



# BYU ACME Labs Initial Setup

Welcome to the BYU ACME program! This document describes protocols for using ACME lab computers, how to access lab materials, and the procedure for submitting your lab assignments.

## ACME Lab Computers

Room 150 TMCB (on the side of the hall opposite the math lab) is reserved at all times for ACME students and classes. The computers have Linux installed and are loaded with the libraries and modules needed to do the labs. After the add/drop deadline, you can unlock the room at all times by swiping your student ID card in the south door. Please **do not** prop the door open.<sup>1</sup>

Room 149 TMCB is shared between ACME and a few other classes. The other classes use Windows,<sup>2</sup> but you can switch to Linux by restarting the computer and selecting Linux. After you're done working, please restart the computer again and put it back on Windows. ACME students only have access to 149 TMCB during class; the TAs will hold office hours in 150 TMCB.

To log in to an ACME computer, enter the username `netid@byu.local` where `netid` is your BYU Net ID. The password is your MyBYU password. If you cannot log in, try a different computer, and if that doesn't work, try restarting the machine. A TA will assist you if you still can't log in.

You may use a personal laptop for the labs, but be warned that you are responsible for installing and maintaining all necessary software.<sup>3</sup> See the Installation Guide for some instructions.

## Lab Assignments and Materials

Each lab assignment is a PDF document, usually about 5-10 pages long, that includes code examples (yellow boxes), important notes (green boxes), warnings about common errors (red boxes), and about 3-7 exercises (blue boxes). Every lab has a corresponding specifications file with some code to get you started and to make your submission compatible with the test drivers. Start by going to <http://www.acme.byu.edu/2017-2018-materials/> and downloading the zip files for your classes: `Volume1.zip` and `Volume2.zip` for juniors, `Volume3.zip` and `Volume4.zip` for seniors. The folder inside of each zip file has some setup scripts and a collection of folders, one per lab. Each lab folder contains the specifications file and any auxiliary files for that lab.

---

<sup>1</sup>Please keep the math lab students out!

<sup>2</sup>If you're planning on using Windows, don't. Install Linux (virtual box, dual boot, etc.) and make your life easier.

<sup>3</sup>Seriously though, if you have a Windows laptop, consider just using the lab machines or installing Linux.

**ACHTUNG!**

Do not move or rename any of the provided specifications files or lab folders: the test driver won't be able to find your files and you will fail the assignment.

The assignment PDFs are also located at <http://www.acme.byu.edu/2017-2018-materials/>.

## Assignment Submissions via Git

*Git* is a tool for collaborative software development, and it is what we use to manage lab submissions. The basic idea behind git is that a group of files is stored in a *repository* online, and copies of the repository called *clones* are stored locally on computers. When a change is made locally to a clone, a series of git commands sends the updates to the repository. You and the TA will both have clones of your repository so that the TA can access your files, grade them, and provide feedback.

If you are using a laptop and git is not already installed on your computer, download it at <http://git-scm.com/downloads>. If you have never used git, you might want to read a few of the following resources at some point.

- <https://git-scm.com/docs/gittutorial>
- <https://www.atlassian.com/git/tutorials>

**NOTE**

You need one repository per class, meaning you have to complete the following instructions twice (once with **Volume1.zip** or **Volume3.zip**, and once with **Volume2.zip** or **Volume4.zip**). Make sure that you give permissions to the right TAs for each repository.

There are many websites for hosting online git repositories. For these classes, we use Bitbucket. Set up your Bitbucket repository with the following steps.

1. *Sign up.* Create a Bitbucket account at <https://bitbucket.org>. If you use an academic email address (ending in .edu, ac.il, edu.sg, etc.), you will get free unlimited public and private repositories.
2. *Make a new repository.* On the Bitbucket page, click the **Repositories** button from the menu at the top and select **Create repository**. Provide a name for the repository, mark the repository as **private**, and make sure the repository type is **Git**. Under **Advanced settings**, enter a short description for your repository, select **No forks** under forking, and select **Python** under language. Finally, click the blue **Create repository** button. Take note of the URL of the webpage that is created; it should be something like <https://bitbucket.org/username/repo>.
3. *Give the instructor access to your repository.* On your Bitbucket page, click the blue **Send invitation** button at top right part of the page. Enter your instructor's Bitbucket username and click **Add**. Select the blue **Write** button (so your instructor can write feedback to your repository) and click **Share**.

On Learning Suite, take the quiz to verify that you have completed this step and to give your TA the URL to your repository page.

4. *Connect your folder to the new repository.* Extract the folder from the lab materials zip file (Volume1.zip, etc.). In a shell application (Terminal on Linux or Mac, or Git Bash on Windows<sup>4</sup>), enter the following commands (here `username` is your Bitbucket username and `repo` is the name of your new repository).

```
# Navigate to your folder, extracted from one of the zip files.
$ cd ~/Desktop/foldername

# Connect this folder with the online repository.
$ git init
$ git remote add origin https://username@bitbucket.org/username/repo.git

# Add your credentials to the folder.
$ git config --local user.name "your name"
$ git config --local user.email "your email"

# Add the contents of this folder to git and update the repository.
$ git add --all
$ git commit -m "initial commit"
$ git push origin master
```

If you enter the repository URL incorrectly, you can reset it with the following line.

```
$ git remote set-url origin <repository url>
```

At this point you can go to your Bitbucket page and see the repository. The home page should have these same setup instructions and a list of git commands with examples.

5. *Download data files.* Many labs have accompanying data files. To download these files, navigate to your clone and run the `download_data.sh` bash script, which downloads the files and places them in the correct lab folder for you.

```
$ cd ~/Desktop/foldername      # Navigate to your folder.
$ bash download_data.sh        # Run the script.
```

6. (Optional) *Clone your repository.* If you want a clone of your repository on another computer after completing steps 1–4, use the following commands. Repeat step 5 in this new clone as well.

```
# Clone the folder from the online repository.
$ cd ~/Desktop
$ git clone https://username@bitbucket.org/username/repo.git foldername

# Add your credentials to the new folder.
$ cd foldername
$ git config --local user.name "your name"
$ git config --local user.email "your email"
```

---

<sup>4</sup>Git Bash comes with the git download on Windows.