How to manage code with git

This assumes you have completed the github setup (described separately).

All submissions for the class will be done via git and github.com.

At the deadline we will run a script which clones the current state of all student homework repositories. **Ensure that at the deadline the head of branch “main” points to the project state you want us to grade.** This is the default behavior if you only use “git commit” and “git push” but remember you have to push or your changes stay local and we can’t see them! You can push as many times as you like, we will grade the version in your repository at the due date and time. PLEASE REMEMBER TO PUSH, not just commit.

Linux, OSX, *nix Instructions

Get your local clone:

```
git clone https://github.com/ut314-f2023/prog1-(githubid).git
```

We will seed all of the repositories with a README file, but you need to fill it out with this information.

1. Your name
2. Your EID
3. Your email
4. Group member names and email (if it is a group project, which this homework is not)
5. How much time this project took you to complete
6. If you are using any slip days (if you are turning in the assignment late)
7. Any comments to the grader

When you modify the file you will have to add its changes and then commit and push them.

Commit your changes

```
git commit -m “initial commit”
```

Push your changes

```
git push
```

Done! Remember to add, commit and push further changes, especially when you are done. Your changes stay local to your laptop (or desktop, or lab machine, etc.) until you push. You must push your code before the deadline for us to see it.

Windows Instructions

You can use Cygwin, wsl, or another command line utility in Windows to do it the Linux way.

Or use the git GUI initialize your submission repository using this path and create your local clone:

```
git clone https://github.com/ ut314-f2023/prog1-(githubid).git
```

This will clone a folder named prog1-(githubid) (you will have to choose the location).   Deal with the README and Android studio files as described above.

Next, you need to commit. Press the commit button and add "initial commit" to the message. Press commit.

Finally, you need to push. In the menus, go to repository > push

Done! Remember to commit and push further changes.

Validating your repository

Students often get nervous about whether their push "worked." You can visit the github website and look at your code to make sure that your push was successful. If you see the code, then we will too.

If you want, create a new directory and check out your code into that directory.