Student Name:
Student ID:

1. Question Number 8

   (a) Git allows multiple working directories through the use of ”Git worktree” feature. This feature allows users to have multiple working trees, which are essentially copies of the repository’s working directory that can be used simultaneously.
   The worktree feature works by creating additional working directories, or ”worktrees,” for the same repository. These worktrees share the same repository database and history, but can have different working directories and branches checked out at the same time. Each worktree has its own HEAD file, which keeps track of the currently checked out branch or commit.

   (b) Four git commands that lead to a directed edge from the staging area (Index) to working directory are as follow :

      i. git checkout $< filename >$
      ii. git reset $< filename >$
      iii. git reset
      iv. git restore –staged .

   Example of git flow :

      i. git init
      ii. touch file
      iii. git add file
      iv. git checkout file
      v. git add file
      vi. git reset file
      vii. git add file
      viii. git reset
      ix. git add file
      x. git restore –staged .

   (c) Example of committing single file in parts:

      i. git init
      ii. echo”first part” > file
      iii. git add file
      iv. echo”second one” >>file
      v. git add –patch file
      vi. git commit

   (d) Example of converting changes committed in two different commits into a single commit is :

      i. git init
      ii. touch file1 file2 file3
      iii. git add file1

iv. git commit -m”first”

v. git add file2

vi. git commit -m”second”

vii. git add file3

viii. git commit -m”third”

ix. git-graph -n commits

x. git rebase -i HEAD 2

xi. pick 37c36 third
squash 928a32 second

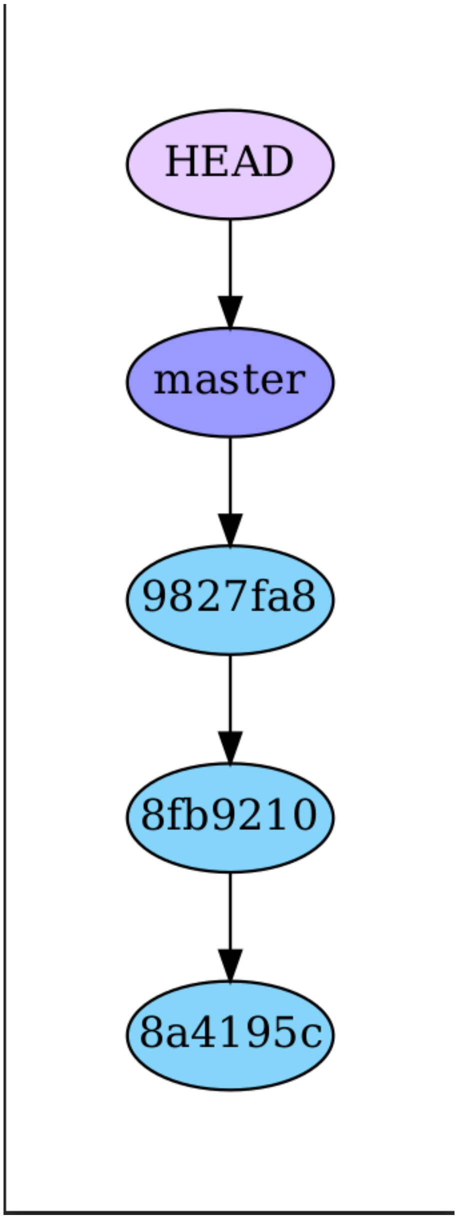

Figure 1: Before combining second and third commits

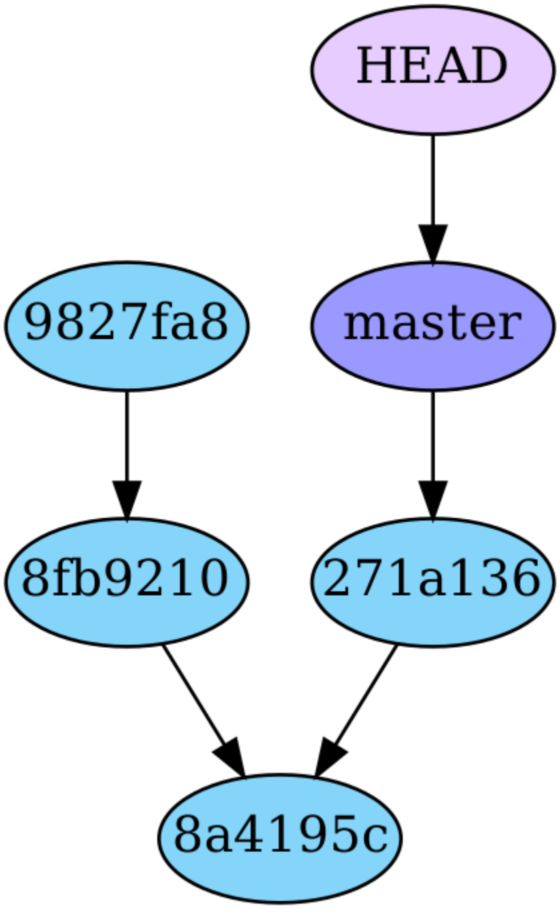

Figure 2: After combining