

# CS 351 Lab #0 Part 1 - Linux & Tools

Alexis Edwards

A20455965

## 1. Login to Fourier

```
alex ~ aedwards4@fourier ~ ssh aedwards4@fourier.cs.iit.edu — 80x43
Last login: Thu Sep 3 13:29:37 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run 'chsh -s /bin/zsh'.
For more details, please visit https://support.apple.com/kb/HT208850.
alex@fourier:~$ ssh aedwards4@fourier.cs.iit.edu
The authenticity of host 'fourier.cs.iit.edu (216.47.157.250)' can't be established.
RSA key fingerprint is SHA256:0K9vT63urA5/cjK5pVd76Shjd4Kqz0a+NQ3Q2WJkQ.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'fourier.cs.iit.edu,216.47.157.250' (RSA) to the list of known hosts.
aedwards4@fourier.cs.iit.edu's password:
Permission denied, please try again.
aedwards4@fourier.cs.iit.edu's password:
Last login: Tue Sep 1 11:45:57 2020 from 145.288.30.136.in-addr.arpa
=====
This system is for the use of authorized users only. Individuals using
this computer system without authority, or in excess of their authority,
are subject to having all of their activities on this system monitored
and recorded by system personnel. In the course of monitoring this
system, or in the course of system maintenance, the activities of
authorized users may also be monitored. Anyone using this system
expressly consents to such monitoring and is advised that if such
monitoring reveals possible evidence of criminal activity, system
personnel may provide the evidence of such monitoring to law enforcement
officials. All users of this system must abide by the acceptable user
policies of the Office of Technology Services and the Illinois Institute
of Technology.

Visit http://www.iit.edu/ots/our_policies.shtml to read the OTS and IIT policies
.
=====
Reminder to all Fourier Users: Kindly delete all unneeded data from your home di
rectory !!!

[aedwards4@fourier ~]$
```

## 2. Attach tmux session

```
[aedwards4@fourier ~]$ tmux at
no sessions
[aedwards4@fourier ~]$ tmux
[detached (from session 0)]
[aedwards4@fourier ~]$ tmux at
[detached (from session 0)]
[aedwards4@fourier ~]$
```

## 3. Clone github repository

```
[aedwards4@fourier ~]$ git clone https://gitlab.com/cs351-fall2020/student-77
Cloning into 'student-77'...
Username for 'https://gitlab.com': aedwards4
Password for 'https://aedwards4@gitlab.com':
remote: Enumerating objects: 24, done.
remote: Counting objects: 100% (24/24), done.
remote: Compressing objects: 100% (22/22), done.
remote: Total 24 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (24/24), done.
Checking connectivity... done.
[aedwards4@fourier ~]$
```

## 4. Signing the README

```
alex ~ aedwards4@fourier ~ ssh aedwards4@fourier.cs.iit.edu — 80x43
# CS 351 Code Repository

Alexis Edwards
aedwards4@hawk.iit.edu

[... many lines of README content ...]

*README.md* 4L, 64C                                     4,1      All
[... footer ...]
```

## 5. Committing the new changes

```
[aedwards4@fourier student-77]$ git commit -am "Signing honor pledge"
[master 9100fc2] Signing honor pledge
Committer: Edwards Alexis <aedwards4@fourier.cs.iit.edu>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 2 insertions(+), 6 deletions(-)
[aedwards4@fourier student-77]$
```

## 6. Pushing the new changes

```
[aedwards4@fourier student-77]$ git commit -am "Signing honor pledge"
[master 9100fc2] Signing honor pledge
Committer: Edwards Alexis <aedwards4@fourier.cs.iit.edu>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 2 insertions(+), 6 deletions(-)
[aedwards4@fourier student-77]$ git push
warning: push.default is unset; its implicit value is changing in
Git 2.8 from 'matching' to 'simple'. To squelch this message
and maintain the current behavior after the default changes, use:

    git config --global push.default matching

To squelch this message and adopt the new behavior now, use:

    git config --global push.default simple

When push.default is set to 'matching', git will push local branches
to the remote branches that already exist with the same name.

In Git 2.8, Git will default to the more conservative 'simple'
behavior, which only pushes the current branch to the corresponding
remote branch that 'git pull' uses to update the current branch.

See 'git help config' and search for 'push.default' for further information.
(the 'simple' mode was introduced in Git 1.7.11. Use the similar mode
'current' instead of 'simple' if you sometimes use older versions of Git)

Username for 'https://gitlab.com': aedwards4@hawk.iit.edu
Password for 'https://aedwards4@hawk.iit.edu@gitlab.com':
Counting objects: 5, done.
Delta compression using up to 12 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 386 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://gitlab.com/cs351-fall12020/student-77
 88debbe..9100fc2 master -> master
[aedwards4@fourier student-77]$
```

## 7. Try to compile main.c

```
lex — aedwards4@fourier:~ — ssh aedwards4@fourier.cs.iit.edu — 80x25
[aedwards4@fourier lab0]$ ls
hello.c hello.h main.c Makefile Part 1
[aedwards4@fourier lab0]$ gcc main.c
/tmp/ccFaUMXm.o: In function 'main':
main.c:(.text+0x24): undefined reference to `say_hello_to'
main.c:(.text+0x30): undefined reference to `say_hello_to'
collect2: error: ld returned 1 exit status
[aedwards4@fourier lab0]$
```

## 8. Try to compile hello.c

```
lex — aedwards4@fourier:~ — ssh aedwards4@fourier.cs.iit.edu — 80x25
[aedwards4@fourier lab0]$ gcc hello.c
/usr/lib/../lib64/crt1.o: In function '_start':
(.text+0x20): undefined reference to 'main'
collect2: error: ld returned 1 exit status
[aedwards4@fourier lab0]$
```

## 9. Compiling them together

```
lex — aedwards4@fourier:~ — ssh aedwards4@fourier.cs.iit.edu — 80x25
[aedwards4@fourier lab0]$ gcc hello.c main.c
[aedwards4@fourier lab0]$ ls
a.out hello.c hello.h main.c Makefile Part 1
[aedwards4@fourier lab0]$
```

## 10. Testing

```
[aedwards4@fourier lab0]$ ./a.out
Hello world!
[aedwards4@fourier lab0]$ ./a.out Master
Hello Master!
[aedwards4@fourier lab0]$
```

11. Renaming and allowing compiler to store in correct place Building intermediate object files and linking them together in separate steps

```
lex — aedwards4@fourier:~ — ssh aedwards4@fourier.cs.iit.edu — 80x25
[aedwards4@fourier lab0]$ rm a.out
[aedwards4@fourier lab0]$ gcc -o hello hello.c main.c
[aedwards4@fourier lab0]$ ./hello Overlord
Hello Overlord!
[aedwards4@fourier lab0]$
```

12. Building intermediate object files and linking them together in separate steps

```
lex — aedwards4@fourier:~ — ssh aedwards4@fourier.cs.iit.edu — 80x25
[aedwards4@fourier lab0]$ gcc -c hello.c
[aedwards4@fourier lab0]$ gcc -c main.c
[aedwards4@fourier lab0]$ gcc -o hello hello.o main.o
[aedwards4@fourier lab0]$ ls
hello  hello.c  hello.h  hello.o  main.c  main.o  Makefile  Part 1
[aedwards4@fourier lab0]$
```

13. Using make to automate builds

```
lex — aedwards4@fourier:~ — ssh aedwards4@fourier.cs.iit.edu — 80x25
[aedwards4@fourier lab0]$ rm -f *.o hello
[aedwards4@fourier lab0]$ make
gcc -g -Wall -O2 -c -o hello.o hello.c
gcc -g -Wall -O2 -c -o main.o main.c
gcc -g -Wall -O2 -o hello hello.o main.o
[aedwards4@fourier lab0]$
```

14. Trying again

```
[aedwards4@fourier lab0]$ make
make: Nothing to be done for `all'.
[aedwards4@fourier lab0]$
```

15. Updating timestamp to force a rebuild

```
lex — aedwards4@fourier:~ — ssh aedwards4@fourier.cs.iit.edu — 80x25
[aedwards4@fourier lab0]$ touch hello.c
[aedwards4@fourier lab0]$ make
gcc -g -Wall -O2 -c -o hello.o hello.c
gcc -g -Wall -O2 -o hello hello.o main.o
[aedwards4@fourier lab0]$
```

16. Run a test, then clean up

```
lex — aedwards4@fourier:~ — ssh aedwards4@fourier.cs.iit.edu — 80x25
[aedwards4@fourier lab0]$ make test
Running test...
./hello tester
Hello tester!
[aedwards4@fourier lab0]$ make clean
rm -f hello.o main.o hello
[aedwards4@fourier lab0]$
```