# Midterm – Git Rebase Example with Two Merge Conflicts

#### Assignment Goal:

This assignment will familiarize you with **git rebase**.

The major steps in this assignment are:

* creating a branch-feature branch with two commits
* adding two commits to master to simulate parallel development with the branch-feature branch
* rebasing the branch-feature branch on master and resolving the two merge conflicts (i.e., one merge conflict for each commit in the branch that is rebased)
* merging the branch-feature branch into master to maintain a linear history of the commits (but not preserving the temporal aspect of the commits)

**Resolution Screenshots:**

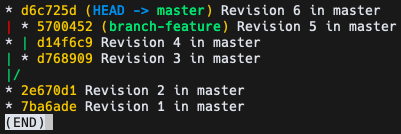
****

Figure 1: GIT\_VISUALIZATION\_ALL\_BRANCHES after completing Step 6

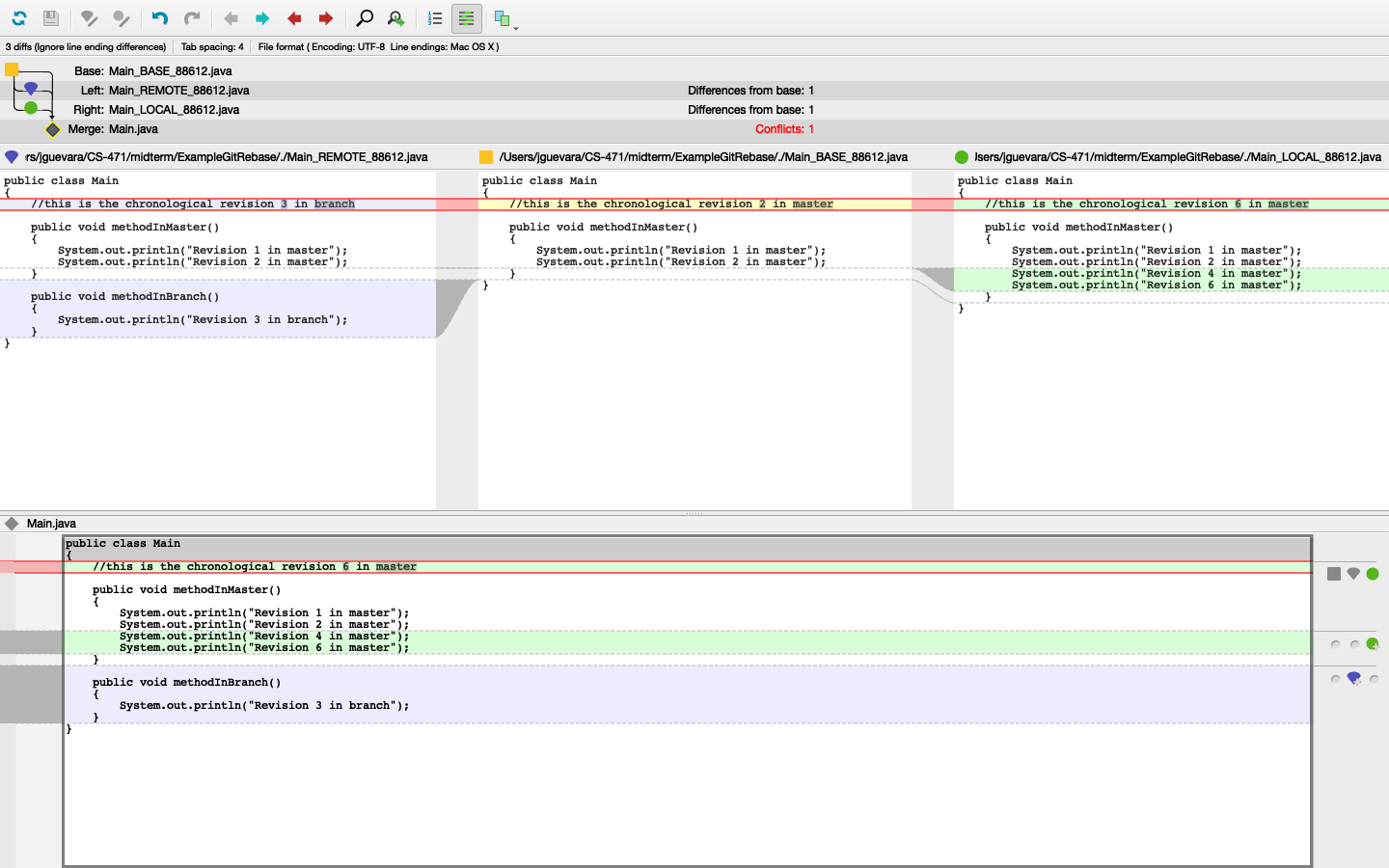


Figure 2: GIT\_MERGE\_CONFLICT\_VISUALIZATION after completing Step 9

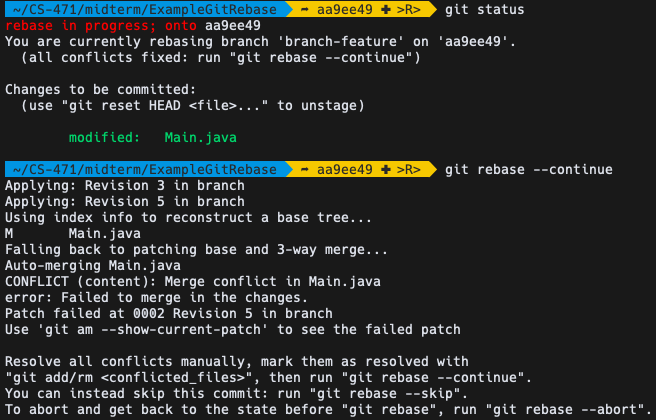
****

Figure 3: GIT\_COMMAND\_LINE\_OUTPUT after completing Step 11

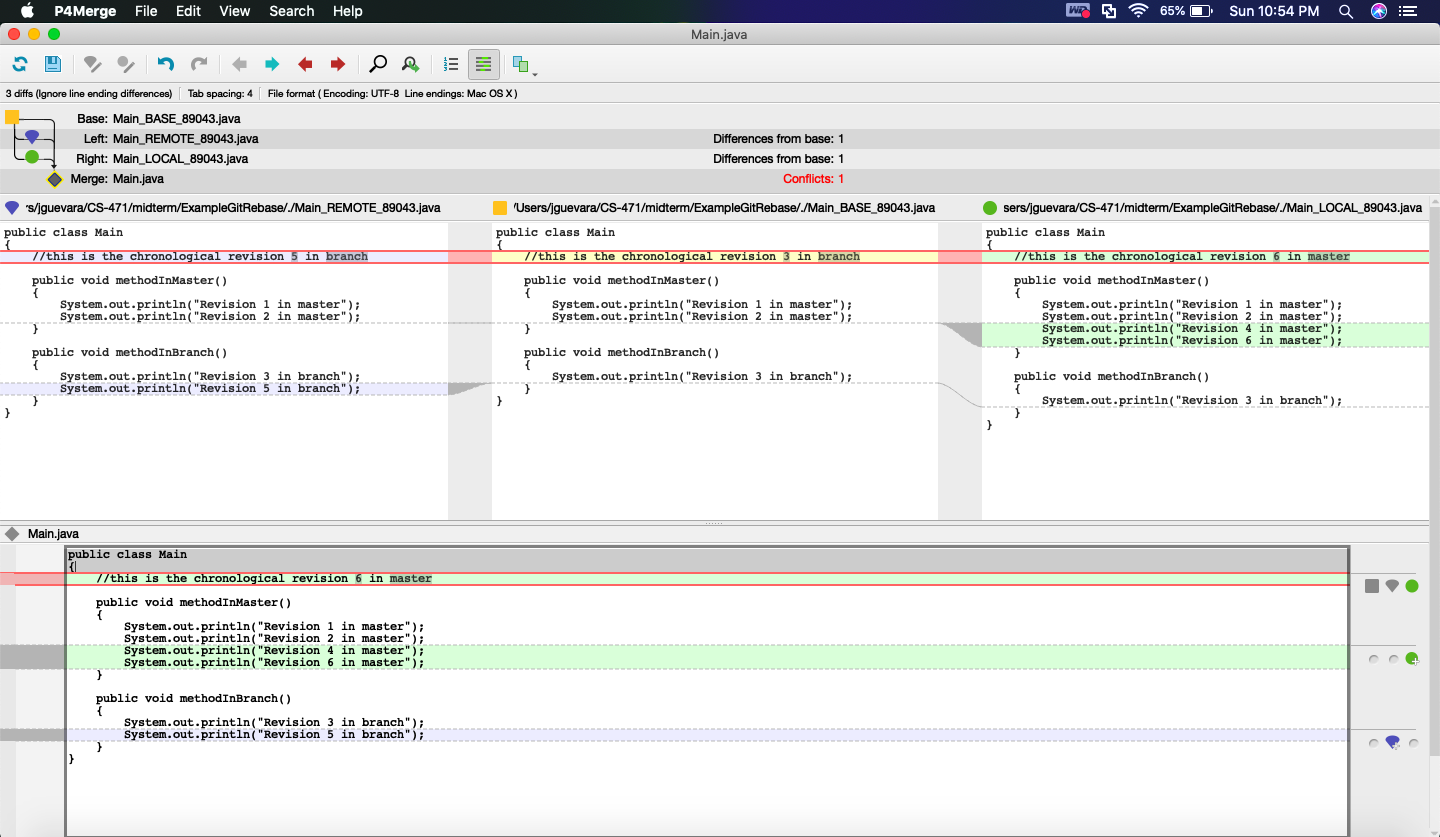


Figure 4: GIT\_MERGE\_CONFLICT\_VISUALIZATION after completing Step 12

****

Figure 5: GIT\_VISUALIZATION\_ALL\_BRANCHES after completing Step 14

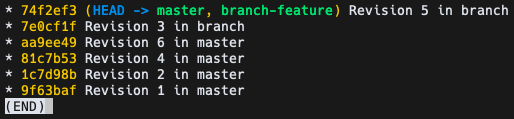
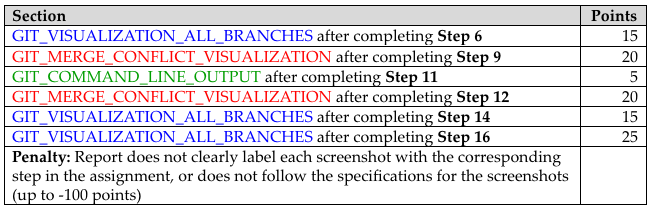


Figure 6: GIT\_VISUALIZATION\_ALL\_BRANCHES after completing Step 16

**Grading Rubric:**

****