//Francsico Rodriguez (frodriguez127@toromail.csudh.edu)

These notes were written by two senior research software engineers at the University of Warwick for a half-day workshop on version control using Git. The authors first explain what version control is and why it is important for software development, providing benefits such as the ability to revert back to previous versions of code, work on different features independently, and store backups of code. They also mention that version control is becoming increasingly important. The authors then touch on why Git is the most likely version control system to be used and mention other online services such as GitHub that can be used to store and manage code.

In terms of using Git, the authors describe various commands such as git revert, git branch, git checkout, and git merge. They explain how to revert one or more commits using the command git revert {lower bound} {upper bound}, where the lower bound is the last commit that you want to leave unchanged and the upper bound is the last commit that you want to undo. They also describe how to create a new branch using git branch {name} and change to the new branch using git checkout {name}, as well as how to merge changes from another branch into the current one using git merge {other branch name}. Conflicts may occur during a merge, and the authors explain that these must be resolved before committing the changes. Git is a distributed version control system, allowing for linking between a local repository and a remote one, such as on Github.

The text also describes a process for version control of source code, including the importance of embedding version information into output files and using Git tags to connect a version number to a particular code state. The authors mention using Git commands such as git tag and git show to set and view tags, and describe the process of extracting version information from Git tags into code using a makefile.