

暨南大学本科实验报告专用纸

课程名称 ACM 程序设计实验 成绩评定
实验项目名称 贪心算法 指导教师 张震
实验项目编号 0806017904 实验项目类型 综合性 实验地点 116
学生姓名 阮炜霖 学号 2020101603
学院 信息科学技术学院 系 计算机科学系 专业 网络工程
实验时间 2022 年 6 月 1 日 下午~6 月 1 日 下午 温度 °C 湿度

(一) 实验目的

- 1) 掌握贪心算法原理。
- 2) 学会如何使用贪心算法解决实际问题。

(二) 实验内容和要求

实验内容:

- 1.HDU 2037
- 2.HDU 1052
- 3.LibreOJ-2123

实验要求:

完成所有题目的在线测评。

(三) 主要仪器设备

仪器: 计算机

实验环境: Windows 2000 或 Windows XP + Visual C++6.0 或 Turbo C 2.0

(四) 源程序

HDU-2037 今年暑假不 AC

```

1. //pragma GCC optimize("Ofast", "inline", "-ffast-math")
2. //pragma GCC target("avx,sse2,sse3,sse4,mmx")
3. #include<bits/stdc++.h>
4. #define inf 0x3f3f3f3f
5. #define int long long
6. using namespace std;
7. const int N=207;
8. const int mod=1e9+7;
9.
10. //int read(){ int x=0,f=1;char ch=getchar();while(ch<'0' || ch>'9'){i
    f(ch=='-') f=f*-1;ch=getchar();}while(ch>='0'&&ch<='9'){x=x*10+ch-'0'
    ;ch=getchar();}return x*f;}
11. //void write(int x){if(x>9) write(x/10);putchar(x%10+'0');}
12. struct node{
13.     int l,r;
14. }s[N];
15.
16. bool cmp(node A,node B){
17.     return A.r<B.r;
18. }
19.
20. int n;
21.
22. void Solve(){
23.     for(int i=1;i<=n;++i){
24.         cin>>s[i].l>>s[i].r;
25.     }
26.     sort(s+1,s+1+n,cmp);
27.     int now=0,ans=0;
28.     for(int i=1;i<=n;++i){
29.         if(s[i].l>=now) ++ans,now=s[i].r;
30.     }
31.     cout<<ans<<"\n";
32. }
33.
34. signed main(){
35. // ios::sync_with_stdio(0);
36. // cin.tie(0);cout.tie(0);
37. // freopen("in.cpp","r",stdin);
38. // freopen("out.cpp","w",stdout);
39.     int T=1;
40. // cin>>T;
41. // clock_t start,finish;
42. // start=clock();

```

```

43.     while(cin>>n){
44.         if(!n) break;
45.         Solve();
46.     }
47. //  finish=clock();
48. //  cerr<<((double)finish-start)/CLOCKS_PER_SEC<<endl;  return 0;
49. }

```

HDU-1052 Tian Ji -- The Horse Racing

```

1. //pragma GCC optimize("Ofast", "inline", "-ffast-math")
2. //pragma GCC target("avx,sse2,sse3,sse4,mmx")
3. #include<bits/stdc++.h>
4. #define inf 0x3f3f3f3f
5. #define int long long
6. using namespace std;
7. const int N=2e5+7;
8. const int mod=1e9+7;
9.
10. //int read(){  int x=0,f=1;char ch=getchar();while(ch<'0' || ch>'9'){i
    f(ch=='-') f=f*-1;ch=getchar();}while(ch>='0'&&ch<='9'){x=x*10+ch-'0'
    ;ch=getchar();}return x*f;}
11. //void write(int x){if(x>9) write(x/10);putchar(x%10+'0');}
12.
13. int n,a[N],b[N];
14.
15. void Solve(){
16.     for(int i=1;i<=n;++i) cin>>a[i];
17.     for(int i=1;i<=n;++i) cin>>b[i];
18.     sort(a+1,a+1+n);
19.     sort(b+1,b+1+n);
20.     int L1=1,L2=1,R1=n,R2=n,ans=0;
21.     for(int i=1;i<=n;++i){
22.         if(a[R1]>b[R2]) --R1,--R2,ans+=200;
23.         else if(a[L1]>b[L2]) ++L1,++L2,ans+=200;
24.         else if(a[L1]<b[R2]) ++L1,--R2,ans-=200;
25.     }
26.     cout<<ans<<"\n";
27. }
28.
29. signed main(){
30. //  ios::sync_with_stdio(0);
31. //  cin.tie(0);cout.tie(0);
32. //  freopen("in.cpp","r",stdin);

```

```

33. // freopen("out.cpp","w",stdout);
34.   int T=1;
35. // cin>>T;
36. // clock_t start,finish;
37. // start=clock();
38.   while(cin>>n){
39.       if(n==0) break;
40.       Solve();
41.   }
42. // finish=clock();
43. // cerr<<((double)finish-start)/CLOCKS_PER_SEC<<endl; return 0;
44. }

```

LibreOJ-212 最短不公共子串

```

1. //pragma GCC optimize("Ofast", "inline", "-ffast-math")
2. //pragma GCC target("avx,sse2,sse3,sse4,mmx")
3. #include<bits/stdc++.h>
4. #define inf 0x3f3f3f3f
5. //define int long long
6. using namespace std;
7. const int N=2e5+7;
8. const int mod=1e9+7;
9.
10. int lena,lenb,lst,sz,root,pre[N],ch[N][26],dp[N<<1],sum[N],mp[N<<1];
11.
12. char A[N],B[N];
13. struct Tree{int len,fa,nxt[26];}T[N<<1]; //状态结点
14.
15. inline void SAM_extend(int c){ //在 SAM 中添加字符 c
16.     int np=++sz;
17.     T[np].len=T[lst].len+1;
18.     int x=lst;
19.     while(x&&!T[x].nxt[c]) T[x].nxt[c]=np,x=T[x].fa;
20.     if(!x) T[np].fa=root;
21.     else{
22.         int y=T[x].nxt[c];
23.         if(T[y].len==T[x].len+1) T[np].fa=y;
24.         else{
25.             int nq=++sz;
26.             T[nq]=T[y];
27.             T[nq].len=T[x].len+1;
28.             T[nq].fa=T[y].fa;

```

```

29.         T[y].fa=T[np].fa=nq;
30.         while(x&&T[x].nxt[c]==y) T[x].nxt[c]=nq,x=T[x].fa;
31.     }
32. }
33. }
34. void SAM_init(){
35.     lst=sz=root=1;
36.     for(int i=1;i<=lenb;i++) SAM_extend(B[i]-'a');
37.     for(int i=1;i<=lenb;i++){
38.         int c=B[i]-'a';
39.         for(int j=i-1;j>=pre[c];j--) ch[j][c]=i;
40.         pre[c]=i;
41.     }
42.     for(int i=1;i<=sz;i++) sum[T[i].len]++;
43.     for(int i=1;i<=lenb;i++) sum[i]+=sum[i-1];
44.     for(int i=1;i<=sz;i++) mp[sum[T[i].len]--]=i;
45. }
46.
47. inline int trans(int x,int c,int y){
48.     if(!y)return T[x].nxt[c];
49.     else return ch[x][c];
50. }
51.
52. void Solve(int f1,int f2){
53.     memset(dp,inf,sizeof(dp));
54.     dp[f2^1]=0;
55.     int ans=inf;
56.     for(int i=1;i<=lena;i++){
57.         int c=A[i]-'a';
58.         for(int z=(f2?lenb:sz);z>=(f2^1);z--){
59.             int x=f2?z:mp[z];
60.             int y=trans(x,c,f2);
61.             if(!y) ans=min(ans,dp[x]+1);
62.             else{
63.                 dp[y]=min(dp[y],dp[x]+1);
64.                 if(!f1&&x!=(f2^1)) dp[x]=inf;
65.             }
66.         }
67.     }
68.     if(ans>=inf) puts("-1");
69.     else printf("%d\n",ans);
70. }
71.
72. signed main(){

```

```

73. // ios::sync_with_stdio(0);
74. // cin.tie(0);cout.tie(0);
75. // freopen("in.cpp","r",stdin);
76. // freopen("out.cpp","w",stdout);
77.     scanf("%s%s",A+1,B+1);
78.     lena=strlen(A+1);lenb=strlen(B+1);
79.     SAM_init();
80.     Solve(0,0);
81.     Solve(0,1);
82.     Solve(1,0);
83.     Solve(1,1);
84. // finish=clock();
85. // cerr<<((double)finish-start)/CLOCKS_PER_SEC<<endl; return 0;
86. }

```

（五）实验步骤与调试

HDU-2037 今年暑假不 AC

题目问最多选中多少个区间使得彼此不重叠。经典题，按右端点从小到大排序，然后贪心地往右选即可。可以证明这样选的区间数总是最多的（用弦图和区间图的知识证明）

HDU-1052 Tian Ji -- The Horse Racing

题目给定 n 匹马的战斗力，问最多能赢多少局，进而算出总分。可以用双指针来做，把两人的马按战力从小到大排序，那么对于下等马和上等马，我们都需要尽量令其赢，如果两边都赢不了，那我们就选择用该下等马去博掉上等马。

LibreOJ-2123 最短不公共子串

SAM 的经典题吧应该。以 B 串建立后缀自动机，问题转化为最短的没有出现的字符串。在 SAM 上做 dp，设 $dp[i]$ 表示到结点 i 时的最短长度。如果这个点右不是 T 中字符的出边则 $dp[i]=1$ ，否则 $dp[i]=\min(dp[j])+1$ 。

（六）实验结果与分析

HDU-2037 今年暑假不 AC

结构体排一遍序直接遍历区间，判左端点和当前结束时间。


[#36673840](#) | [LINNO's solution for \[HDU-2037\]](#)

Status	Time	Memory	Length	Lang	Submitted	Open	Share text 	RemoteRunId
Accepted	15ms	1412kB	1062	G++	2022-06-14 17:30:34	<input checked="" type="checkbox"/>	<input type="checkbox"/>	37608979

HDU-1052 Tian Ji -- The Horse Racing

双指针分别指向田忌的下等马和上等马、王的下等马和上等马。然后判 n 轮的结果就行了。结果一共有 4 种，平局就是直接跳过。

[#36674662](#) | [LINNO's solution for \[HDU-1052\]](#)

Status	Time	Memory	Length	Lang	Submitted	Open	Share text 	RemoteRunId
Accepted	109ms	1436kB	1137	G++	2022-06-14 19:08:30	<input checked="" type="checkbox"/>	<input type="checkbox"/>	37609025

LibreOJ-2123 最短不公共子串

不懂贪心，板子套上去做。字符串永远的痛。

[#36677682](#) | [LINNO's solution for \[LibreOJ-2123\]](#)

Status	Time	Memory	Length	Lang
Accepted	192ms	6964kB	2167	G++ (ISO C++20) -O2 64bit

