

TITLE. Олимпиадная информатика

File: ljjournal.c

Задача. Программист Саша участвует в создании блог-сервиса Живой Журнал. Планируется, что этот сервис будет предоставлять гораздо больше возможностей, чем известный всем LiveJournal. В настоящее же время проблему составляет реализация все базовых возможностей LiveJournal'a. Одной из таких возможностей является поддержка списка друзей для пользователя.

Заданы: список пользователей, являющихся друзьями данного пользователя, и список пользователей, у которых данный пользователь содержится в списке друзей.

Необходимо получить список друзей данного пользователя (Friends), список его взаимных друзей (MutualFriends), и список тех пользователей, у кого данный пользователь содержится в списке друзей, но которые не являются его взаимными друзьями (Also Friend of).

Первая строка входного файла INPUT.TXT содержит число n ($0 \leq n \leq 200$) друзей данного пользователя. Следующие n строк содержат каждая по одному имени пользователя, который является другом данного, $(n + 2)$ -ая строка содержит число m ($0 \leq m \leq 200$) пользователей, у которых данный содержится в списке друзей. Далее заданы имена пользователей, у которых данный содержится в списке друзей. Эти пользователи заданы в том же формате, что и друзья данного.

Имена пользователей – строки длиной не более 20 символов, содержащие только строчные буквы латинского алфавита и символы тире ('-'). Каждый пользователь указан не более одного раза в каждом из списков.

В выходной файл OUTPUT.TXT следует вывести список друзей данного пользователя (Friends), список его взаимных друзей (Mutual Friends), и список тех пользователей, у кого данный пользователь содержится в списке друзей, но они не являются его взаимными друзьями (Also Friend of).

В каждом списке пользователи должны быть отсортированы по алфавиту. Следовать формату, приведенному в примерах.

INPUT.TXT

2

vasya-pupkin

OUTPUT.TXT

Friends: bill-hates, vasya-pupkin,

Mutual Friends: vasya-pupkin

```
bill-hates      Also Friend of: destroyer
2
vasya-pupkin
destroyer
```

```
-----
INPUT.TXT      OUTPUT.TXT
0              Friends:
0              Mutual Friends:
              Also Friend of:
-----
```

Решение. Калмыков Вадим (ProCrypt),
г. Сургут, ЦНИТ "Северная Звезда",
11:30, 24.05.2008

Source : <http://acm.dvpion.ru>

Edit. MSP, 02.12.09, 11:26, (c) LIST2008

*/

```
60 # include <stdio.h>
61 # include <string.h>

62 int main ()
63 {
64     int
65         i, j, // счетчики циклов
66         n, // число друзей пользователя
67         m, // число пользователей, у которых данный есть в списке друзей
68         k, // число взаимных друзей
69         u; // число Also Friend of

70     char s [21];

71     int Search (char A [] [21], int n, char s [21]);

72     void Sort (char A [] [21], int n); // функция сортировки строк
73                                         // в списке

74     char F [200] [21]; // Friends
75     char AF [200] [21]; // Also Friend of
76     char MF [200] [21]; // Mutual Friends

77     freopen ("input.txt", "r", stdin); // файл для чтения
78     freopen ("output.txt", "w", stdout); // файл для записи

79     // считываем данные из входного файла
80     scanf ("%i\n", & n);
81     for (i = 0; i < n; i ++)
82         gets (F [i]);
```

```

83     Sort (F, n); // сортируем список друзей

84     k = -1;
85     u = -1;
86     scanf ("%i\n", & m);
87     for (i = 0; i < m; i ++)
88     {
89         gets (s);
90         if (Search (F, n, s))
91             {
92                 k ++;
93                 strcpy (MF [k], s);
94             }
95         else
96             {
97                 u ++;
98                 strcpy (AF [u], s);
99             }
100    }

101    Sort (MF, k + 1);
102    Sort (AF, u + 1);

103    printf ("Friends: ");
104    for (i = 0; i < n - 1; i ++, printf (" "))
105        for (j = 0; j < strlen (F [i]); j ++)
106            if (F [i] [j] != ' ')
107                putchar (F [i] [j]);

108    if (n >= 1)
109        for (j = 0; j < strlen (F [n - 1]); j ++)
110            if (F [n - 1] [j] != ' ')
111                putchar (F [n - 1] [j]);

112    putchar ('\n');

113    printf ("Mutual Friends: ");
114    for (i = 0; i < k; i ++, printf (" "))
115        for (j = 0; j < strlen (MF [i]); j ++)
116            if (MF [i] [j] != ' ')
117                putchar (MF [i] [j]);

118    if (k >= 0)
119        for (j = 0; j < strlen (MF [k]); j ++)
120            if (MF [k] [j] != ' ')
121                putchar (MF [k] [j]);

122    putchar ('\n');

123    printf ("Also Friend of: ");
124    for (i = 0; i < u; i ++, printf (" "))

```

```

125     for (j = 0; j < strlen (AF [i]); j ++)
126         if (AF [i] [j] != ' ')
127             putchar (AF [i] [j]);

128     if (u >= 0)
129         for (j = 0; j < strlen (AF [u]); j ++)
130             if (AF [u] [j] != ' ')
131                 putchar (AF [u] [j]);

132     fclose (stdin);
133     fclose (stdout);

134     return 0;
135 }

136 void Sort (char A [] [21], int n)
137 {
138     int i, j, k; // счетчики циклов

139     char ch [21]; // массив для обмена строк

140     int L; // длина сравнения (длина имени)

141     for (i = 0; i < (n - 1); i ++)
142         for (j = 0; j < (n - 1); j ++)
143             {
144                 // определяем длину сравнения
145                 if (strlen (A [j]) > strlen (A [j + 1]))
146                     L = strlen (A [j]);
147                 else
148                     L = strlen (A [j + 1]);

149                 // теперь посимвольно сравниваем строки
150                 for (k = 0; k < L; k ++)
151                     {
152                         if (A [j] [k] > A [j + 1] [k])
153                             {
154                                 strcpy (ch, A [j]);
155                                 strcpy (A [j], A [j + 1]);
156                                 strcpy (A [j + 1], ch);

157                                 break;
158                             }
159                         else
160                             if (A [j] [k] < A [j + 1] [k])
161                                 break;
162                     }

163                 } // for (j = 0; j < (n - 1 - i); j ++)
164 } // void Sort (char A [] [21], int n)

```

```

165     int Search (char A [] [21], int n, char s [21])
166     {
167         int i, j;

168         int flag;

169         int L;

170         for (i = 0; i < n; i ++)
171             {
172                 if (strlen (A [i]) > strlen (s))
173                     L = strlen (A [i]);
174                 else
175                     L = strlen (s);

176                 flag = 1;
177                 for (j = 0; j < L; j ++)
178                     if (A [i] [j] != s [j])
179                         {
180                             flag = 0;
181                             break;
182                         }

183                 if (flag)
184                     return 1;
185             }

186         if (flag == 0)
187             return 0;
188         else
189             return 1;
190     }

```

Listing данной задачи опубликован в сети Internet по адресу
<http://www.Best-Listing.ru/color-11-task-143.html>

Sergey Mitrofanov, 19.09.14, 20:16

E-mail: infostar@mail.ru

© <http://www.Best-Listing.ru/>, 2006–2014