

7.6 - The Rest of Lab6 (10 points)

1. Be sure you have completed 7.4 - A Little Bit of Lab6. You also want to have read through the 7.3 slides about `grep` - the Linux command
2. Copy your `grep_one.c` program into `grep_two.c`

add to `grep_two.c` so that at a minimum it is able to report whether a string is or is not in a file, and if it is, **which line it is on** and **some or all of the content** of that line.

A possible implementation of `grep_two.c` (this is my implementation) would produce the following output:

```
./grep_two tring lab6.txt
Found string 'tring' on line 3 in string '- finds a string'
Found string 'tring' on line 4 in string '- finds a substring'
Found string 'tring' on line 7 in string '- reports when it did not
find the string'
```

```
./grep_two strang lab6.txt
Did not find string 'strang' in file lab6.txt
```

In my implementation, it is:

- letting you know what string it found
- letting you know what line it occurred on
- providing some context where it was found

The amount of context is up to you – it could be an entire line of the file or it could be just a portion of the line. Mine is a portion of the line.

Another possible implementation would be to just report that it was found and dump the contents of the whole line of the file:

```
./grep_two tring lab6.txt
Found string 'tring' on line 3:
- finds a string in a line of text
Found string 'tring' on line 4:
- finds a substring in a line of text
Found string 'tring' on line 7 in string
- reports when it did not find the string
```

```
./grep_two strang lab6.txt
Did not find string 'strang' in file lab6.txt
```

Please do not dump the contents of the entire **file** each time you find the string.

Update your `lab6.txt` to indicate what you finished and what your `grep_two.c` does, including anything you attempted but were unable to finish or debug. Also include whatever limitations / design decisions you made for your `grep_two.c`

Navigate to your `csci235` directory. Then `tar` and compress the lab directory with the linux command (from your `csci235` directory):

```
tar -cvzf lab6.2.tgz lab6
```

Turn in your code to moodle in `lab6.2.tgz`

It should have:

- `grep_two.c`
- `grep_one.c`
- `lab6.txt` (that has your name as the driver)

Since the design decisions for this are up to you, there is no autograder. Test your code with files in your directory – you can use `lab6.txt` as shown above.