

What's up brahs?



Aiden Cox - CEG 2350 Lab Lead

Austin Kellough - CEG 2350 Lab Assistant

Quote of the week:

“Everyday might not be a good day but there’s something good in everyday”

How was Midterm and Lab09?

```
top - 07:25:42 up 94 days, 18:52, 1 user, load average: 0.02, 0.02, 0.00
task: 126 total, 1 running, 125 sleeping, 0 stopped, 0 zombie
Meminfo: 0.0 us, 0.0 sy, 0.0 ri, 99.7 fr, 0.0 sw, 0.0 hi, 0.0 si, 0.0 st
Mem Mem: 1087.7 total, 286.5 free, 247.7 used, 1039.0 buff/cache
Mem Swap: 0.0 total, 0.0 free, 0.0 used, 0.0 swap Mem

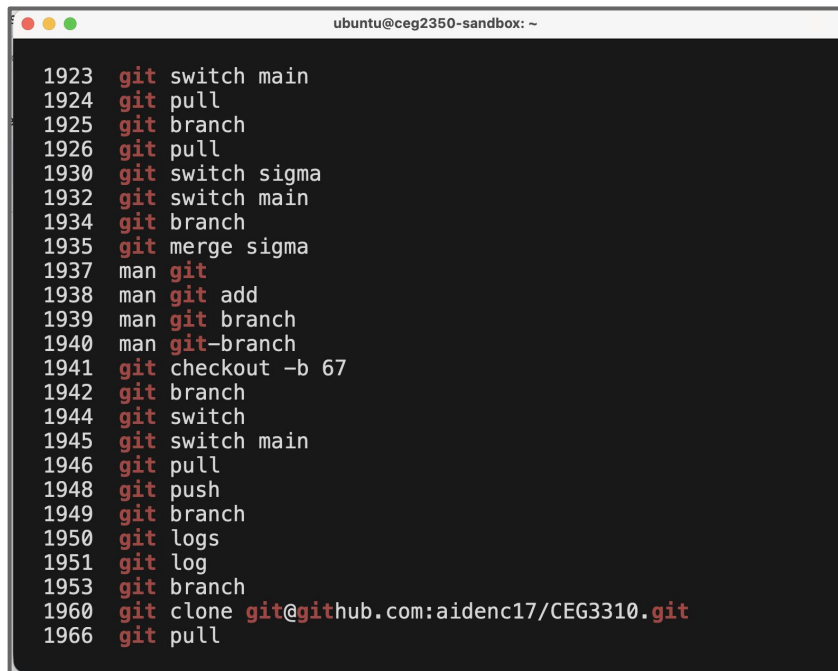
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
1 root 20 0 167312 13144 8866 S 0.0 0.0 2:40.93 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.50 kthreadd
3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_gp
4 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_p
6 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/0/0H-kblockd
7 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 mgfsd
10 root 20 0 0 0 S 0.0 0.0 1:24.00 ksoftirqd/0
11 root 20 0 0 0 S 0.0 0.0 3:14.64 rcu_sched
12 root 21 0 0 0 S 0.0 0.0 0:04:56 migration/0
13 root -51 0 0 0 S 0.0 0.0 0:00.00 idle_inject/0
15 root 20 0 0 0 S 0.0 0.0 0:00.00 cpufreqd
16 root 20 0 0 0 S 0.0 0.0 0:00.00 kdevtmpfs
18 root 0 -20 0 0 I 0.0 0.0 0:00.00 netns
17 root 20 0 0 0 S 0.0 0.0 0:00.00 rcu_tasks_kthre
18 root 20 0 0 0 S 0.0 0.0 0:00.00 kauditd
19 root 20 0 0 0 S 0.0 0.0 0:03:17 khungtaskd
20 root 20 0 0 0 S 0.0 0.0 0:00.00 ksm, kswapd
21 root 0 -20 0 0 I 0.0 0.0 0:00.00 writeback
22 root 20 0 0 0 S 0.0 0.0 0:00.00 ksmcgattd
23 root 25 5 0 0 S 0.0 0.0 0:00.00 ksm
24 root 30 15 0 0 S 0.0 0.0 0:11:51 kswapd0
116 root 0 -20 0 0 I 0.0 0.0 0:00.00 kintegrityd
117 root 0 -20 0 0 I 0.0 0.0 0:00.00 kblockd
118 root 0 -20 0 0 I 0.0 0.0 0:00.00 bling_pont_bio
119 root 0 -20 0 0 I 0.0 0.0 0:00.00 tps_dev_wq
120 root 0 -20 0 0 I 0.0 0.0 0:00.00 ata_sff
121 root 0 -20 0 0 I 0.0 0.0 0:00.00 md
122 root 0 -20 0 0 I 0.0 0.0 0:00.00 edac-poller
123 root 0 -20 0 0 I 0.0 0.0 0:00.00 devfreq_wq
124 root 21 0 0 0 S 0.0 0.0 0:00.00 watchdog
127 root 20 0 0 0 S 0.0 0.0 0:29:16 kswapd0
130 root 0 -20 0 0 S 0.0 0.0 0:00.00 scsi_scsi-athre
131 root 0 -20 0 0 I 0.0 0.0 0:00.00 kthreadd
132 root 0 -20 0 0 I 0.0 0.0 0:00.00 scsi_thermal_pm
133 root 0 -20 0 0 I 0.0 0.0 0:00.00 vfio-ttcr-lua
134 root 0 -20 0 0 I 0.0 0.0 0:00.00 ipv6_addrconf
140 root 0 -20 0 0 I 0.0 0.0 0:00.00 kworker/0:0
141 root 0 -20 0 0 I 0.0 0.0 0:00.00 httpd
142 root 0 -20 0 0 I 0.0 0.0 0:00.00 charger_manager
286 root 20 0 0 0 S 0.0 0.0 0:00.00 scsi_ah_0
286 root 0 -20 0 0 S 0.0 0.0 0:00.00 scsi_ah_1
286 root 20 0 0 0 S 0.0 0.0 0:00.00 scsi_ah_1
286 root 0 -20 0 0 S 0.0 0.0 0:00.00 scsi_ah_2
286 root 0 -20 0 0 I 0.0 0.0 0:01:36 kworker/0/1H-kblockd
210 root 20 0 0 0 S 0.0 0.0 0:00.00 rcu_gp_2
211 root 0 -20 0 0 I 0.0 0.0 0:00.00 scsi_tmr_2
212 root 20 0 0 0 S 0.0 0.0 0:00.00 scsi_ah_3
```

```
~ > ps au 06:25:59 PM
USER      PID %CPU %MEM    VSZ   RSS  TT  STAT STARTED
TIME COMMAND
aidencox  840  1.8  0.0 435314448 6432 s000 S Thu03PM
0:05.03 -zsh
root      5565  0.0  0.0 435299424 1824 s000 R+ 6:26PM
0:00.01 ps au
aidencox  1052  0.0  0.0 410763248 3872 s000 S Thu03PM
0:00.94 /opt/homebrew/Ce
aidencox  1051  0.0  0.0 435310368 1104 s000 S Thu03PM
0:00.24 -zsh
aidencox  1049  0.0  0.0 435310384 880 s000 S Thu03PM
0:00.00 -zsh
aidencox  853  0.0  0.0 435305392 976 s000 S Thu03PM
0:00.01 -zsh
root      838  0.0  0.0 435308576 6048 s000 Ss Thu03PM
0:00.02 /usr/bin/login -
```

Processes and system-y things?
Midterm questions?

Second AWS Check-In!

1. Ability to connect to AWS instance
2. Command history relevant to Lab 09 and / or 10
3. At least one branch in GitHub repository

A terminal window titled 'ubuntu@ceg2350-sandbox: ~' with a dark background and light-colored text. It displays a list of git commands with line numbers on the left. The commands include switching branches, pulling, pushing, merging, and cloning a repository from GitHub.

```
1923 git switch main
1924 git pull
1925 git branch
1926 git pull
1930 git switch sigma
1932 git switch main
1934 git branch
1935 git merge sigma
1937 man git
1938 man git add
1939 man git branch
1940 man git-branch
1941 git checkout -b 67
1942 git branch
1944 git switch
1945 git switch main
1946 git pull
1948 git push
1949 git branch
1950 git logs
1951 git log
1953 git branch
1960 git clone git@github.com:aidenc17/CEG3310.git
1966 git pull
```

Check off in-person (with your instructor or TAs)

Beginning Lab10

Lab Instructions: <https://pattonsgirl.github.io/CEG2350/Labs/Lab10/Instructions>

Lab Template:

<https://raw.githubusercontent.com/pattonsgirl/CEG2350/refs/heads/main/docs/Labs/Lab10/LabTemplate.md>

Feeling comfortable with the course?
Questions from lecture?

Issues and Branches - Part 0

Part 0 - Issues & Branches

Useful git commands: branch, switch, commit, push, merge

1. To begin, create and add the Lab Template to your `main` branch in your `Lab10` folder
2. In your GitHub repository, go to the `Issues` tab. Create an Issue **for each part** of this lab - you may do this all at once or as you start each part.
3. Through this lab, you will create **one** branch **for each** issue.
4. Merge the branch with the `main` branch as your **complete each part**. You may then mark your issue as complete.
5. If you make **adjustments after merging** you need to create an Issue describing what needs correcting, then resolve the issue.

One merge must be completed via a **Pull Request** - all others should be done via branch merging on the command line.

Make sure that **EACH PART** of lab has an issue, a branch, and that each one gets **MERGED** via command line, or a Pull Request.

My Girlfriend Says I have a lot of Issues? - Part 0

The screenshot shows the GitHub interface for the repository 'pattonsgirl / CEG2350'. The 'Issues' tab is selected and highlighted with a black box. In the top right, the 'New issue' button is also highlighted with a black box. The search bar contains the text 'is:issue state:open'. Below the search bar, there are filters for 'Open' (17) and 'Closed' (14). A list of issues is displayed, each with a checkbox, a green circle icon, a title, and a description. A black box with the text 'Our course Repo's Issues!' is overlaid on the right side of the issue list.

pattonsgirl / CEG2350

Search: Type to search

<> Code **Issues 17** Pull requests Actions Projects Wiki Security Insights Settings

is:issue state:open

Labels Milestones **New issue**

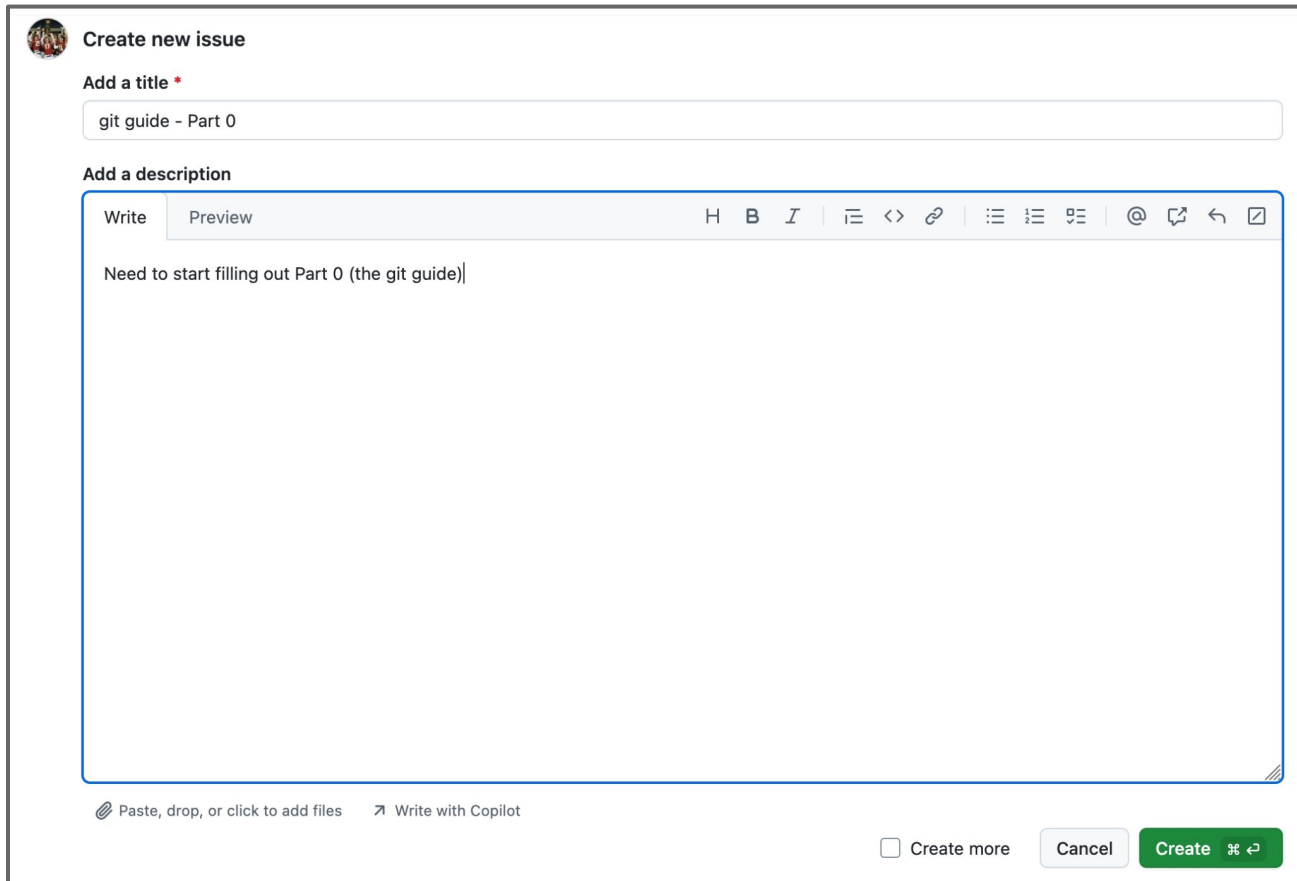
☐ **Open 17** Closed 14

Author Labels Projects Milestones Assignees Newest

- ☐ **Lab 08 - unique instance?**
#54 · pattonsgirl opened 2 weeks ago
- ☐ **Lab 04 - Add Main to bash?**
#47 · aidenc17 opened on Sep 26
- ☐ **lab 02 clarification / wording issue**
#45 · b-wens-kedar opened on Sep 11
- ☐ **Grading & GH Issues**
#43 · pattonsgirl opened on Sep 5
- ☐ **Lab 12 - Citations used for Part 5 (need some bold letters)**
#41 · aidenc17 opened on Apr 26
- ☐ **Lab 11 - Extract and Profit Permissions**
#40 · aidenc17 opened on Apr 21
- ☐ **Exam 2 Missing**
#38 · pattonsgirl opened on Apr 4
- ☐ **Lab 06 - relative pathing or use ENV variable**
#25 · pattonsgirl opened on Mar 11

Our course Repo's Issues!

My Girlfriend Says I have a lot of Issues? - Part 0



The screenshot shows the GitHub 'Create new issue' interface. At the top left is a profile picture and the text 'Create new issue'. Below this is a section 'Add a title' with a red asterisk, containing a text input field with the value 'git guide - Part 0'. Underneath is the 'Add a description' section, which has a 'Write' tab and a 'Preview' tab. The 'Write' tab is active, showing a rich text editor with a toolbar containing icons for bold, italic, link, list, and other formatting options. The text area contains the text 'Need to start filling out Part 0 (the git guide)|'. At the bottom of the form, there is a link to 'Paste, drop, or click to add files', a link to 'Write with Copilot', and three buttons: 'Create more' (disabled), 'Cancel', and 'Create' (green button with a keyboard shortcut icon).

Create new issue

Add a title *



git guide - Part 0

Add a description

Write Preview

H B I | | < > | | | | | @ ↗ ↶ ↷

Need to start filling out Part 0 (the git guide)|

 Paste, drop, or click to add files  Write with Copilot

☐ Create more Cancel Create ⌘ ↵

You will end up making an issue for each part of the lab!

The button for this is found under issue, and new issue on the top right.

After creating the issue, the next slide shows what it looks like to click back into it.

My Girlfriend Says I a have lot of Issues? - Part 0

The screenshot shows a GitHub issue page for "git guide - Part 0 #2". The issue title is highlighted with a box and an arrow. The issue description, "Need to start filling out Part 0 (the git guide)", is also highlighted with a box and an arrow. The "Close issue" button at the bottom right is highlighted with a box and an arrow. The "Add a comment" section is visible, showing a text input area and a "Comment" button. The right sidebar contains metadata for the issue, including assignees, labels, type, projects, milestone, and relationships.

git guide - Part 0 #2

Issue title

Open

aidenc17 opened 22 minutes ago

Need to start filling out Part 0 (the git guide)

Issue description

Create sub-issue

Add a comment

Write Preview

Use Markdown to format your comment

Button to close issues

Close issue

Comment

No one - [Assign yourself](#)

Labels

No labels

Type

No type

Projects

No projects

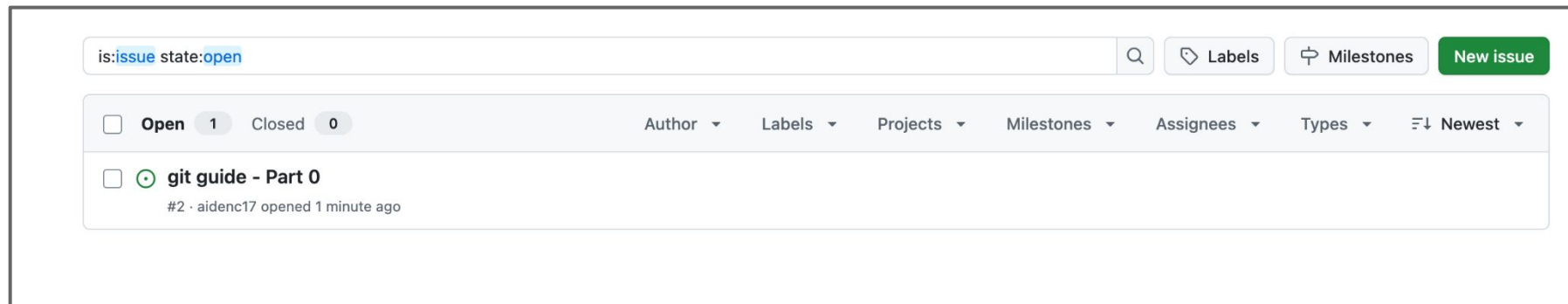
Milestone

No milestone

Relationships

None yet

My Girlfriend Says I have a lot of Issues? - Part 0



After clicking create, you'll go to the main issue page. Your's will all be populated with your grades for the labs, but you can catch my drift!

My Girlfriend Says I have a lot of Issues? - Part 0

WSU-kduncan / ceg2350f25-aidenc17

Q Type / to search

+

<> Code Issues 6 Pull requests Actions Projects Wiki Security Insights Settings

is:issue state:open

Q

Labels


Milestones

New issue


☐ Open 6

Closed 0


Author Labels Projects Milestones Assignees Types Newest

☐  makey makey - Part 5


#7 · aidenc17 opened 2 minutes ago

☐  document - Part 4


#6 · aidenc17 opened 2 minutes ago

☐  ignore that - Part 3


#5 · aidenc17 opened 3 minutes ago

☐  compiling - Part 2

#4 · aidenc17 opened 5 minutes ago

☐  common compilers - Part 1

#3 · aidenc17 opened 7 minutes ago

☐  git guide - Part 0

#2 · aidenc17 opened 38 minutes ago

What my repo looks like after creating all the issues

Branches (But not tree ones. Well, kinda) - Part 0

On the course repo, make a branch!

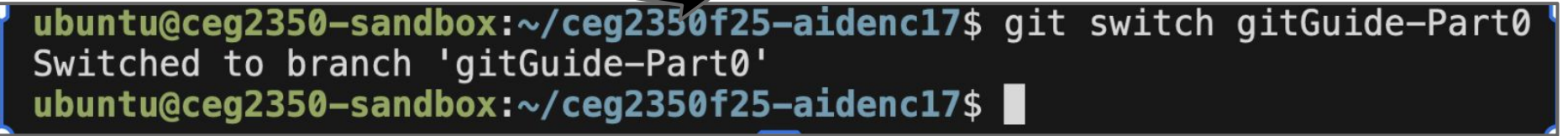


ubuntu@ceg2350-sandbox: ~/ceg2350f25-aidenc17

The terminal window has a title bar with three colored circles (red, yellow, green) on the left. A black arrow points from the instruction box above to the terminal. The terminal shows the command 'git branch gitGuide-Part0' being executed.

```
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17$ git branch gitGuide-Part0
```

Swap to new branch!



The terminal window shows the command 'git switch gitGuide-Part0' being executed, followed by the output 'Switched to branch 'gitGuide-Part0''. A black arrow points from the instruction box above to the terminal. The prompt is now 'ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17\$'.

```
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17$ git switch gitGuide-Part0  
Switched to branch 'gitGuide-Part0'  
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17$
```

Do this for each of your issues, (all of the parts) and commit to each of those branches. When complete, merge branches. One **MUST** be done using a Pull Request (PR)

Branches (But not tree ones. Well, kinda) - Part 0

When trying to push from a new branch, you will get an error message!

```
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17$ git push
fatal: The current branch compiling-Part2 has no upstream branch.
To push the current branch and set the remote as upstream, use
```

```
git push --set-upstream origin compiling-Part2
```

```
To have this happen automatically for branches without a tracking
upstream, see 'push.autoSetupRemote' in 'git help config'.
```

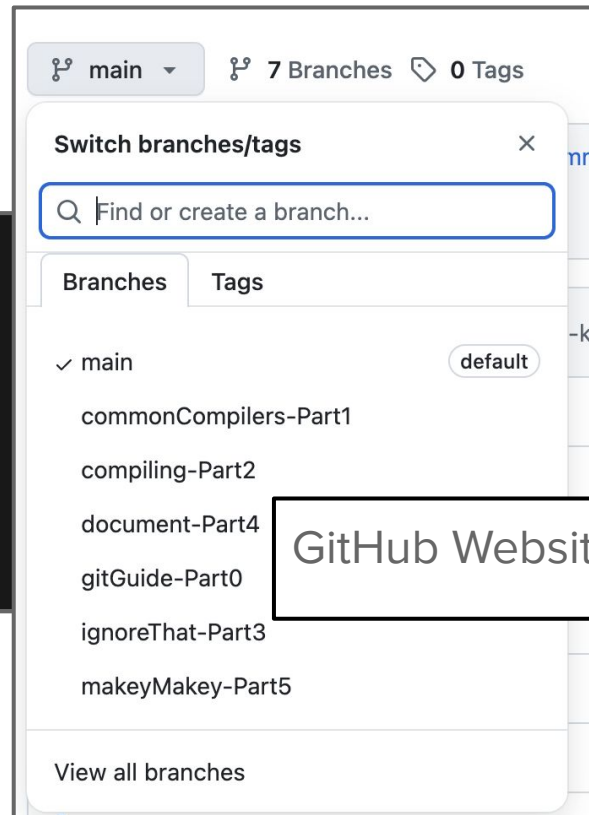
That's expected! Just run the command it tells you to and life will be good!

Branches (But not tree ones. Well, kinda) - Part 0

What my repo looks like after the branches are made:

```
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17$ git branch
commonCompilers-Part1
compiling-Part2
document-Part4
gitGuide-Part0
ignoreThat-Part3
* main
makeyMakey-Part5
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17$
```

Inside my AWS instance / Command Line



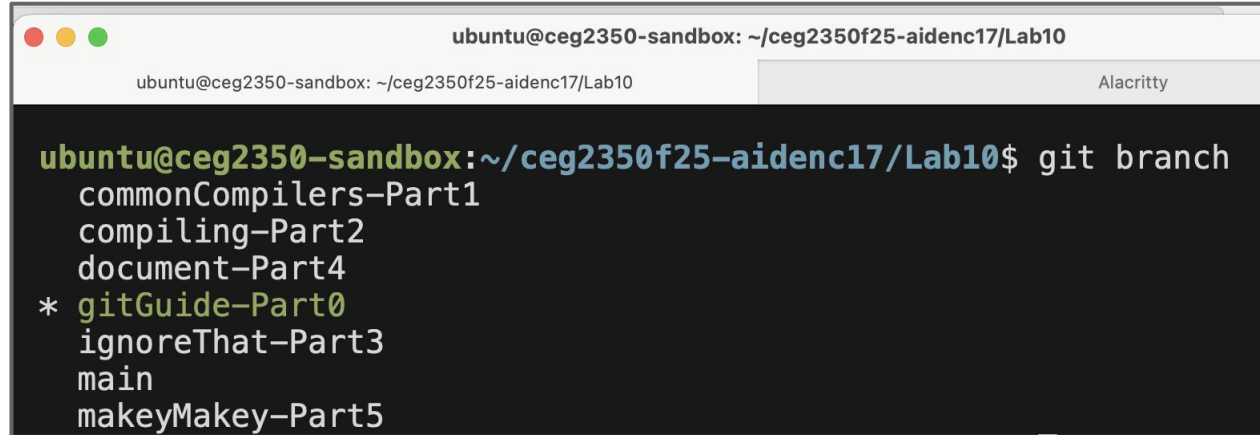
GitHub Website

Merging (But Not as Bad as Merge Sort) - Part 0

After making changes to a branch, the changes only pertain to that branch. Swapping to the Main branch will contain the changes that were made before branching.

To bring the changes over from our Part0 branch, (and the rest of the branches for this lab), you will need to do a `git merge`.

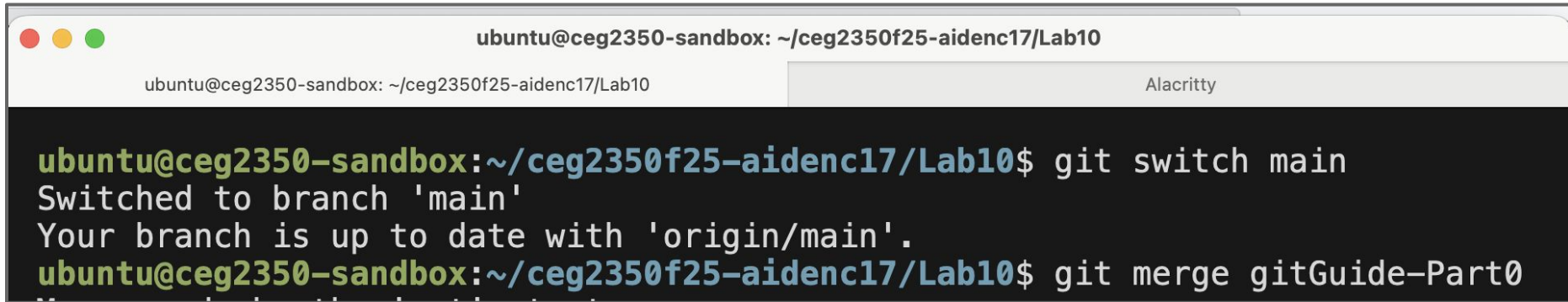
One of them will be a Pull Request, more on that later

A terminal window screenshot from an Ubuntu machine. The title bar shows 'ubuntu@ceg2350-sandbox: ~/ceg2350f25-aidenc17/Lab10'. The terminal prompt is 'ubuntu@ceg2350-sandbox: ~/ceg2350f25-aidenc17/Lab10\$'. The command 'git branch' has been executed, and the output lists several branches: 'commonCompilers-Part1', 'compiling-Part2', 'document-Part4', '* gitGuide-Part0' (which is the current branch, indicated by an asterisk), 'ignoreThat-Part3', 'main', and 'makekeyMakekey-Part5'.

```
ubuntu@ceg2350-sandbox: ~/ceg2350f25-aidenc17/Lab10
ubuntu@ceg2350-sandbox: ~/ceg2350f25-aidenc17/Lab10$ git branch
commonCompilers-Part1
compiling-Part2
document-Part4
* gitGuide-Part0
ignoreThat-Part3
main
makekeyMakekey-Part5
```

We want to be on the branch that we are merging INTO. Swap on over to Main!

Merging (But Not as Bad as Merge Sort) - Part 0

A terminal window titled 'ubuntu@ceg2350-sandbox: ~/ceg2350f25-aidenc17/Lab10' with a tab labeled 'Alacritty'. The terminal shows the following commands and output:


```
ubuntu@ceg2350-sandbox: ~/ceg2350f25-aidenc17/Lab10$ git switch main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
ubuntu@ceg2350-sandbox: ~/ceg2350f25-aidenc17/Lab10$ git merge gitGuide-Part0
```

Congrats! You have successfully merged (moved)
your changes from the Part0 branch into Main!


Next on how to do a Pull Request

Pull That Shit In - Part 0

After making changes to a branch, you may see the big scary green button on GitHubs website for your course repo. You will need to use this button for ONE of your merges.


 **ceg2350f25-aidenc17** Private Edit Pins Watch 0


forked from [WSU-kduncan/ceg2350-f25-ceg2350f25-CEG2350-LabTemplate](#)

 **gitGuide-Part0** had recent pushes 2 seconds ago Compare & pull request

main 7 Branches 0 Tags t Add file <> Code

This branch is 38 commits ahead of and 50 commits behind [WSU-kduncan/ceg2350-f25-ceg2350f25-CEG2350-LabTemplate:main](#) .
Contribute Sync fork



 **aidenc17** git ignore updated 9e0b8aa · 5 minutes ago 39 Commits


 Browns added packer file 2 months ago

Pull That Shit In - Part 0

Comparing changes


Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#) or [learn more about pull requests](#).


 base: main  compare: gitGuide-Part0 ✓ **Able to merge.** These branches can be automatically merged.



H B I

merged in via accepting this PR

 Markdown is supported

 Paste, drop, or click to add files

Create pull request

Notice we are going to now “merge” or bring our changes from our Part0 branch over to our main branch.

“Able to merge” means there aren’t any merge conflicts going to arise. Perfect!

Pull That Shit In - Part 0

The screenshot shows a GitHub pull request interface. At the top, the repository is 'WSU-kduncan / ceg2350f25-aidenc17'. The navigation bar includes 'Code', 'Issues (6)', 'Pull requests (1)', 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. The pull request title is 'did part 0 #8'. A green 'Open' button is next to the text 'aidenc17 wants to merge 1 commit into main from gitGuide-Part0'. Below this, there are tabs for 'Conversation (0)', 'Commits (1)', 'Checks (0)', and 'Files changed (1)'. A comment from 'aidenc17' is visible, stating 'part 0 is now complete and will be merged in via accepting this PR'. At the bottom, a green box indicates 'No conflicts with base branch' and 'Merging can be performed automatically.' with a 'Merge pull request' button. A footer note says 'Still in progress? Convert to draft'.

WSU-kduncan / ceg2350f25-aidenc17

<> Code Issues 6 Pull requests 1 Actions Projects Wiki Security Insights Settings

did part 0 #8

aidenc17 wants to merge 1 commit into `main` from `gitGuide-Part0`

Conversation 0 Commits 1 Checks 0 Files changed 1

aidenc17 commented now

part 0 is now complete and will be merged in via accepting this PR

did part 0

No conflicts with base branch
Merging can be performed automatically.

Merge pull request You can also merge this with the command line. [View command line instructions.](#)

Still in progress? Convert to draft

PR is now under the PR tab, and upon clicking the “merge pull request button”, main will have all of the changes that our branch had!

Compilery Things - Part 1

Part 1 - Compilers

Reminder - create an Issue for this part -> create a branch to work on this issue -> make your commits to this branch -> verify your changes (no broken markdown, good spelling) -> merge to main

Fill out the following charts regarding the three most commonly used languages - C/C++, Java, and Python. The bonus is you'll make a guide for yourself on how to install these or how to check if they are already available. I'm going to give you the answers for C / C++ - you'll need to complete the chart for Java and Python.

Fill out the tables with information on where to install the languages and "how to verify" - which / whereis and version (--version) on each Linux, Mac, and Windows.

Compilery Things - Part 1

```
ubuntu@ceg2350-sandbox: ~  
  
ubuntu@ceg2350-sandbox:~$ gcc --version  
gcc (Ubuntu 13.3.0-6ubuntu2~24.04) 13.3.0  
Copyright (C) 2023 Free Software Foundation, Inc.  
This is free software; see the source for copying conditions.  T  
here is NO  
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR  
PURPOSE.  
  
ubuntu@ceg2350-sandbox:~$ python3 --version  
Python 3.12.3  
ubuntu@ceg2350-sandbox:~$ javac --version  
javac 21.0.8  
ubuntu@ceg2350-sandbox:~$ █
```

Your AWS
instance is
Linux!

Compiling on the Command Line - Part 2

Part 2 - Compiling

Reminder - create an Issue for this part -> create a branch to work on this issue -> make your commit to this branch -> verify your changes (no broken markdown, good spelling) -> merge to main

Useful Commands: `gcc`, `g++`, `wget`

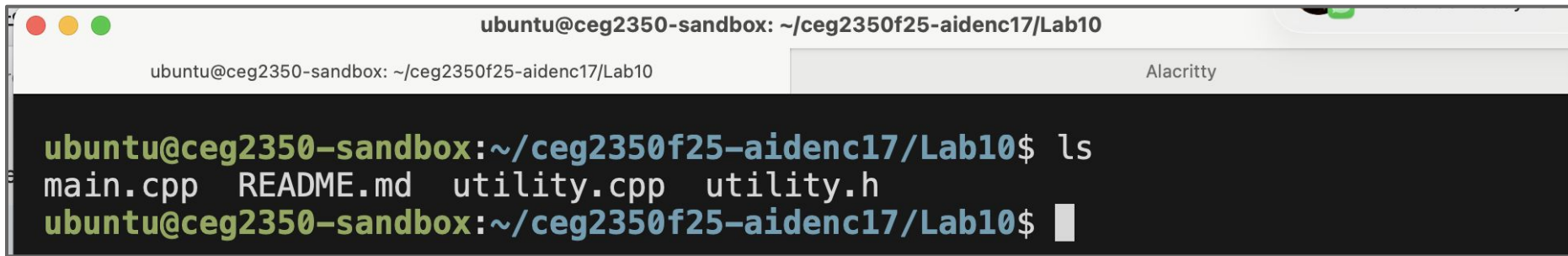
1. Select from either `TODO-C` or `TODO-CPP`. Download the files in the folder to the `Lab10` folder in your repository folder. `add`, `commit` and `push` the source code files.
2. Install the C / C++ compiler on your AWS instance.
3. Verify the installation of the C / C++ compiler on your AWS instance.
 - It is recommended to query for the version of the program to validate installation.
4. Using your AWS instance, compile the source code into an executable program.
5. Using your AWS instance, run the executable program.
6. Add the program source code file - `.c` or `.cpp` files - to your repository. Do not add the compiled executable for tracking.

Grab either the ``c`` or ``c++`` code from the course GitHub repo, and figure out a way to pull them onto your system!

Follow Part 1 on how to install compilers and verification of them.

Next slide for compilation process!

Compiling on the Command Line - Part 2

A terminal window titled 'ubuntu@ceg2350-sandbox: ~/ceg2350f25-aidenc17/Lab10' with the Alacritty window manager. The terminal shows the command 'ls' being executed, listing the files 'main.cpp', 'README.md', 'utility.cpp', and 'utility.h' in the current directory. The prompt is 'ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/Lab10\$' and the cursor is at the end of the line.

```
ubuntu@ceg2350-sandbox: ~/ceg2350f25-aidenc17/Lab10
ubuntu@ceg2350-sandbox: ~/ceg2350f25-aidenc17/Lab10
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/Lab10$ ls
main.cpp  README.md  utility.cpp  utility.h
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/Lab10$
```

Grab either the `c` or `c++` code from the course GitHub repo, and figure out a way to pull them onto your system!

Follow Part 1 on how to install compilers and verification of them.

Next slide for compilation process!

Compiling on the Command Line - Part 2

A terminal window with a title bar showing three colored circles (red, yellow, green) on the left and the text 'ubuntu@ceg2350-sandbox: ~/ceg2350f25-aidenc17/Lab10' on the right. The terminal has a dark background with light green text. It shows a sequence of commands and their outputs: a compilation command, a listing command, and the resulting files.

```
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/Lab10$ g++ main.cpp utility.cpp utility.h -o aidensProgram
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/Lab10$ ls
aidensProgram  main.cpp  README.md  utility.cpp  utility.h
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/Lab10$
```

Since I snagged the `c++` files, I used `g++` to compile. The `-o` specifies the output file, and the ARG after give the program the name.

Run the program like you would an executable with the executable bit!

.gitignore Me Like My Girlfriend Does - Part 3

aidenc17 git ignore updated		9e0b8aa · 17 hours ago	🕒 39 Commits
📁 Browns	added packer file		2 months ago
📁 Lab02	pasted template in		2 months ago
📁 Lab03	added lab 3 template		2 months ago
📁 Lab04	idk what i messed up		last month
📁 Lab05	idoit		3 weeks ago
📁 Lab07	Delete Lab07/00C5D5C1-2E02-4D1D-8AD3-31CFF98E1...		last month
📁 Lab08	idoit		3 weeks ago
📁 Lab10	test to see if i need to merge		last week
📁 Scripts	added loose files		2 months ago
📁 WeHateNano	i dont like using nano		2 months ago
📁 sample	Initial commit		3 months ago
📄 .gitignore	git ignore updated		17 hours ago
📄 KalliShoppingList	added her shopping list that she needs to grab ASAP		2 days ago
📄 README.md	Initial commit		3 months ago
📄 ThisWasMadeFromABranch	here is a commit on the new branch		last week

[illegible]

Put a file named `.gitignore` in the root of your repository! (Outside all of the Lab folders)

Inside the file, list files you do not want to track. This way, they aren't added to GitHub, and when using shortcuts git will completely skip over it

Document the Code - Part 4

Example of best practice formatting - do this for each function in your .c or .cpp files:

```
/*
 * Function: approx_pi
 * -----
 * computes an approximation of pi using:
 *  $\pi/6 = 1/2 + (1/2 \times 3/4) 1/5 (1/2)^3 + (1/2 \times 3/4 \times 5/6) 1/7 (1/2)^5$ 
 *
 * n: number of terms in the series to sum
 *
 * returns: the approximate value of pi obtained by summing the first n terms
 *          in the above series
 *          returns zero on error (if n is non-positive)
 */

double approx_pi(int n) {
    ...
}
```

```
#include "utility.h"

ToDoList::ToDoList() {}

// this is where the comment should go
// what func does
// params of func
// what it returns
// example JAVA code provided if you are unsure what the code does!
void ToDoList::displayMenu() {
    std::cout << "\n===== To-Do List Menu =====> std::endl;
    std::cout << "1. Add Task" << std::endl;
    std::cout << "2. Mark Task as Completed" << std::endl;
    std::cout << "3. Remove Task" << std::endl;
    std::cout << "4. View To-Do List" << std::endl;
    std::cout << "5. Exit" << std::endl;
}
```

Using the code from Part 2, go over all of the files and make a comment block on each function including what the function does, parameters, and returns.

Makey Makey - Part 5

Tired of using Part 2's code? Too bad. Let's create a `Makefile` to build, run, and clean up our TaskToDo program.

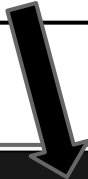
```
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/Lab10$ touch Makefile
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/Lab10$ ls
main.cpp  Makefile  README.md  utility.cpp  utility.h
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/Lab10$
```

A `Makefile` is just a text file that describes how to build your system. In this case, how to build our program so we don't have to type out the long command each time

Makey Makey - Part 5

Tired of using Part 2's code?
Too bad. For this part, create
a `Makefile` to build, run, and
clean up our TaskToDo
program.

Demo directory with example
cpp files!



```
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/demoMakeFile$ ls  
file2.cpp  header.h  main.cpp  Makefile  
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/demoMakeFile$
```

Makey Makey - Part 5

Example Makefile:

These are targets
(what you put after
`make` is called)

```
aiden: main.cpp file2.cpp header.h
      g++ -Wall main.cpp file2.cpp -o aiden

execute: aiden
        ./aiden

remove:
        rm -f aiden
```

Dependencies (what
is needed to do the
things

Commands. The “do
things part”

Makey Makey - Part 5

Running the targets:

```
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/demoMakeFile$ ls
file2.cpp header.h main.cpp Makefile
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/demoMakeFile$ make aiden
g++ -Wall main.cpp file2.cpp -o aiden
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/demoMakeFile$ ls
aiden file2.cpp header.h main.cpp Makefile
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/demoMakeFile$ make execute
./aiden

===== To-Do List Menu =====
1. Add Task
2. Mark Task as Completed
3. Remove Task
4. View To-Do List
5. Exit
Enter your choice (1-5): 5
Exiting program. Goodbye!
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/demoMakeFile$ ls
aiden file2.cpp header.h main.cpp Makefile
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/demoMakeFile$ make remove
rm -f aiden
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/demoMakeFile$ ls
file2.cpp header.h main.cpp Makefile
ubuntu@ceg2350-sandbox:~/ceg2350f25-aidenc17/demoMakeFile$
```

Have a Good Weekend!



Don't hesitate to reach out and ask questions!

Quote of the week:

“Everyday might not be a good day but there's something good in everyday”