CS35L Software Construction Laboratory

Lab 5: Sneha Shankar

Week 10; Lecture 1

Important Note

- Submit assignment 10 in the link for Lab 5. If already submitted in the common link, please resubmit
- **No lateness allowed for Assignment 9 and 10 !!!**
- **Final Exam: Tuesday, March 20 - 3pm-6pm**
- **Location: BH 5422**
- Open book, open notes
- Written test. (No access to linux terminals)
- 50% weightage

Branching in git

- Create a new branch: **git checkout –b <branchname>**
- Create a file and commit changes to the branch
- Switch to master: **git checkout master**
- Merge the branch: **git merge <branchname>**

You should now see the new file (you created in your branch) in the master's working repo

- Add, commit and push to the master
- Delete the branch: **git branch –d <branchname>**

More Git Commands

- Reverting
  - $ git checkout HEAD main.cpp
    - Gets the HEAD revision for the working copy
  - $ git checkout -- main.cpp
    - Reverts changes in the working directory
  - $ git revert
    - Reverting commits (this creates new commits)
- Cleaning up untracked files
  - $ git clean
- Tagging
  - Human readable pointers to specific commits
  - $ git tag -a v1.0 –m 'Version 1.0'
    - This will name the HEAD commit as v1.0

Lab Assignment 9

- Installing Git
  - Ubuntu: $ sudo apt-get install git
  - SEASnet
    - Git is installed in /usr/local/cs/bin
    - Add it to PATH variable or use whole path
      - $ export PATH=/usr/local/cs/bin:$PATH
- Make a directory ‘gitroot’ and get a copy of the Diffutils Git repository
  - $ mkdir gitroot
  - $ cd gitroot
  - $ git clone git://git.savannah.gnu.org/diffutils.git
- Follow steps in lab and use man git to find commands

Lab Assignment 9

- Backporting
  - Apply a patch to a previous version
- Fix an issue with the diff diagnostic
- Commands to be used

  git clone

  git log

  git tag

  git show <hash_value>

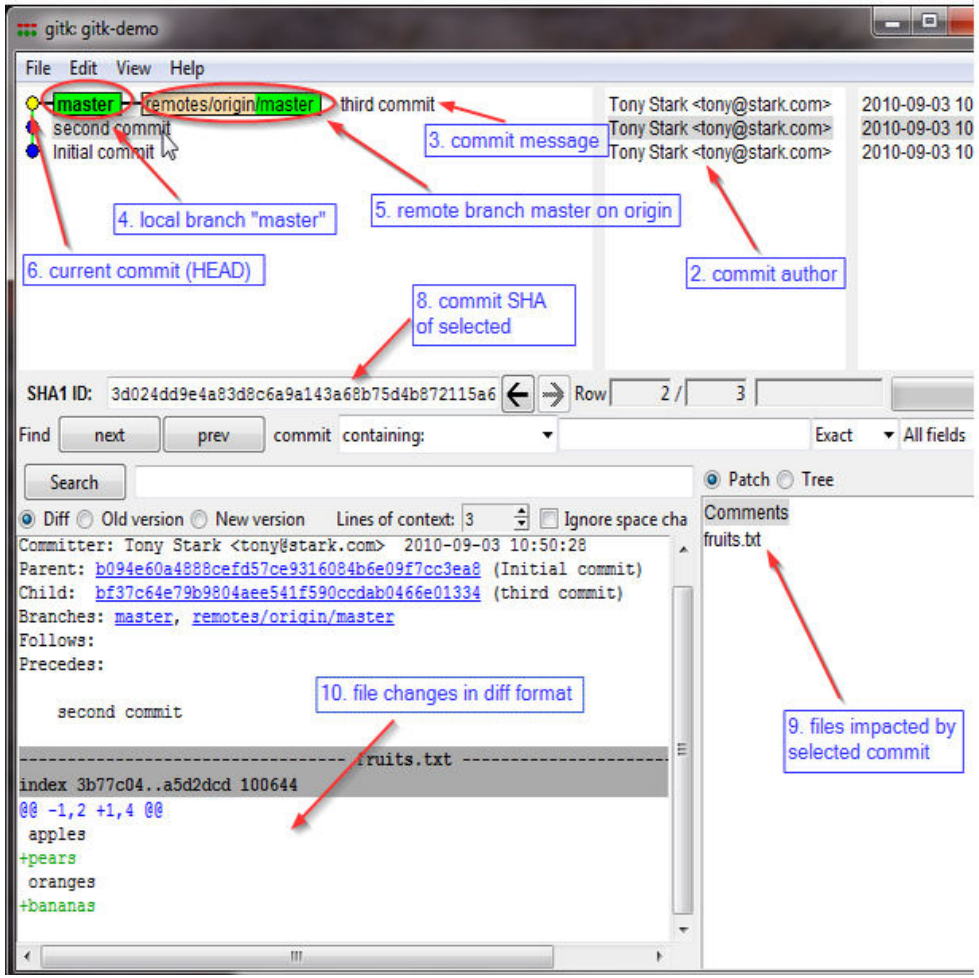
  git checkout v3.0 –b <branchname>

Homework 9

- Publish patch you made in lab 9
  - Create a new branch "quote" off of version 3.0
    - Branch command + checkout command **(git branch quote v3.0; git checkout quote)**
    - `$ git checkout v3.0 -b quote`
  - Use patch from lab 4 to modify this branch
    - Patch command
    - `$ patch –pnum <` quote-3.0-patch.txt
  - Modify ChangeLog file in diffutils directory
    - Add entry for your changes similar to entries in ChangeLog
  - Commit changes to the new branch
    - `$ git add .` `$ git commit –F <Changelog file>`
  - Generate a patch that other people can use to get your changes
    - `$ git format-patch -[num] --stdout > formatted-patch.txt`
  - Test your partner's patch
    - Check out version 3.0 into a tmp branch
    - Apply patch with `git am` command: $ `git am <` formatted-patch.txt
    - Build and test with $ `make check`
    - Make sure partner's name is in HW9.txt for #8

Gitk

- A repository browser
  - Visualizes commit graphs
  - Used to understand the structure of the repo
  - Tutorial: http://lostechies.com/joshuaflanagan/2010/09/03/use-gitk-to-understand-git/



Gitk

- SSH into the server with X11 enabled
  - ssh -X for OS with terminal (OS X, Linux)
  - Select “X11” option if using putty (Windows)
- Run gitk in the ~eggert/src/gnu/emacs directory
  - Need to first update your PATH
    - $ export PATH=/usr/local/cs/bin:$PATH