

Knowledge Test K2

COMP 4081 • Software Engineering • Fall 2019

Solutions

Name: _____, _____
Last name First name

Rules:

- No potty breaks.
- Turn off cell phones/devices.
- Closed book, closed note, closed neighbor.
- WEIRD! Do not write on the backs of pages. If you need more pages, ask me for some.

Reminders:

- Verify that you have all pages.
- Don't forget to write your name.
- Read each question carefully.
- Don't forget to answer every question.

Each of the following problems presents a Git log graph (log messages omitted) of a local repo and a scenario. Update the graph by crossing out and/or adding appropriate text.

- You may or may not need to use the two blank lines above the graph.
- If you need to add a commit, use the hash `c1c1c`.
- If a command would be rejected by GitHub (e.g., because the remote contains work that you do not have locally), write “REJECTED” on the top line.
- Assume that all remote bookmarks depicted are up to date.

I have included an example problem and solution below to help clarify what’s expected.

Example Problem

Scenario: Developer makes changes to the code, stages the changes, and commits.

```
* 86b81 (HEAD -> master, iss1)
* dc003
* 026c6
```

Example Solution

```
* c1c1c (HEAD -> master)


---


* 86b81 (iss1)


---


* dc003


---


* 026c6


---


```

1. [5] Scenario: Developer makes changes to the code, stages the changes, and commits.

```
* 26c94 (iss1)
* 98d8b
| * 2e1ec (HEAD -> master)
|/
* cb157
```

*** c1c1c (HEAD -> master)**

*** | 26c94 (iss1)**

*** | 98d8b**

| * 2e1ec

|/

*** cb157**

2. [5] Scenario: Developer runs `git checkout iss1`.

```
* 26c94 (iss1)
* 98d8b
| * 2e1ec (HEAD -> master)
|/
* cb157
```

*** 26c94 (HEAD -> iss1)**

*** 98d8b**

| * 2e1ec (master)

|/

*** cb157**

3. [5] Scenario: Developer runs `git checkout -b iss2`.

```
* 26c94 (iss1)
* 98d8b
| * 2e1ec (HEAD -> master)
|/
* cb157
```

*** 26c94 (iss1)**

*** 98d8b**

| * 2e1ec (HEAD -> iss2, master)

|/

*** cb157**

4. [5] Scenario: Developer runs `git merge iss1`. Assume that auto-merge, if used, would complete successfully with no merge conflicts.

```
* 26c94 (iss1)
* 98d8b
| * 2elec (HEAD -> master)
|/
* cb157
```

*** c1c1c (HEAD -> master)**

/ |

*** | 26c94 (iss1)**

*** | 98d8b**

| * 2elec

|/

*** cb157**

5. [2] Would this be a fast-forward merge?

a) Yes

b) No

6. [5] Scenario: Developer runs `git merge iss1`. Assume that auto-merge, if used, would complete successfully with no merge conflicts.

```
* 26c94 (iss1)
* 98d8b
| * 2elec (master)
|/
* cb157 (HEAD -> iss3)
```

*** 26c94 (iss1, HEAD -> iss3)**

*** 98d8b**

| * 2e1ec (master)

|/

*** cb157**

7. [2] Would this be a fast-forward merge?

a) Yes

b) No

8. [1] Which of the following commands would show you which files in your working directory are untracked, which changes are staged, which branch you're on, and whether your working directory is clean?

a) `git init`

b) `git log`

c) `git status`