Emoticons

Assignment 6

Due date: Wednesday, 11 Dec (11:59 pm)

1 Overview

In this assignment, students will develop and implement a program that performs “lookups” in a table of character strings. The primary objective of this assignment is for students to implement search algorithms for arrays of character strings.

2 The Program

Emoticons are character strings that convey some kind of emotion, mood, or action; they are more commonly called “smiley”, and used in Internet programs such as email or chat. For example, the emoticon :-) conveys the act of “smiling”.

The program will work with the following fixed set of 16 emoticons:

<table>
<thead>
<tr>
<th>EMOTICON</th>
<th>MEANING</th>
</tr>
</thead>
<tbody>
<tr>
<td>O:-)</td>
<td>angelic</td>
</tr>
<tr>
<td>:-</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-I</td>
</tr>
<tr>
<td>:)-(</td>
<td>crying</td>
</tr>
<tr>
<td>%-)</td>
<td>drunk</td>
</tr>
<tr>
<td>:-(</td>
<td>frowning</td>
</tr>
<tr>
<td>:-x</td>
<td>kissing</td>
</tr>
<tr>
<td>:-D</td>
<td>laughing</td>
</tr>
<tr>
<td>:-[</td>
<td>pouting</td>
</tr>
<tr>
<td>:-@</td>
<td>screaming</td>
</tr>
<tr>
<td>:-/</td>
<td>skeptical</td>
</tr>
<tr>
<td>:-)</td>
<td>smiling</td>
</tr>
<tr>
<td>8-)</td>
<td>sunglasses</td>
</tr>
<tr>
<td>:-&amp;</td>
<td>toungtied</td>
</tr>
<tr>
<td>;-)</td>
<td>winking</td>
</tr>
<tr>
<td></td>
<td>-O</td>
</tr>
</tbody>
</table>
Specifically, the program allows the user to determine either (i) the meaning of a specific emoticon; or (ii) the emoticon conveying a specific meaning.

3 Program Structure

The table of emoticons can be represented by two arrays of character strings: one for the emoticons, and one for the meanings. The element with subscript $i$ of the “meaning” array gives the meaning of the element of the “emoticon” array with subscript $i$. (Arrays used in this fashion are called parallel arrays.)

3.1 The main Function

The main function of the program prompts the user to choose which of the two possible actions they wish to have performed. After that choice is made, the user’s text string is scanned in, and a search function is called; the value returned by this function is used in producing the final output.

3.2 The linearSearch Function

The main function calls the linearSearch function to search the “emoticon” array for a specific emoticon. The linearSearch function has the following prototype:

```c
int linearSearch(char *a[], int n, char key[]);
```

It performs an linear search of the array of character strings $a$ with effective size $n$ for the specific character string $key$. If found, it returns the index of element of $a$ that is identical to $key$; otherwise, it returns $-1$.

The only “trick” to searching an array of character strings is to use the `strcmp` library function to compare two strings for equality.

3.3 The binarySearch Function

In theory, the linearSearch function could also be used to search the “meaning” array for a specific meaning. But notice that the meanings in the emoticon table have been sorted into alphabetical order. By keeping this order intact in the “meaning” array, the binary search algorithm can be used to search the “meaning” array.

The binarySearch function has the same parameters as the linearSearch function, and performs the same actions. The only difference is the algorithm used.
4 Sample Output

Here, the user enters an emoticon, and get its meaning.

OPTIONS

(1) Look up the meaning conveyed by a given emoticon
(2) Look up the emoticon that conveys a given meaning

Choice: 1

Enter an emoticon: 8-)

The emoticon 8-) corresponds to the meaning sunglasses

Here, the user performs the reverse action.

OPTIONS

(1) Look up the meaning conveyed by a given emoticon
(2) Look up the emoticon that conveys a given meaning

Choice: 2

Enter an meaning: drunk

The meaning drunk corresponds to the emoticon %-)

Here, the user enter an invalid meaning.

OPTIONS

(1) Look up the meaning conveyed by a given emoticon
(2) Look up the emoticon that conveys a given meaning

Choice: 2

Enter an meaning: elated

Sorry, the meaning elated is not in our database