

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 搜索算法 指导教师 张震
实验项目编号 0806017907 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) 掌握深度优先搜索算法。
- 2) 能够利用剪枝算法提高搜索效率。

（二）实验内容和要求

实验内容:

1. HDU 1238
2. HDU 1010
3. LibreOJ-10029
4. LibreOJ-10249

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

1) HDU 1238 Substrings

```
1. // #pragma GCC optimize("Ofast", "inline", "-ffast-math")
2. // #pragma GCC target("avx,sse2,sse3,sse4,mmx")
3. #include <bits/stdc++.h>
4. using namespace std;
5. const int N=207;
6. const int mod=1e9+7;
7.
8. // int read(){ int x=0,f=1;char ch=getchar();while(ch<'0' || ch>'9'){if(
    ch=='-') f=f*-1;ch=getchar();}while(ch>='0' && ch<='9'){x=x*10+ch-'0';c
    h=getchar();}return x*f;}
9. // void write(int x){if(x>9) write(x/10);putchar(x%10+'0');}
10.
11. string s[N];
12. int n;
13.
14. void Solve(){
15.     cin>>n;
16.     for(int i=1;i<=n;++i) cin>>s[i];
17.     s[0]=s[1];
18.     reverse(s[0].begin(),s[0].end());
19.     int ans=0,LEN=s[1].length();
20.     for(int i=LEN;i>=1;--i){
21.         for(int st=0;st+i-1<LEN;++st){
22.             string s1=s[1].substr(st,i);
23.             string s2=s[0].substr(st,i);
24.             int f1=1,f2=1;
25.             for(int j=2;j<=n;++j){
26.                 if(s[j].find(s1)==s[j].npos) f1=0;
27.                 if(s[j].find(s2)==s[j].npos) f2=0;
28.             }
29.             // cout<<i<<" "<<f1<<" "<<f2<<" "<<s1<<" "<<s2<<"!!\n";
30.             if(f1 || f2){
31.                 ans=i;
32.                 break;
33.             }
34.         }
35.         if(ans) break;
36.     }
37.     cout<<ans<<"\n";
38. }
39.
40. signed main(){
```

```

41. ios::sync_with_stdio(0);
42. cin.tie(0);cout.tie(0);
43. // freopen("in.cpp","r",stdin);
44. // freopen("out.cpp","w",stdout);
45. int T=1;
46. cin>>T;
47. // clock_t start,finish;
48. // start=clock();
49. while(T--){
50.     Solve();
51. }
52. // finish=clock();
53. // cerr<<((double)finish-start)/CLOCKS_PER_SEC<<endl;
54. return 0;
55. }

```

2) HDU 1010 Tempter of the Bone

```

1. //pragma GCC optimize("Ofast", "inline", "-ffast-math")
2. //pragma GCC target("avx,sse2,sse3,sse4,mmx")
3. #include<bits/stdc++.h>
4. using namespace std;
5.
6. int read(){ int x=0,f=1;char ch=getchar();while(ch<'0' || ch>'9'){if(ch==
    '-') f=f*-1;ch=getchar();}while(ch>='0'&&ch<='9'){x=x*10+ch-'0';ch=ge
    tchar();}return x*f;}
7. void write(int x){if(x>9) write(x/10);putchar(x%10+'0');}
8.
9. const int xx[]={0,0,1,-1},yy[]={1,-1,0,0};
10. int n,m,t,sx,sy,dx,dy,flag;
11. int ex[15][15];
12. char mp[15][15];
13.
14. inline int Abs(int x){return x<0?-x:x;}
15.
16. inline void dfs(int x,int y,int lf){
17.     if(x==dx&&y==dy&&lf==0) flag=1;
18.     int dis=lf-Abs(dx-x)-Abs(dy-y);
19.     if(flag || dis<0 || (dis&1)) return;
20.     for(int d=0;d<4;++d){
21.         int nx=x+xx[d],ny=y+yy[d];
22.         if(nx<1 || ny<1 || nx>n || ny>m || mp[nx][ny]=='X' || ex[nx][ny]) conti
nue;

```

```

23.         ex[nx][ny]=1;
24.         dfs(nx,ny,lf-1);
25.         ex[nx][ny]=0;
26.     }
27. }
28.
29. void Solve(){
30.     memset(ex,0,sizeof(ex));
31.     flag=0;
32.     for(int i=1;i<=n;++i){
33.         for(int j=1;j<=m;++j){
34.             cin>>mp[i][j];
35.             if(mp[i][j]=='S') sx=i,sy=j;
36.             if(mp[i][j]=='D') dx=i,dy=j;
37.         }
38.     }
39.     ex[sx][sy]=1;
40.     dfs(sx,sy,t);
41.     if(flag) cout<<"YES\n";
42.     else cout<<"NO\n";
43. }
44.
45. signed main(){
46.     // freopen("in.cpp","r",stdin);
47.     // freopen("out.cpp","w",stdout);
48.     // int T=1;
49.     // cin>>T;
50.     // clock_t start,finish;
51.     // start=clock();
52.     while(cin>>n>>m>>t){
53.         if(!n&&!m&&!t) break;
54.         Solve();
55.     }
56.     // finish=clock();
57.     // cerr<<((double)finish-start)/CLOCKS_PER_SEC<<endl;
58.     return 0;
59. }

```

3) LibreOJ-10029 棋盘游戏

```

1. //pragma GCC optimize("Ofast", "inline", "-ffast-math")
2. //pragma GCC target("avx,sse2,sse3,sse4,mmx")
3. #include<bits/stdc++.h>
4. using namespace std;
5. const int N=2e5+7;

```

```

6. const int mod=1e9+7;
7.
8. //int read(){ int x=0,f=1;char ch=getchar();while(ch<'0' || ch>'9'){if(
    ch=='-') f=f*-1;ch=getchar();}while(ch>='0'&&ch<='9'){x=x*10+ch-'0';c
    h=getchar();}return x*f;}
9.
10. struct X{
11.     string a[4];
12.     int step;
13. }s,t;
14.
15. int ans=1e7;
16. int xx[]={1,0,-1,0},yy[]={0,1,0,-1};
17. queue<X>q;
18. map<string,int>vis;
19.
20. string get(X x){
21.     string s="";
22.     for(int i=0;i<4;i++) s+=x.a[i];
23.     return s;
24. }
25.
26. signed main(){
27.     ios::sync_with_stdio(0);
28.     cin.tie(0);cout.tie(0);
29.     for(int i=0;i<4;i++){
30.         cin>>s.a[i];
31.         s.step=1;
32.     }
33.     for(int i=0;i<4;i++){
34.         cin>>t.a[i];
35.     }
36. // cout<<get(s)<<"\n";
37. q.push(s);
38. while(!q.empty()){
39.     X fro=q.front();
40.     q.pop();
41.     if(vis[get(fro)]) continue;
42.     vis[get(fro)]=fro.step;
43.     for(int i=0;i<4;i++){
44.         for(int j=0;j<4;j++){
45.             if(fro.a[i][j]!=t.a[i][j]){
46.                 for(int d=0;d<4;d++){
47.                     int nx=i+xx[d],ny=j+yy[d];

```

```

48.             if(nx>=0&&nx<4&&ny>=0&&ny<4){
49.                 swap(fro.a[i][j],fro.a[nx][ny]);
50.                 fro.step++;
51.                 q.push(fro);
52.                 swap(fro.a[i][j],fro.a[nx][ny]);
53.                 fro.step--;
54.             }
55.         }
56.     }
57. }
58. }
59. }
60. cout<<vis[get(t)]-1<<"\n";
61. return 0;
62. }

```

4) LibreOJ-10249

```

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=5e5+7;
8. const int mod=1e9+7;
9.
10. //int read(){ int x=0,f=1;char ch=getchar();while(ch<'0' || ch>'9'){
11.     f(ch=='-') f=f*-1;ch=getchar();}while(ch>='0'&&ch<='9'){x=x*10+ch-'0'
12.     ;ch=getchar();}return x*f;}
13. int n,m,sum=0,flag=0,a[N];
14. int ans[N];
15.
16. vector<int>vis(N);
17.
18. void dfs(int pos,int l,int r,int pre,int suf){
19.     if(flag) return;
20.     if(!vis[a[pos]-pre]&&!vis[a[pos]-suf]) return;
21.     if(l==r){ //取完 n 个了
22.         ans[l]=a[2*n]-pre-suf; //这是最后一个数
23.         if(ans[l]<1||ans[l]>500) return;
24.         for(int i=1;i<=n;i++) cout<<ans[i]<<" ";
25.         flag=1;
26.         return;
27.     }

```

```

26.     if(vis[a[pos]-pre]){ //顺着来取
27.         ans[l]=a[pos]-pre;
28.         dfs(pos+1,l+1,r,a[pos],suf);
29.     }
30.     if(vis[a[pos]-suf]){ //逆着来取
31.         ans[r]=a[pos]-suf;
32.         dfs(pos+1,l,r-1,pre,a[pos]);
33.     }
34. }
35.
36. signed main(){
37.     ios::sync_with_stdio(0);
38.     cin.tie(0);cout.tie(0);
39.     cin>>n;
40.     for(int i=1;i<=2*n;i++) cin>>a[i];
41.     sort(a+1,a+1+2*n);
42.     cin>>m;
43.     for(int i=1,x;i<=m;i++){
44.         cin>>x;
45.         vis[x]=1;
46.     }
47.     dfs(1,1,n,0,0);
48.     return 0;
49. }

```

（五）实验步骤与调试

1) HDU 1238 Substrings

要求找到一个子串，使得在所有（字符串或其反串）都出现过。使用字符串的 `find` 操作，遍历 $2 \sim n$ 所有字符串，对 `s1` 和 `s1` 的反串的所有子串判一下就完事了。

2) HDU 1010 Tempter of the Bone

深搜剪一下枝。首先为了能达到恰好 `t` 时间到达也许会绕一下圈嘛，这里可以用曼哈顿距离的奇偶性剪一下，然后就是正常做。

3) LibreOJ-10029 棋盘游戏

直接 BFS。用一个 `map` 记录棋盘的状态以及他需要的步数即可。

4) LibreOJ-10249 Weight

这道题要求剪枝力度比较足，不过他要搜的数是非常有限的，我们记一下出现的前缀和以及后缀和，然后可以记录根据他们的出现次数来 `dfs`，最好双向，这样可以对每个数都找一遍，然后判断正放还是反放。单向的话要搜比较多的数，不太好剪。

（六）实验结果与分析

1) HDU 1238 Substrings

直接找子串完事了。

[#36680779](#) | [LINNO's solution for \[HDU-1238\]](#)

Status	Time	Memory	Length	Lang	Submitted	Open
Accepted	140ms	1480kB	1272	G++	2022-06-15 13:49:42	<input checked="" type="checkbox"/>

2) HDU 1010 Tempter of the Bone

这题的输入真的有毒……WA 了半天。

[#36681430](#) | [LINNO's solution for \[HDU-1010\]](#)

Status	Time	Memory	Length	Lang	Submitted	Open
Accepted	296ms	1408kB	1391	G++	2022-06-15 15:06:26	<input checked="" type="checkbox"/>

3) LibreOJ-10029 棋盘游戏

没啥好注意的。

[#36681467](#) | [LINNO's solution for \[LibreOJ-10029\]](#)

Status	Time	Memory	Length	Lang
Accepted	300ms	8716kB	1247	G++ (ISO C++20) -O2 64bit

4) LibreOJ-10249

搞定！

[#36681548](#) | [LINNO's solution for \[LibreOJ-10249\]](#)

Status	Time	Memory	Length	Lang
Accepted	47ms	6448kB	1125	G++ (ISO C++20) -O2 64bit

暨南大学本科实验报告专用纸(附页)
