Breakout / Lab 07

Return of the Curses

Continuation of Lab 5

Last Updated: February 28, 2017

Problem / Exercise

Write a C++ program that displays the contents of a file in *curses mode*. If the contents of the file are too big to fit on the screen, then your program needs to allow the user to scroll through the output using the *up* and *down* arrow keys.

Running Your Program

Assuming your executable is called `lab07`, you should be able to run your program as follows to open and display the contents of a file called `filename.txt`:

```
$ ./lab07 filename.txt
```

The `main` Function

For this program, you will need to use the following prototype for `main`:

```
int main(const int argc, const char * argv []);
```

Details about this `main` prototype will be covered by the TAs upon request. You may also find more information at `cppreference.com` ↗ .

References

You may find the following reference materials useful:

- `https://www.gnu.org/software/ncurses/`
- `http://tldp.org/HOWTO/NCURSES-Programming-HOWTO/`
- Ch. 5 & 8.4 in Hoover. "System Programming with C and UNIX" (1st Ed.) (ISBN-13: 9780136067122)
- Ch. 3 in Stevens & Rago. "Advanced Programming in the UNIX Environment" (3rd Ed.) (ISBN-13: 9780321637734)

1 Group Brainstorm

You are NOT allowed to use the computers during this time.

Breakup into groups based on your seating and brainstorm about how to solve the problem or exercise. Make sure everyone understands the problem, and sketch out potential ways to move towards a solution. Perhaps something that was discussed during lecture might be useful?

2 Submit Individual Brainstorm

You may use a computer from this point forward.

Login to eLC and submit a version of your group's brainstorm, written in your own words. You may add additional information if you want. You need to write enough in order to convince the grader that you understand the problem or exercise and that you have a plan for moving forward towards a solution. Please include the last names of the other people in your group in your submission. The brainstorm submission should be available on eLC in your assignment dropbox. You should submit your individual brainstorms before the end of your breakout period. **NOTE:** Submissions that do not include an individual brainstorm will not be graded.

3 Some Nonfunctional Requirements

Your submission needs to satisfy the following nonfunctional requirements:

- **Directory Setup:** Make sure that all of your files are in a directory called `LastName-FirstName-lab07`, where `LastName` and `FirstName` are replaced with your actual last name and first name, respectively.
- **Documentation:** All classes, structs, and functions must be documented using Javadoc (or Doxygen) style comments. Use inline documentation, as needed, to explain ambiguous or tricky parts of your code.
- **`Makefile`:** You need to include a `Makefile`. Your `Makefile` needs to compile and link separately. That is, make sure that your `Makefile` is setup so that your `.cpp` files each compile to individual `.o` files. This is very important. The expectation is that the grader should be able to type `make clean` and `make` to clean and compile/link your submission, respectively.
- **Standards & Flags:** Make sure that when you compile, you pass the following options to `g++` in addition to the `-c` option:

  ```
  -Wall -std=c++14 -g -O0 -pedantic-errors
  ```

  Other compiler/linker options may be needed in addition to the ones mentioned above.
- **`README`:** Make sure to include a `README` file that includes the following information presented in a reasonably formatted way: i) your Name and 810/811; ii) instructions on how to compile and run your program; and iii) reflection section. Make sure that each line in your `README` file does not exceed 80 characters. Do not assume line-wrapping. Please manually insert a line break if a line exceeds 80 characters.
- **Compiler Warnings:** Since you should be compiling with both the `-Wall` and `-pedantic-errors` options, your code is expected to compile without `g++` issuing any warnings.

4 Submission

Before your next breakout lab session, you need to submit your code. You will still be submitting your project via nike. Make sure your work is on `nike.cs.uga.edu` in a directory called `LastName-FirstName-lab07`. From within the parent directory, execute the following command:

```
$ submit LastName-FirstName-lab07cs1730a
```

It is also a good idea to email a copy to yourself. To do this, simply execute the following command, replacing the email address with your email address:

```
$ tar zcvf LastName-FirstName-lab07.tar.gz LastName-FirstName-lab07
$ mutt -s "lab07" -a LastName-FirstName-lab07.tar.gz -- your@email.com < /dev/null
```