CPSC 256 - Lecture 2

The goals of this lecture are to:

1. Fork my NetBeansProjects repository (this contains all of my code for this class)
2. Clone your new repository on the Ubuntu laptop in front of you
3. Edit the main.c program of the HelloWorld Project
4. Compile and execute this edited file from the command line
5. Push these changes back up to your newly forked repo

Step 1: Cloning my NetBeansProjects Repo

a) In Firefox, go to https://github.com/ and log into your account
b) In the search bar at the top left of the page, type in "user:brash99"
c) Click on the brash99/NetBeansProjects link
d) Click on "Fork" in the upper right corner

That's it! You now have your OWN COPY of all of my code.

Step 2: Clone your new repository on the Ubuntu laptop in front of you

a) Log into the machine ... username is XXXXXXXX@cnuadmin.cnu.edu ... password is your usual CNU password

b) Open a terminal window (click on the "9 dots" icon in the bottom left, and find the terminal app)

Aside: Some basic Unix commands:

```
whoami – gives your username
ls     – lists all of the files in the
            current directory
ls –altr   – lists all of the files, in long
            format, in reverse time order
pwd    – shows the present working directory
cd     – returns to your 'home' directory

cd ~/Documents  – takes you to the Documents directory
                    ("~" is a shorthand for your
```

```
                                    home directory)
```

c) From your home directory, clone your GitHub code repo

```
cd
git clone https://github.com/<githubusername>/
NetBeansProjects.git
```

where <githubusername> is your GitHub username (not necessarily the same as your local username!!!!)

If this step is successful, you should see that there is a new directory, under your home directory, called NetBeansProjects (check by doing 'ls -altr')

Step 3: Edit the main.c file of the HelloWorld project

a) Get to the correct directory

```
cd ~/NetBeansProjects/HelloWorld
ls –altr
```

b) Edit the main.c file

```
nano main.c
```

(We will discuss the changes to be made in class … Save the changes with CTRL-X)

Step 4: Compile and execute the program

```
gcc –c main.c (What did this do?  Check with ls –altr)
gcc –o myFirstProgram main.o (What did this do?
                                  Check with ls –altr)
./myFirstProgram (Did it work????)
```

Step 5: Push these changes back up to your newly forked repo

a) Check which files have changed/been added with:

```
git status
```

b) Remove the object file and executable, for cleanliness

```
rm main.o
rm myFirstProgram
git status
```

c) Commit the changes

```
git commit –am “Edits to HelloWorld/main.c”
```

d) Now comes the only cumbersome part … pushing the changes back to your GitHub repo on GitHub.com

(i) Go back to your GitHub account in Firefox. Click on Settings, in the pulldown menu under your account avatar in the upper right corner

(ii) On the left side menu, go down all the way to the bottom, and click on Developer Settings

(iii) On the subsequent left side menu, click in Personal Access Tokens

(iv) Click on Generate New Token … add a note to describe this token (anything you like, maybe CPSC256) … choose the expiration to be No Expiration … Click on EVERY radio button in the long list, and then finally click on Generate Token.

(v) On the next page, and this is very important, copy this newly generated token onto the clipboard!!!!

(vi) Save this token in a file:

```
cd ~/NetBeansProjects
nano myPersonalAccessToken (past the clipboard into
                            this file, and save it!)
git add myPersonalAccessToken
git commit –am “Add personal access token for repo”
git status
```

(vii) Push the changes

```
git push origin main
```

(You may or may not be asked for your GitHub username. You will definitely be asked for a password … paste the personal access token for the password!)

(viii) Go back to your GitHub account in Firefox, and check to see that the newly edited main.c actually got pushed!