=sum[i-1]+la[i].l;**

**for(i=1;i<=n;i++)**

**{**

**dp[i]=0x3f3f3f3f;**

**for(j=i; j>=1; j--)**

**dp[i]=min(dp[i],dp[j-1]+(sum[i]-sum[j-1])\*la[i].c+la[i].k);**

**}**

**printf("%d\n",dp[n]);**

**}**

**return 0;**

**}**