

```

1 // [反片語/Anagrams](1/2)
2 #define IN "P16IN.txt"
3 #define OUT "P16OUT.txt"
4 //*****
5 #include <iostream>
6 #include <ctime>
7 using namespace std;
8 void redir(void);
9 //*****
10 /* Work Space*/
11 #include <string>
12 #include <vector>
13 #include <algorithm>
14 #include <map>
15
16 string normalize(string &s);
17 //*****
18 int main(void)
19 {
20     redir(); //redirection
21 //*****
22 /* Work Space*/
23     string s, n;
24     vector<string> words, ans;
25     map<string, int> cnt;
26     int i;
27
28     while(cin >> s){
29         if(s[0] == '#'){
30             break;
31         }
32         words.push_back(s);
33         n = normalize(s); //'=' is overloaded
34
35         if(!cnt.count(n)){
36             cnt[n] = 0;
37         }
38         cnt[n]++;
39     }
40
41     for(i=0; i<words.size(); i++){
42         if(cnt[normalize(words[i])] == 1){
43             ans.push_back(words[i]);
44         }
45     }
46     sort(ans.begin(), ans.end());
47
48     for(i=0; i<ans.size(); i++){
49         cout << ans[i] << endl;
50     }
51 //*****
52     freopen("CON", "r", stdin); //取消重新導向
53     freopen("CON", "w", stdout);
54
55     printf("Time used = %.2f\n", (double)clock()/CLK_TCK); //傳回程式目前為止執行的時間
56
57     system("pause");
58     return 0; //the end...
59 }
60
61
62
63

```

```

64 //[反片語/Ananagrams](2/2)
65 void redir(void)
66 {
67     freopen(IN, "r", stdin);
68     freopen(OUT, "w", stdout);
69 }
70 //*****
71 /* Work Space*/
72 //把每個單字標準化(轉換為小寫字母再按進行排序)
73 string normalize(string &s)
74 {
75     string ans = s; //'=' is overloaded
76     int i;
77
78     for(i=0; i<ans.length(); i++){
79         ans[i] = tolower(ans[i]);
80     }
81
82     sort(ans.begin(), ans.end());
83     return ans;
84 }
85
86 //Input(IN) Sample
87 /*
88 ladder came tape soon leader acme RIDE lone Dreis peat
89 ScAlE orb eye Rides dealer NotE derail LaCes drIed
90 noel dire Disk amce Rob Dries
91 #
92 */
93
94 //Output(OUT)
95 /*
96 Disk
97 NotE
98 derail
99 drIed
100 eye
101 ladder
102 soon
103 */

```