Coding Assignment 6

Spring 2022

Coding Assignment 6 starts with your Coding Assignment 4. Please copy Coding Assignment 4 (CA4) to Coding Assignment 6 (CA6) to start with this assignment. CA4 prompted the user for draw commands and executed those commands on a map of a size the user chose. CA6 will prompt the user for any 3 letters and will draw those 3 letters using the commands read in from a file. The draw commands in the file will be stored in a linked list and each requested letter will be drawn by finding all of the commands associated with each letter that are stored in the linked list.

Step 1 – Alter your makefile.

For this assignment, you will be adding a `ListLib.h` and a `ListLib.c` and a `FileLib.c` and a `FileLib.h`. The `ListLib` library will contain the function you create to add nodes into the linked list and the function to search the linked list for a given letter. The `FileLib` library will contains the functions to open and read the file.

You will add to your makefile from CA4 by keeping `DrawTool.c/DrawTool.h` and adding `ListLib.c` and `FileLib.c`.

Part 2 – File Processing – `FileLib.c/FileLib.h`

Use the provided template – `FileLib-Template.c`.

CA6 will need to open a file and read it. Each record will be tokenized and added to the linked list. The format of each line will be…

```
Letter|Draw Command
```

For example

```
D|V(0,0,10)&
D|H(0,1,2)(
D|H(9,1,2)@
D|V(1,3,8)!
```

The filename will be passed into the program as a command line argument. Do not hardcode a file name in your program – you do not know the filename that will be used to test your code. Call `OpenFile()` from `main()` and pass in `argc` and `argv` and return the file handle.

After calling `OpenFile()` in `main()`, call `ReadFileIntoLinkedList()` from `main()`. Pass in the file handle and the address of the linked list head. Nothing is returned. Use `fgets()` with the file handle to read through the file. Use `strtok()` to parse each line into a letter and a draw command. Call function `AddDrawCommandToList()` and pass it the letter, the draw command and the linked list head. `AddDrawCommandToList()` resides in `ListLib.c`.

The prototypes for `OpenFile()` and `ReadFileIntoLinkedList()` both reside in `FileLib.h`. **You are required to use `FileLib.h` exactly as is – do not change it.** Your code will be graded using the assignment's version – not your version. Do not submit it with your assignment.

BONUS – 10%

Add advanced command line parameter handling so that your program can be run as

Code6_xxxxxxxxxx.e FILENAME=yyyyyy

where yyyyy is the name of the containing the draw commands.

Part 3 – Linked List – `ListLib.c/ListLib.h`

Use the provided template – `ListLib-Template.c`

Each record from the file will contain a letter and its draw command. Each letter will have 1 or more draw commands. Each letter and draw command should be stored as data in the linked list. Create the linked list node such that it can hold this data. A node will be defined in `ListLib.h` and the **data** in it will consist of a letter (single character), a draw command (char *). You don't know how long each draw command will be; therefore, you should `malloc()` the needed memory and store the pointer in the linked list node. The code for handling the linked list should be in `ListLib.c/ListLib.h`.

The prototypes for `AddDrawCommandToList()` and `FindLetter()` both reside in `ListLib.h`. Please note that prototypes are **not** required to include variable names. **You are required to use `ListLib.h` exactly as is – do not change it.** Your code will be graded using the assignment's version – not your version.

Step 4 – In `main()` in your `Code6.c` file

Your linked list head must be declared local to `main()` and be passed to functions that need it.

Rather than prompt the user for a single draw command, you should prompt for 1-3 letters. We are not going above 3 to ensure they properly fit on the screen. Ensure that at least one letter is entered and that no more than 3 are entered.

Your program should take each letter from the input, search the linked list for all commands associated with that letter and print out each command for that letter as they are found in the linked list. When printing the 2nd and 3rd letters, you will need to take into account that the letters need to shift over. My suggestion is to shift the 2nd letter over by 7 places and the 3rd letter over by 14 places. The draw commands are all created based on being the first letter printed – you need to move them over when they are the second or third letter.

The `FindLetter()` function returns the address of the node where your linked list search left off. The first call to `FindLetter()` should use the linked list head so that the search starts at the beginning of linked list. Every call after the first one should use the `NODE` value returned by `FindLetter()` so that the search starts where it left off. When your search reaches the end of the linked list, `NULL` will be returned; therefore, you need to base your `while` loop in `main()` on whether or not the draw command passed back by `FindLetter()` is empty or not. `FindLetter()` will set the draw command to `NULL` when the end of the linked list is reached; therefore, you can check either the `strlen()` of the draw command or check if the 0th element is `\0` to see if the draw command has been `NULL`ed by `FindLetter()`.

```
for loop over strlen of numbers of letters entered
{
      uppercase the current letter

      // Find the first node containing your first letter and
      // get back a bookmark to the next node
      TempPtr = FindLetter(LinkedListHead, current letter, draw command);

      // Either check the strlen() of draw command (0 means it was set to NULL)
      // or check if element [0] of draw command is '\0'
      // You want to continue to loop while your draw command has something in it
      while (draw command has a value that needs to be processed)
      {
            tokenize and draw the command

            // Look for the next node containing the letter
```

```
            // draw command will be NULL if not found
            TempPtr = FindLetter(TempPtr, current letter, draw command);
        }
    }
```

Part 5 - Testing

Create a test file using your original input file submitted with CA4. Modify it to use include the letter and the pipe delimiter. Run your `Code6.e` and confirm that your program can draw any combination of your three letters. Your program will be graded with a file that contains all letters and multiple commands per letter.

Part 6 – Code Submission

Submit a zip file containing the following files

`Code6_xxxxxxxxxx.c`

`ListLib.c`

`DrawTool.c`

`FileLib.c`

`makefile`

Coding Hints

Make no assumptions about how many lines will be in the file and about how many commands will be needed per letter. Make no assumptions about the ordering of the commands themselves in the file. The first command to create Z may be in the middle of the file when the second command is the first record and the 3rd command is last in the file. The ordering of the commands and the number of commands per letter should not matter to your logic. You should never save multiple linked list elements into an array at any point – that would be making an assumption about how many commands there may be per letter.

Please remove the prompt for the size of the map in order to make grading easier for the GTAs. Set your map size to 20. You should be using your map size define and not hardcoded 20.

Your program should be altered to no longer accept drawing commands – your program will only accept 1-3 letters and print them using the draw commands from the input file. You should no longer print the draw command instructions since the user is not entering a draw command.

When you pass the draw command from the linked list node, make a copy of it in a local array and run `strktok()` on that copy. Do not use `strtok()` directly on your node data since `strtok()` changes the array given to it.

Input and Output

I took my input file from Coding Assignment 4 and altered it to make an input file for this assignment. You can do the same for testing. A different file will be used for grading.

Input File

```
D|V(0,0,10)D
D|H(0,1,2)D
D|H(9,1,2)D
D|V(1,3,8)D
M|V(0,0,10)M
M|V(0,4,10)M
M|P(1,1,1)M
M|P(2,2,1)M
M|P(1,3,1)M
F|H(4,1,2)F
F|V(0,0,10)F
F|H(0,1,3)F
```

```
student@maverick:/media/sf_VM/CSE1320/CA6$
./Code6_1000074079.e input.txt
What is the background character? .
Please enter 1-3 letters mmm

M . . . M . . M . . . M . . M . . . M .
M M . M M . . M M . M M . . M M . M M .
M . M . M . . M . M . M . . M . M . M .
M . . . M . . M . . . M . . M . . . M .
M . . . M . . M . . . M . . M . . . M .
M . . . M . . M . . . M . . M . . . M .
M . . . M . . M . . . M . . M . . . M .
M . . . M . . M . . . M . . M . . . M .
M . . . M . . M . . . M . . M . . . M .
. . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . .
```

```
student@maverick:/media/sf_VM/CSE1320/CA6$
./Code6_1000074079.e input.txt

What is the background character? *

Please enter 1-3 letters fmd



F   F   F   F   *   *   *   M   *   *   *   M   *   *   D   D   D   *   *   *
F   *   *   *   *   *   *   M   M   *   M   M   *   *   D   *   *   D   *   *
F   *   *   *   *   *   *   M   *   M   *   M   *   *   D   *   *   D   *   *
F   *   *   *   *   *   *   M   *   *   *   M   *   *   D   *   *   D   *   *
F   F   F   *   *   *   *   M   *   *   *   M   *   *   D   *   *   D   *   *
F   *   *   *   *   *   *   M   *   *   *   M   *   *   D   *   *   D   *   *
F   *   *   *   *   *   *   M   *   *   *   M   *   *   D   *   *   D   *   *
F   *   *   *   *   *   *   M   *   *   *   M   *   *   D   *   *   D   *   *
F   *   *   *   *   *   *   M   *   *   *   M   *   *   D   *   *   D   *   *
F   *   *   *   *   *   *   M   *   *   *   M   *   *   D   D   D   *   *   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
```

```
student@maverick:/media/sf_VM/CSE1320/CA6$
./Code6_1000074079.e input.txt

What is the background character? !

Please enter 1-3 letters fmf




F  F  F  F  !  !  !  M  !  !  !  M  !  !  F  F  F  F  !  !

F  !  !  !  !  !  !  M  M  !  M  M  !  !  F  !  !  !  !  !

F  !  !  !  !  !  !  M  !  M  !  M  !  !  F  !  !  !  !  !

F  !  !  !  !  !  !  M  !  !  !  M  !  !  F  !  !  !  !  !

F  F  F  !  !  !  !  M  !  !  !  M  !  !  F  F  F  !  !  !

F  !  !  !  !  !  !  M  !  !  !  M  !  !  F  !  !  !  !  !

F  !  !  !  !  !  !  M  !  !  !  M  !  !  F  !  !  !  !  !

F  !  !  !  !  !  !  M  !  !  !  M  !  !  F  !  !  !  !  !

F  !  !  !  !  !  !  M  !  !  !  M  !  !  F  !  !  !  !  !

F  !  !  !  !  !  !  M  !  !  !  M  !  !  F  !  !  !  !  !

!  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !

!  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !

!  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !

!  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !

!  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !

!  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !

!  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !

!  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !

!  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !

!  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !  !
```

Output when program is run without a filename on the command line and the correct filename is provided at the prompt.

```
student@maverick:/media/sf_VM/CSE1320/CA6$ ./Code6_1000074079.e
Must be run with an input file name.
Enter a file name at the prompt input.txt
What is the background character?
```

Output when program is run without a filename on the command line and the correct filename is provided at the prompt after several tries.

```
student@maverick:/media/sf_VM/CSE1320/CA6@ ./Code6_1000074079.e
Must be run with an input file name.
Enter a file name at the prompt cat.dog
Could not open input file named cat.dog.
Enter a file name at the prompt dog.cat
Could not open input file named dog.cat.
Enter a file name at the prompt input
Could not open input file named input.
Enter a file name at the prompt input.txt
What is the background character?
```