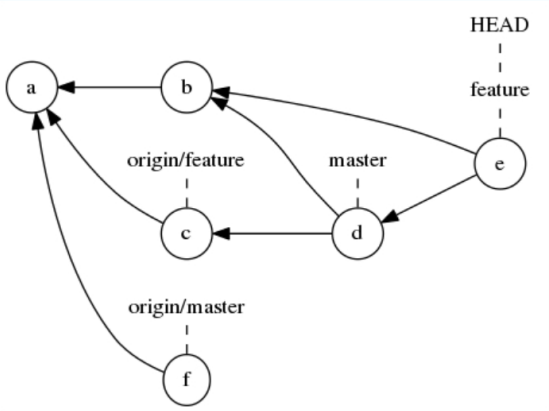
# TRIMESTER 2 2023 COSC220 Software Development Studio 2

Information
Flag question

Imagine you are working on a project using a git repository. Your local repository was created (via git clone) some days ago, and has the following revision graph:




For convenience, the commits have been labeled. HEAD represents the currently checked out branch. An arrow leading from a commit points to its parent (or parents).

Question 8
Not yet answered
Marked out of 2.00
Flag question

The *git log* command describes the commits on the current branch. If you ran *git log*, which commits in this graph would be listed?

Note: select *all* the commits that would be listed, and *none* of the commits that would not be listed.

- [ ] a
- [ ] b
- [ ] c
- [ ] d
- [ ] e
- [ ] f

Question 9
Not yet answered
Marked out of 2.00
Flag question

Which commits in the repository must be *merge commits*? Select which they are.

Note: Select *all* the commits that must be merge commits, and *none* of the commits that are not.

- [ ] a
- [ ] b
- [ ] c
- [ ] d
- [ ] e
- [ ] f

Question 10
Not yet answered
Marked out of 1.00
Flag question

A tag and a branch are both pointers to commits. Select which statement is true.

- a. Tag pointers are used only in merge requests, whereas branch pointers are used in commit operations
- b. Branch pointers cannot be pushed to a remote repository but tag pointers can
- c. Tag pointers cannot be pushed to a remote repository but branch pointers can
- d. Branch pointers and tag pointers are both static references to a single commit, that do not normally move when invoking with git commands.

Quiz navigation

i 1 2 3 4 5 6 7
i 8 9 10 11 12 13 14
15 16

Finish attempt ...

- e. A tag pointer moves as developers commit their work, but a branch is a static reference to a single commit
- f. A branch pointer moves as developers commit their work, but a tag is a static reference to a single commit

**Question 11**
Not yet answered
Marked out of 1.00
Flag question

If you ran the command *git fetch*, which of these options could happen?

Note: for this question assume the git graph is as it is shown in the diagram (i.e. none of the commands talked about in other questions have been run)

- a. The feature might move but the origin/feature pointer would not
- b. Neither the origin/feature nor the feature pointer could move
- c. The origin/feature might move but the feature pointer would not
- d. Both the origin/feature and the feature pointer could move

**Question 12**
Not yet answered
Marked out of 2.00
Flag question

A Continuous Integration server is set up, regularly building and deploying from the *master* branch on the *origin* server. Which commits would the currently deployed version of the software contain?

Note: select *all* the commits it would contain, and *none* of the commits it would not contain

- [ ] a
- [ ] b
- [ ] c
- [ ] d
- [ ] e
- [ ] f

**Question 13**
Not yet answered
Marked out of 1.00
Flag question

Suppose the continuous integration server is running the unit tests from the master branch (and only the master branch) on the origin server every 15 minutes. Suppose the tests succeeded on the most recent build on the continuous integration server, but the tests *fail* when you run them locally on your currently checked out commit, due to a bug introduced at commit *b*.

Note: for this question assume the git graph is as it is shown in the diagram (i.e. none of the commands talked about in other questions have been run).

Consider the git push commands you could run. Which of the following is true?

- a. Neither *git push origin feature* nor *git push origin master* would cause the continuous integration tests to fail
- b. *git push origin feature* would cause the continuous integration test to fail but *git push origin master* should not
- c. *git push origin master* would cause the continuous integration tests to fail but *git push origin feature* should not
- d. *git push origin feature* and *git push origin master* would both cause the continuous integration tests to fail

**Question 14**
Not yet answered
Marked out of 1.00
Flag question

If you ran the command *git merge master*, and the command completed successfully, what would happen?

Note: for this question assume the git graph is as it is shown in the diagram (i.e. none of the commands talked about in other questions have been run)

- a. It would create a merge commit
- b. It would be a fast-forward merge that would move the master pointer
- c. It would have no effect - the graph and all pointers would remain unchanged

**Question 15**
Not yet answered
Marked out of 1.00

If you ran the command *git merge origin/master*, and the command completed successfully, what would happen?

Note: for this question assume the git graph is as it is shown in the diagram (i.e. none of the

Flag question

commands talked about in other questions have been run)

- a. It would be a fast-forward merge
- b. It would have no effect - the graph and all pointers would remain unchanged
- c. It would create a merge commit

Question 16

Not yet answered

Marked out of 1.00

Flag question

If you ran the command *git cherry-pick f* (using the commit hash for "f") which of these options could happen?

Note: for this question assume the git graph is as it is shown in the diagram (i.e. none of the commands talked about in other questions have been run).

Also note: this is a more advanced command that we do not always describe in the course notes (though its use may come up in course announcements). You may wish to run "git help cherry-pick" to read the documentation on what it does before answering.

- a. The HEAD and feature pointers would move back to commit f
- b. The HEAD and feature pointers would move back to commit a
- c. A new commit would be created. The HEAD and feature pointers would move to this new commit

Previous page

Finish attempt ...

## Acknowledgement of Country

*The University of New England acknowledges that we are on Country of the Anaiwan people. UNE respects and acknowledges that its people, programs and facilities are built on land, and surrounded by a sense of belonging, both ancient and contemporary, of the world's oldest living culture. UNE also acknowledges the Gumbaynggirr, Kamilaroi, and Dhunghutti nations and pays its respect to the Elders, past and present, of these nations.*

Oorala Aboriginal Centre