CS6326 Human-Computer Interactions

Fall 2017

Assignment 5: Multithreaded Text Search

Write a program that searches a text file for a string. It must find all occurrences of the string and show them in a list. Since the text file could be very long, and lists in C# do not display until the button-press code that fills them exits, your program must be multithreaded. The program should do the following:

1. Have a textbox for the name of a text file. You can also have a Browse button that will let you find a file to put in the textbox.
2. Another textbox will accept text to be found in the document.
3. A search button will cause the document to be searched.
4. A listview control displays the entire line on which the text was found and the line number within the document.
5. Search each line of text **as you read it** for the search string. You should probably do a case-insensitive comparison. Every time the string is found, put the entire line in which it occurs in the list along with its line number. To make it interesting, put in a pause of 1 millisecond every time you read a line. This will simulate a long document.
6. The Search button text should be change to “Cancel” once the search starts, and should cancel the operation when pressed.
7. Add any other features to this program that you think may make it more usable or understandable to the average computer user. (Do not e-mail me to ask if this or that feature is allowable. Use your knowledge of design, and material provided, to make the determination. In-class questions and discussion are welcome, however.)

You don’t need to have regular expressions to do the comparison; a simple string search will suffice.

I have provided a long text file that is complete text of *The Divine Comedy* (Inferno, Purgatorio, and Paradiso) by Dante Alighieri. Searching for words like *paradise, fantasy, love, human, gold,* and others should yield interesting results.The document contains approximately 18,000 lines of text, so with the delay it should take over a minute to search. (My version takes nearly 4 minutes.)

A possible partial screen design is shown on the next page.

|  |  |
| --- | --- |
| **Grading: 100 points possible** | |
| Program is truly multithreaded and works according to the above specification | 50 |
| Program follows other UI design rules and guidelines | 30 |
| Additional features for usability/learnability | 10 |
| Program comments and good variable names | 10 |

Grading guidelines:

1. The program reads everything into memory, then searches an array: -50
2. The program waits until it finishes to display anything: -50
3. The program uses only the screen below as is, without any further changes: -10

