CSC207: Useful Unix and Git Commands

Unix commands:

- Editing and showing file content
  - `setenv SVN_EDITOR nedit` → make `nedit` your editor.
    - If you'd prefer to use a different editor, `nano`, `vim`, `emacs` and `ed` are all available on CDF.
    - For anyone using `bash`, the command to use is: `export SVN_EDITOR=nedit`
  - `nedit` *file* → edit file named *file* using editor `nedit`
  - `cat` *file* → show the content of the file *file*
- Showing current directory and directory content
  - `pwd` → display the path to the current directory ("print working directory")
  - `ls` → list files in the current directory
  - `ls` *dir* → list files in the directory named *dir*
    - The option `ls -l` *dir* produces more details ("l for long").
    - The option `ls -a` *dir* also displays hidden files and directories (the ones that begin with ".").
- Changing to a particular directory
  - `cd` *dir* → enter the directory named *dir*
  - `cd ..` → go to parent directory
  - `cd` → go to your home directory
  - `cd ~` → go to your home directory
  - `cd -` → go back to a directory we just came from
- Creating a directory
  - `mkdir` *dir* → create a new directory named *dir*
- Copying, renaming, deleting, and moving files and directories
  - `cp` *file1 file2* → create a copy of *file1* named *file2*
  - `cp` *file dir* → create a copy of *file* in the directory named *dir*
  - `rm` *file* → delete the file named *file* ("rm for remove")
  - `rmdir` *dir* → delete the directory named *dir* (*dir* must be empty)
  - `rm -r` *dir* → delete the directory named *dir* and all its content. **Use with caution! You can delete ALL your files with this!**
  - `mv` *file1 file2* → rename *file1* as *file2*, or move *file1* to new location ("mv for move")

Common Git definitions

- *Definition:* "repo" means repository; a compressed representation of a file system and all changes for one or more branches
- *Definition:* "local repo" means a local repository; a *repo* which exists in the current directory within the `.git` folder
- *Definition:* "remote repo" means a remote repository; a *repo* which exists on another computer.
- *Definition:* "branch" is a copy of files with a name, and a collection of revisions. We will usually be working with the branch called "master".

- *Definition:* "revision" and "commit" are sequential snapshots of a branch, where the changes of files are tracked
- *Definition:* "current revision" or "current commit" means the local files checked out, in the working directory, which correspond with a local repo branch

Git commands

- `git fetch REPO BRANCH` — fetch a branch named BRANCH, and save into the local repo. Copy from a repo, labelled REPO, with label BRANCH. For CSC207, usually `git fetch origin master` .
- `git checkout REPO` — check out a current revision of a local repo labeled BRANCH
- `git pull REPO BRANCH` — merge a BRANCH of a repo named REPO onto the current revision. Also, run fetch REPO BRANCH in the background. For CSC207, usually `git pull origin master` .
- `git add FILE` — mark FILE as being part of the next commit
- `git commit -m ‘‘DESCRIPTIVE MESSAGE’’` — save all of the changes made so far as a new revision (can be executed many times), into the local repo and current branch.
- `git push REPO BRANCH` — push all changes made to a remote; to the repo REPO and branch BRANCH. For CSC207, usually `git push origin master` .
- `git status` — print the status of working copy files