== svn co ==
svn co svn+ssh://<ix username>@ix.cs.uoregon.edu/home/users/hank/SVN/mc410

This command does a "check out", i.e., creates a working copy of the repo in your file system.

410 students do mc410
510 students do mc510

Note that the svn+ssh protocol requires you to issue your password twice for check outs.

== svn ci ==
svn ci –m "update marching cube cases for YOUR NAME"

This command commits your changes ("svn checkin") to the repo. Until you invoke this command, all changes you make are local to your working copy, and are not accessible to others.

You should do an "svn up" before issuing this command, to prevent the case where your working copy is out of date, which prevents commits.

== svn add ==

Tells SVN that it should put a file in your working copy under version control. Without this step, SVN will just ignore unknown files.

You will not need this command for project 6B. Please do not put files like proj6B.cxx under revision control ... it will create big problems for others.

Repeat: do not issue an svn add

== svn up ==

"svn up" updates your working copy to have the current state of the repo. If you have local changes, those changes are preserved (i.e., not thrown out). That said, if your local changes conflict with recent changes in the repo, then your file will be marked in the conflict state and you'll need to either revert your own changes, or resolve the changes yourself (and tell SVN that you have resolved them via "svn resolved").

== svn blame ==
svn blame tricases.cxx

This will show, for each line of a file (tricases.cxx), the history of who was the last person to modify that line, and what revision they did so. Useful for tracking down who caused a problem, i.e., blaming.

== svn diff ==
svn diff tricases.cxx

This will show differences on your local copy. The differences are relative to the last time you updated your working copy ... it does not go to repo and check for differences against the latest version.