CS 1704: Program #2
Reference Set Manager
Due: Friday Nov. 1, 1996

Problem

Code a program to manage a set of files for on-line code library reference.

Discussion

This program will allow a user to create and manage a set of category and documentation files for an on-line documentation system. (See program 1, “Validator” specifications for a description of the category and documentation file contents.) In addition to allowing a user to create new reference sets, the program will also allow a user to perform management operations upon files, (elements) of a reference set.

The interface for this program will be simple one-line menus. A user chooses an operation by entering the first letter of the operation. After the operation completes a menu line is then (re)displayed. The initial menu line will provide a user with the following operations:

Reference Set: (N)ew, (O)pen, or (Q)uit?

The New operation allows the user to enter the name of an existing root category file. The root category file is read and the reference set is built. (This involves building a double-linked list(s) of all of the category and documentation files derived from the root category files, ordered upon the file names.) After the reference set linked-list has been built, the second menu line is displayed. The Open operation allows a user enter an existing reference set file name, (for a description of the reference set file format see the Input section below). After the reference set file information has been read, it is automatically opened and the second menu line displayed. The Quit operation queries the user to be sure they wish to leave the system (basically to annoy them ☺), and then issues a brief termination message before returning to the operating system.

The second menu line will provide a user with the following operations:

***Reference Set File name*: (A)dd, (D)elete, (M)odify, (L)ist, (V)iew, or (C)lose?**

The Add operation allows a user to enter the name of a file to be inserted into the current reference set (linked-list). The user is queried for the type of the file (category or documentation), a label for the file, followed by the name of the file. (The program does not provide the user with editing facilities to actually create the file. The user is required to use an external editor for actual file content entry.) The Delete operation queries the user for the name of a file, which is then removed from the current reference set. (If the file is not in the current reference set an appropriate error message must be given.) The Modify operation queries the user for the name of a file, it then gives the user the option of changing the name of the file, the label for the file and/or the type of the file. The List operation simply displays the contents of the reference set, (i.e., the name, label and type of each file in the set). It should display the reference set contents in a labeled tabular format, 20 list elements at a time, allowing the user to hit the return key to list the next twenty elements. The View operation allows a user to see the contents of each file in the reference set. It begins by showing the user the name of the first file in the reference set and querying them to see if they wish to View it, Skip it or Halt viewing. If they wish to view the file, it’s contents, unlabelled, are simply displayed on the screen twenty lines at a time, (allowing the user to hit the return key to list the next twenty lines). When the entire file has been displayed or the user chooses to skip a file, the View operation continues with the name of the next file in the reference set. The Halt viewing command simply stops the View operation, returning to the reference set file management menu line above. The Close operation first queries to user to be sure they wish to close the current reference set, (and to continue to really annoy them ☺), writes out the reference set file, (if it has been changed), destroys the reference set list, closes all open files and returns control to the reference set, (initial), menu line.

This program will later be incorporated into an on-line code library reference system. Note: standard double linked-list operations, (insert, delete, etc.), from the text may be used, but must be properly referenced. Code from other sources must be approved by the instructor before it can be used. (To use code from other sources without prior approval is an Honor Code violation.) The reference set linked-list need not store all of the contents of each category and documentation file, (although it could). It need only store the name, label and file type, (category or documentation), of each category and documentation file.

Execution

The program should initially present the user with a startup screen displaying the name of the program, a brief 3-4 line explanation of the program and the programmer's name, and email address. After the user hits the return key, the startup screen is to be cleared. A brief help screen should appear explaining the program, which is cleared upon the pressing of the return key. (The startup and help screen may be combined if desired.) Following the brief help screen the reference set, (initial), menu line is displayed and execution continues as described above in the discussion section until the user Quits the system.

This date was incorrectly listed as Nov. 18 on the class hardcopy.

Structure Charts

An initial structure chart design must be submitted on **Friday Oct. 18**. A final structure chart reflecting design changes and corresponding to the code must also be produced and submitted with the executable. The initial chart will be compared against the final chart to determine the quality of the design.

Input

The format of the reference set file requires that each line in the file adhere to the following column format:

columns	*contents*	*comments*
1	'C' or 'D'	File type entry code: C-category, D-documentation.
5-15	Label	Label for file.
20-80	file name	File name of entry with extension (.gor, .dr). May contain relative or absolute pathname of the file.

Assumptions

It may NOT be assumed that a reference set will contain any maximum number of files limit. Error checking must be performed whenever possible, (file existence, illegal menu operation selection, etc.). However, no error checking, (validation), on the category or documentation files during the viewing phase need to be performed at this time, (this checking will be incorporated in the final project). Data files will be provided. Downloading instructions will be posted on the course World Wide Web site: (**`http://ei.cs.vt.edu/~cs1704`**).

Grading

Separate compilation and an abstract double linked-list data structure, (pointer implementation), is required for this program. Failure to utilize these features in the program will result in severe grade penalties. Turn in hard copies of the source code, input/output files, structure chart and a diskettes (system labeled: Win 3.1, 95 or NT), with files containing: ASCII source code, executable image, all project workspace files, I/O files, and a brief ASCII *readme* file with execution instructions. The executable image must be called *referset.exe*. The files on the disk may be zip compressed in self extracting archives and file protection should be set to allow anyone access. All deliverables must be placed in an envelope folder, neatly labeled. Do NOT submit any 5 1/4" MSDOS disks. Only 3 1/2 inch DOS (FAT)1.4 MB disks are to be submitted.

In addition, the GTAs may require you to demonstrate your program. To receive partial credit for programs that are non-working, or are not fully functional, a brief one or two paragraph description of the problem(s) must be included in the assignment folder. The location, routine minimum, must also be specified along with corrections that need to be made. Non-working/non-implemented menu operations must not bomb the program when selected. They must display a message informing the user that the features are not currently implemented.