Lab 3: I/O in C

Due October 18th, 2016

In this lab, you will learn about several different types of Input and Output in C.

The two most common types of I/O in a C program are **standard I/O** and **file I/O**. Standard I/O takes user input from the terminal and prints output to the same terminal. File I/O involves reading from and writing to files on disk. For this lab, you will write a fake login system in C. Your code should all be contained within a file called **fakelogin.c**. When it is run, your login program should do the following:

1. Read from a list of usernames and passwords stored in a file called **users.dat**. This file will be a text file consisting of a username on one line, followed by that user's password on the next line, repeating for the number of users desired. An example users.dat file would look like this:

<div align="center">

johndoe
password
janedoe
secret
someuser
123456

</div>

Both the username and the password lines can contain any character except the null terminator and the newline character as a valid part of the string.

2. Ask the user to log in by first prompting for a username, then a password. If the user enters an existing username(as defined by users.dat) followed by the correct password **for that user**, the program should print the message "Successfully logged in!" and exit. If the user enters an invalid username or password, the program should print the message "Login failed!" and exit.

Your lab submission must also have a **makefile** that is capable of compiling your program into an executable titled **fakelogin**. It must also be capable of removing this executable when the command ***make clean*** is run within your lab's directory.

Below are two example runs of the program using the users.dat file described above:

```
$ ./fakelogin
  Enter Username: johndoe
  Enter Password: password
  Successfully logged in!

$ ./fakelogin
  Enter Username: gfreeman
  Enter Password: hl3confirmed!
  Login failed!
```

Your lab will be graded according to the following table:

Successfully read from users.dat.	5
Handle valid logins correctly.	10
Handle invalid logins(username or password) correctly	10
TOTAL	25 pts

IMPORTANT NOTE: All of your labs are graded automatically by a grading script. While there is no part of the table above that grades for a Makefile, the grading script is written to expect a makefile in your lab directory. If there is no makefile present, the script will automatically give you a zero for this lab. The instructor **will not** accept regrade requests for labs that have no makefile.

If you have any questions, contact your instructor at `acryker@gmail.com`. All due dates are final unless you choose to use one or more of your late days. Your lab submission **must** consist of a single tarball containing all of the relevant files for your lab. Submission of single code files or anything other than a valid tarball file ending in *.tar* will result in a zero for that lab. Refer to the syllabus for more information.