

# Homework Submission Guide

- [Comp 311 Homework Submission Guide](#)
- Option A: Using SVN commands to access Rice SVN from personal machines
  - [Submitting your Homework from your Linux or Mac machine \(Using SVN commands\)](#)
  - [Submitting your homework from your Windows machine](#)
- Option B: Using the Rice SVN Repository from machines in the CLEAR cluster
  - [One time Setup in Clear](#)
  - [Submitting your Homework from a machine in the Clear Network](#)

## Comp 311 Homework Submission Guide

We will use Rice's Subversion (SVN) repositories for all homework submissions in this course.

All students (regardless of enrollment in Comp311 vs Comp544) must submit their homework to the same repository <https://svn.rice.edu/r/comp311/turnin/F20/>. There is a folder in this repository for each student enrolled in Comp 311 or Comp 544.

In past years, Rice IT supported a *turnin* command on its Clear cluster for uploading a program submission to the repository. The command was supported by a web interface so that you could perform the same operation remotely. This command is now available only on machines in the CLEAR cluster. The web interface no longer exists. For this reason, we strongly recommend that you use regular SVN commands to submit assignments from your personal machines to the Rice SVN repository. Hence, you will need to learn how to use an SVN client to directly interact with the repository, which is not difficult. If you prefer using Git, you can also try using [the Subversion client built into Git \(git svn\)](#). The old *turnin* command simply committed the contents of a directory (corresponding to a particular assignment) specified by an argument to the command. When you develop a program using the same directory structure (which we will stipulate), *all* of your commits to that directory are effectively submissions of your program. Of course, they aren't bona fide solutions until you finish your assignment. We will not inspect them until after the assignment is due.

Note: if you need to use a slip day, please send an email message to [comp311@rice.edu](mailto:comp311@rice.edu) stating that fact before the assignment is due, so we know not to grade your program prematurely.

## Option A: Using SVN commands to access Rice SVN from personal machines

### Submitting your Homework from your Linux or Mac machine (Using SVN commands)

1. One time setup on your machine
  - a. Make a new empty directory comp311 in your home directory and switch to that directory:  

```
mkdir comp311  
cd comp311
```
  - b. Use Subversion to check out your submission directory:  
**svn checkout <https://svn.rice.edu/r/comp311/turnin/F20/> «your-net-id»**  
E.g., for Fall 2020 and the NetID "xyz99":  
**<https://svn.rice.edu/r/comp311/turnin/F20/xyz99>**
2. To check the status of your current directory, run :  
**svn status**  
This should show all your files that are added(A) / not added(?) / deleted(D) / modified (M) to the Subversion repository.
3. To add folders/ files, use :  
**svn add «file\_name»** (without angle brackets)  
**svn add «folder\_name»** (without angle brackets) – recursively adds all files in it  
This should show all files/ folders that have been added to the Repository (this is still a local copy)
  - a. To delete folders/ files, use :  
**svn del «file\_name»** (without angle brackets)  
**svn del «folder\_name»** (without angle brackets) – recursively adds all files in it  
This should show all files/ folders that have been deleted to the Repository (this is still a local copy)
4. To commit your files, use:  
**svn ci -m "Some relevant message here"**  
This should show all the files that have been added/ modified/ deleted ... followed by  
-- Transmitting file data.  
-- Committed revision «rev. no»
5. And you have successfully added your folder/ files to the SVN repository. Visit [svn.rice.edu/r/comp311/turnin/<netID>](https://svn.rice.edu/r/comp311/turnin/<netID>) in your browser to confirm that all of your files (typically only one per HW assignment) are intact.

## Submitting your homework from your Windows machine

If you are using Windows 10, the course staff recommends that you either:

1. Set up the Windows Subsystem for Linux (WSL) and then follow the steps above for using the SVN command on Linux, or
2. Use a standalone visual SVN client like [TortoiseSVN](#), which augments the menus embedded Windows File Explorer with SVN commands akin to those supported in Linux using the command line.

I am following the second approach which is working well. I highly recommend it. I also have WSL installed on my Windows 10 laptop but I strongly prefer running the native Windows 10 edition of Racket on my machine than the Linux edition of Racket inside WSL. If you install WSL, there are two choices:

- WSL 1.0, the original version which embeds the Linux file system inside the Windows 10 file system and implements the Linux kernel as a Windows 10 application built on top of Windows services.
- WSL 2.0, a new implementation based on OS virtualization (as done by VMWare and VirtualBox).

In my experience, WSL 2.0 works better than WSL 1.0. Both are available for download from Microsoft (who developed them) and there are detailed instructions on the web on how to install either system. Regardless of whether you choose WSL 1.0 or WSL 2.0, you can select the particular variety of Linux that you want to install. I strongly recommend installing the latest version of the long-term release of Ubuntu (20.04). I have WSL 2.0 Ubuntu 20.04 installed on my laptop. The principal problem with WSL 1.0 is that the final name space in Windows is much more restrictive than it is in Linux and many apps that you try to install on WSL 1.0 blow up because they use illegal file names. The principal hassle with both WSL 1.0 and 2.0 is that GUI applications are painful to set up because they require some control over the windowing system on Windows 10. The hack that I use (again consult the web for various ways to solve the GUI issue) requires eliminating firewall protection for the base Windows 10 system, which I dislike. In contrast, the conventional command line interface simply relies on Windows 10 terminal applications for program interaction, so security on your laptop is not compromised. Microsoft is feverishly working on providing better support for GUI applications in WSL 2.0 so I expect a much better solution to this problem within a year. For now, WSL 2.0 is an interesting curiosity and works well for Linux apps without a GUI interface, but how many modern apps do not have a GUI?

Note: in the latter part of the course when programming in Java, you may want to use SVN integration supported by your IDE if it is available. DrJava does not support IDE integration, but IntelliJ IDEA and Eclipse do.

## Option B: Using the Rice SVN Repository from machines in the CLEAR cluster

### One time Setup in Clear

1. Login to CLEAR:  
**ssh <your netid>@ssh.clear.rice.edu**  
**<your password>**
2. Make a new empty directory **comp311** in your home directory and switch to that directory:  
**mkdir comp311**  
**cd comp311**
3. Use Subversion to check out your submission directory:  
**svn checkout https://svn.rice.edu/r/comp311/turnin/<current-semester>/<your-net-id>**  
E.g., for Fall 2017 and the NetID "xyz99":  
<https://svn.rice.edu/r/comp311/turnin/F17/xyz99>
4. You might be asked if you wish to store your password unencrypted. Always say **no** when asked this.
5. Now you should have a folder named **<your netid>** in the **comp311** directory you created.

## Submitting your Homework from a machine in the Clear Network

1. Make sure you are in your **turnin** directory.  
**cd ~/comp311/<your netid>**
2. To check the status of your current directory, run :  
**svn status**  
This should show all your files that are added(A) / not added(?) / deleted(D) / modified (M) to the Subversion repository.
3. To add folders/ files, use :  
**svn add <file\_name>** (without angle brackets)  
**svn add <folder\_name>** (without angle brackets) – recursively adds all files in it  
This should show all files/ folders that have been added to the Repository (this is still a local copy)
  - a. To delete folders/ files, use :  
**svn del <file\_name>** (without angle brackets)  
**svn del <folder\_name>** (without angle brackets) – recursively adds all files in it  
This should show all files/ folders that have been deleted to the Repository (this is still a local copy)
4. To commit your files, use:  
**svn ci -m "Some relevant message here"**  
This should show all the files that have been added/ modified/ deleted ... followed by

- Transmitting file data.
- Committed revision «rev. no»

5. And you have successfully added your folder/ files to the SVN repository. Inspect the Rice SVN repository <https://svn.rice.edu/r/comp311/turnin/«current-semester»/«your-net-id»> using your browser ( to confirm that all the file structure is intact