CS61B Lab 1

Aniruth Narayanan

Agenda

1. Introduction
2. Announcements
3. Lab Format & Discord Demo
4. Lab 1 Overview
5. Crash Course "Repos in CS61B"
6. Useful Terminal Commands
7. Work Time in Discord

Introduction

Hey, I'm Aniruth (he/him/his)! I'm a second year EECS + Business major and this is my second time teaching 61B(L), though my first as a TA.

I'm from Tampa, Florida and am a (fairly) huge Bucs fan!

Discussion: Wednesday 9 AM

Lab: Thursday 9 AM

Office Hour: Monday 10 AM

Announcements

1. Lab 1 is due next Friday 1/28
   a. Note: Other labs are usually due every Friday at 11:59 PM PT
2. Fill out the pre-semester survey due 1/24

Lab Format

1. Online for first two weeks of school
   a. Mode of instruction TBD & will be announced later
2. Labs starts on Zoom
   a. Lab Assignment Overview
   b. Lab Presentation (Important Topics Covered in the Lab)
3. Work Time in Discord (CS61B Lab Discord Server)
   a. TAs will process questions in #lab-queue
   b. Feel free (and encouraged) to work with other students!
      i. Make friends!

Breakout Rooms

What's your favorite thing to do?

Rooms will start by being randomly assigned; I'll allow you to then switch breakout rooms if you'd like after about 30 seconds.

Introduce yourself and feel free to share something!

Discord Demo

Lab 1 Overview

1. Set up
    a. Terminal
    b. Instructional account
    c. Personal laptop
    d. CS61B Tools
2. Git
3. Java + First Java Assignment
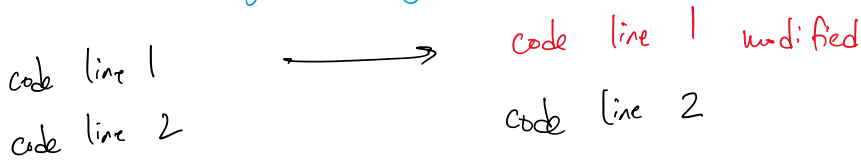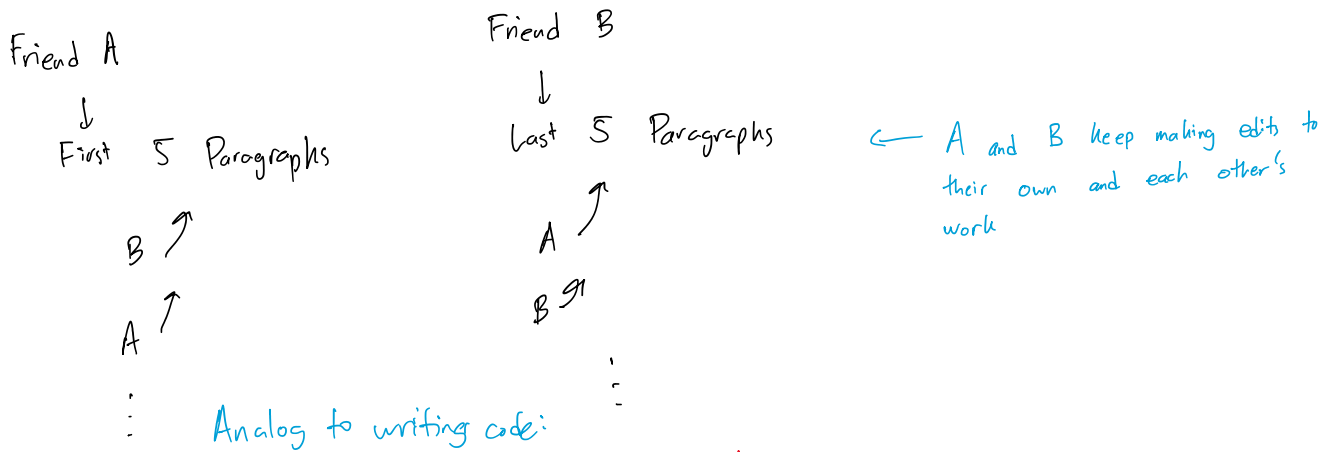
Crash Course: Repos in CS61B

Lab 1: Setup

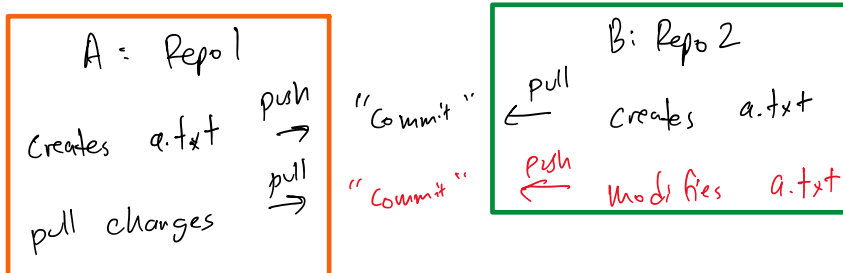Intuition for a Version Control System:

Consider two friends writing an essay together.

They work independently and may edit the same things. They also want to be accounting for each other's changes as well. And, they might need to revert to older versions.

Every time A edits something, they have to send a copy of the whole document to B, who updates their document, and vice versa. They keep old copies and different versio

Friend A
↓
First 5 Paragraphs

B ↗
A ↗
⋮

Friend B
↓
Last 5 Paragraphs

A ↗
B ↗
⋮

← A and B keep making edits to their own and each other's work

Analog to writing code:

code line 1   →   code line 1 modified
code line 2   code line 2

Git = Version Control System ← Solves issue of manually modifying / editing

A = Repo 1

Creates a.txt   push →

pull changes   pull →

"Commit"

"Commit"

B: Repo 2

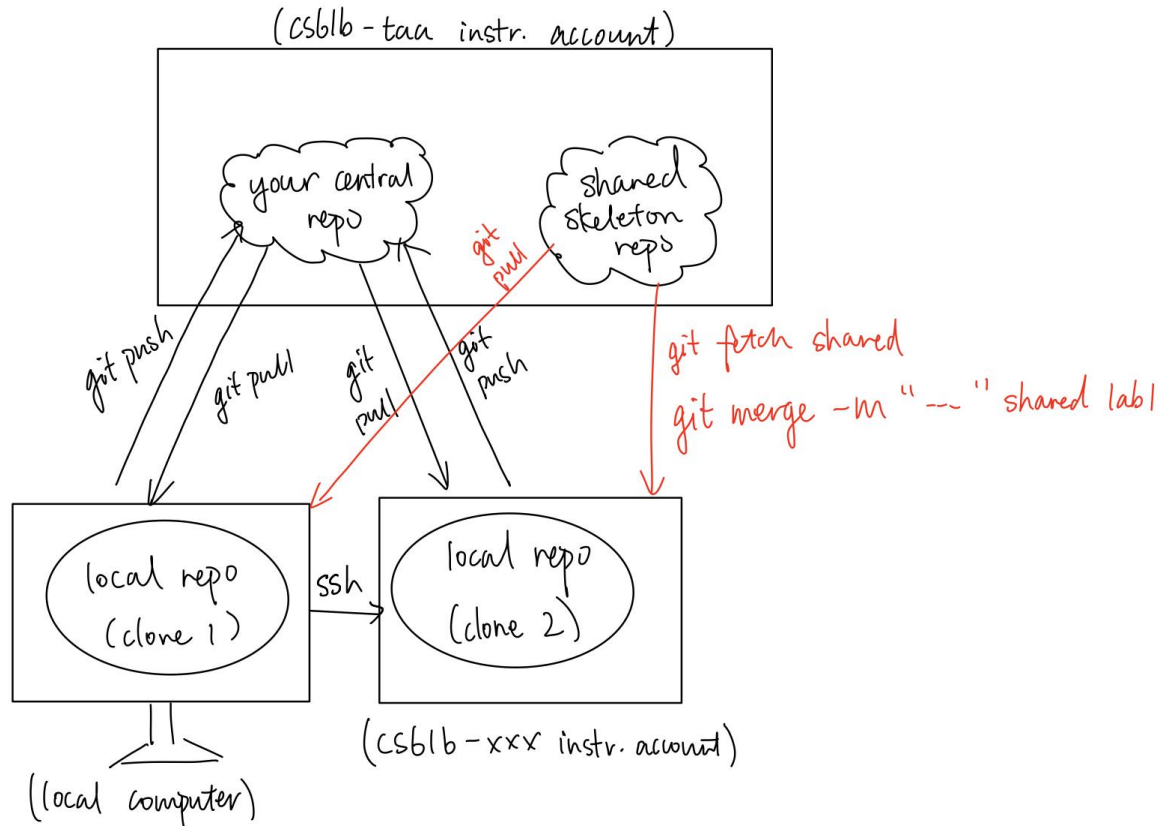← pull   Creates a.txt

← push   modifies a.txt

diffs are "tracked" by the commit object and then applied

If A edits line 50 of a.txt and B deletes line 24, the new line 50 and metadata is all that is stored!

Branches are like working on multiple versions of an essay; merging is combining them together.

Major git commands:

git add   adds a file for staging

git commit   snapshots added files into a commit

git push   pushes commit (to other clone)

git pull   fetches then merges commits (from other clone)

git fetch   gets pushed commits

git merge   combines branches

git status   shows unadded, modified files (if there are any)

Useful Terminal Commands Demo

Work Time

Discord: https://discord.gg/N2wVnmzMfr