

【bzoj1588】[HNOI2002]营业额统计

2014年3月28日

1,248

1

Description

营业额统计 Tiger最近被公司升任为营业部经理，他上任后接受公司交给的第一项任务便是统计并分析公司成立以来的营业情况。Tiger拿出了公司的账本，账本上记录了公司成立以来每天的营业额。分析营业情况是一项相当复杂的工作。由于节假日，大减价或者是其他情况的时候，营业额会出现一定的波动，当然一定的波动是能够接受的，但是在某些时候营业额突变得很高或是很低，这就证明公司此时的经营状况出现了问题。经济管理学上定义了一种最小波动值来衡量这种情况：该天的最小波动值 当最小波动值越大时，就说明营业情况越不稳定。而分析整个公司的从成立到现在营业情况是否稳定，只需要把每一天的最小波动值加起来就可以了。你的任务就是编写一个程序帮助Tiger来计算这一个值。第一天的最小波动值为第一天的营业额。 □ 输入输出要求

Input

第一行为正整数，表示该公司从成立一直到现在的天数，接下来的n行每行有一个正整数，表示第i天公司的营业额。

Output

输出文件仅有一个正整数，即Sigma(每天最小的波动值)。结果小于 2^{31} 。

Sample Input

```
6
5
1
2
5
4
6
```

Sample Output

```
12
```

HINT

结果说明： $5+|1-5|+|2-1|+|5-5|+|4-5|+|6-5|=5+4+1+0+1+1=12$

代码

双向链表（本题参考论文《基本数据结构在信息学竞赛中的应用》）

```

1  #include<iostream>
2  #include<cstdio>
3  #include<algorithm>
4  using namespace std;
5  struct data{
6      int p,v;
7      int left,right;
8  }a[40001];
9  inline bool cmp(data a,data b){return a.v<b.v;}
10 int n,rank[40001]; //排序后每个点的位置
11 int main()
12 {
13     scanf("%d",&n);
14     for(int i=1;i<=n;i++)
15     {
16         scanf("%d",&a[i].v);
17         a[i].p=i;
18     }
19     sort(a+1,a+n+1,cmp);
20     for(int i=1;i<=n;i++) rank[a[i].p]=i;
21     for(int i=1;i<=n;i++)
22     {
23         a[i].left=i-1;
24         a[i].right=i+1;
25     }
26     a[n].right=0;
27     int ans=a[rank[1]].v;
28     for(int i=n;i>1;i--)
29     {
30         int x=rank[i];
31         if(a[x].left&&a[x].right)
32         {
33             ans+=min(a[x].v-a[a[x].left].v,a[a[x].right].v-a[x].v);
34             a[a[x].left].right=a[x].right;
35             a[a[x].right].left=a[x].left;
36         }
37         else if(!a[x].left)
38         {
39             ans+=a[a[x].right].v-a[x].v;
40             a[a[x].right].left=0;
41         }
42         else
43         {
44             ans+=a[x].v-a[a[x].left].v;
45             a[a[x].left].right=0;
46         }
47     }
48     printf("%d",ans);
49     return 0;
50 }

```

splay

```

1  #include<iostream>
2  #include<cstdio>
3  #define inf 1000000000

```

```

4 using namespace std;
5 int ans,n,t1,t2,rt,size;
6 int tr[50001][2],fa[50001],num[50001];
7 void rotate(int x,int &k)
8 {
9     int y=fa[x],z=fa[y],l,r;
10    if(tr[y][0]==x)l=0;else l=1;r=l^1;
11    if(y==k)k=x;
12    else{if(tr[z][0]==y)tr[z][0]=x;else tr[z][1]=x;}
13    fa[x]=z;fa[y]=x;fa[tr[x][r]]=y;
14    tr[y][l]=tr[x][r];tr[x][r]=y;
15 }
16 void splay(int x,int &k)
17 {
18     int y,z;
19     while(x!=k)
20     {
21         y=fa[x],z=fa[y];
22         if(y!=k)
23         {
24             if((tr[y][0]==x)^(tr[z][0]==y))rotate(x,k);
25             else rotate(y,k);
26         }
27         rotate(x,k);
28     }
29 }
30 }
31 void ins(int &k,int x,int last)
32 {
33     if(k==0){size++;k=size;num[k]=x;fa[k]=last;splay(k,rt);return;}
34     if(x<num[k])ins(tr[k][0],x,k);
35     else ins(tr[k][1],x,k);
36 }
37 void ask_before(int k,int x)
38 {
39     if(k==0)return;
40     if(num[k]<=x){t1=num[k];ask_before(tr[k][1],x);}
41     else ask_before(tr[k][0],x);
42 }
43 void ask_after(int k,int x)
44 {
45     if(k==0)return;
46     if(num[k]>=x){t2=num[k];ask_after(tr[k][0],x);}
47     else ask_after(tr[k][1],x);
48 }
49 int main()
50 {
51     scanf("%d",&n);
52     for(int i=1;i<=n;i++)
53     {
54         int x;if(scanf("%d",&x)==EOF)x=0;
55         t1=-inf;t2=inf;
56         ask_before(rt,x);
57         ask_after(rt,x);
58         if(i!=1)ans+=min(x-t1,t2-x);
59         else ans+=x;
60         ins(rt,x,0);
61     }
62     printf("%d",ans);
63     return 0;
64 }

```

平衡树

```

1 #include<iostream>
2 #include<cstdio>
3 #include<cstring>
4 #include<ctime>
5 #include<cstdlib>
6 #define inf 1000000000
7 using namespace std;
8 int t1,t2,n,sz,rt,ans;
9 int l[50001],r[50001],v[50001],rnd[50001],s[50001];
10 void update(int k){s[k]=s[l[k]]+s[r[k]]+1;}
11 void rturn(int &k)
12 {int t=l[k];l[k]=r[t];r[t]=k;s[t]=s[k];update(k);k=t;}
13 void lturn(int &k)
14 {int t=r[k];r[k]=l[t];l[t]=k;s[t]=s[k];update(k);k=t;}
15 void insert(int &k,int x)
16 {
17     if(!k){k=++sz;v[k]=x;s[k]=1;rnd[k]=rand();return;}
18     s[k]++;
19     if(x<v[k]){insert(l[k],x);if(rnd[l[k]]<rnd[k])rturn(k);}
20     else {insert(r[k],x);if(rnd[r[k]]<rnd[k])lturn(k);}
21 }
22 void ask_before(int k,int x)
23 {
24     if(!k)return;
25     if(x>v[k]){t1=v[k];ask_before(r[k],x);}
26     else ask_before(l[k],x);
27 }
28 void ask_after(int k,int x)
29 {
30     if(!k)return;
31     if(x<=v[k]){t2=v[k];ask_after(l[k],x);}
32     else ask_after(r[k],x);
33 }
34 int main()
35 {
36     //srand(time(0));
37     scanf("%d",&n);
38     for(int i=1;i<=n;i++)
39     {
40         int x;if(scanf("%d",&x)==EOF)x=0;
41         t1=-inf;t2=inf;
42         ask_before(rt,x);
43         ask_after(rt,x);
44         if(i!=1)ans+=min(x-t1,t2-x);
45         else ans+=x;
46         insert(rt,x);
47     }
48     printf("%d",ans);
49     return 0;
50 }

```

在bzoj上用srand(time(0))会RE