

University of Washington
CSE 403 Software Engineering
Winter 2024

Exam

February 9, 2024

Name: _____

CSE Net ID (username): _____

This exam is closed book, closed notes. You have **50 minutes** to complete it. It contains 33 questions and 9 pages (including this one), totaling 100 points.

Before you start, please check your copy to make sure it is complete. When you are finished, turn in all pages, together. **Write your initials on the top of ALL pages you turn in** (in case a page gets separated during test-taking or grading).

When you are asked for multiple answers, give answers that are as different as possible. Always give the most important answers.

Unless otherwise directed, answer in **one phrase** (when one answer line is given) or **one sentence** (when two or three answer lines are given).

If your answer to a T/F or multiple-choice answer is contingent on missing information, *briefly* note that. This should be very rare.

Please write neatly; we cannot give credit for what we cannot read.

Good luck!

1 True/False

(1.5 points each, 12 points total) Circle the correct answer. T is true, F is false.

1. **T / F** The Joel Test is a checklist for the quality of software product.
2. **T / F** If code merges without a merge conflict, the merge does not introduce any new defects (bugs). (We define a “new defect” as one that was not in either of the parents of the merge.)
3. **T / F** Each commit to your version control system should pass tests.
4. **T / F** In requirements, an actor is a person who interacts with the system under development.
5. **T / F** In requirements, every stakeholder is also an actor.
6. **T / F** In requirements, every actor is also a stakeholder.
7. **T / F** A mutation is a small modification to program source that introduces a defect (a bug).
8. **T / F** Given test suite T1 with 50% code coverage and test suite T2 with 60% code coverage, T2 detects at least as many failures (bugs) as T1 does.

2 Multiple choice

Choose the best answer.

9. (2 points) A team of 5 programmers is working on a project using git and GitHub. How many repositories exist (each of which stores its own version of the history of your project)?
 - (a) 0 or more
 - (b) 1
 - (c) 1 or more
 - (d) 5
 - (e) 5 or more
 - (f) 6
 - (g) 6 or more
 - (h) 7
 - (i) 7 or more

3 Choose all that apply

(4 points each, 20 points total) Mark all the correct answers, by circling the appropriate letters.

10. Which git commands can be run without network access, assuming typical usage of Git and GitHub?
 - (a) bisect
 - (b) clone
 - (c) commit
 - (d) fetch
 - (e) fork
 - (f) merge
 - (g) pull
 - (h) push

11. Which of the following are true of a merge from the main branch into a feature branch?
 - (a) It should be done prior to a merge of the feature branch into main
 - (b) It should be accomplished via approving a pull request on GitHub.com
 - (c) It cannot result in a merge conflict

12. Which of the following are best practices for avoiding merge conflicts?
 - (a) Create a branch for each distinct task
 - (b) Pull often
 - (c) Design your software modularly

13. Which of the following might appear in an architectural diagram?

- (a) a web client
- (b) CSS
- (c) a database
- (d) design patterns
- (e) Java
- (f) React
- (g) a server
- (h) the user

14. Which of the following events might trigger (start) a continuous integration workflow?

- (a) save code edits
- (b) compile
- (c) run static analysis
- (d) run tests locally
- (e) git commit
- (f) git push
- (g) git pull
- (h) create a pull request
- (i) open an issue or bug report

4 Very short answer

Answer in one word or phrase.

15. (3 points) What architectural style is most appropriate for implementing a compiler?

16. (4 points) Name the 7 main distinct steps in the software development lifecycle, in order.

(a) _____

(b) _____

(c) _____

(d) _____

(e) _____

(f) _____

(g) _____

17. (3 points) What is the purpose of mutation testing?

18. (4 points) State three ways that a build system can speed up a build, in one brief phrase (1–2 words each is adequate).

(a) _____

(b) _____

(c) _____

19. (6 points) Requirements

(a) Give an example of a functional requirement.

(b) Give an example of a non-functional requirement.

(c) Give an example of a requirement that is neither functional nor non-functional.

20. (3 points) What are the two major concepts that an architecture defines?

(a) _____

(b) _____

21. (3 points) Give an example of a file that should *not* be committed to version control, or say “none” if all files should be committed to version control.

22. (5 points) Alice and Bob are developing using GitHub, each one with a clone on their own laptop. Alice’s repository is up to date. Alice makes an edit and saves a file. What is the smallest number of git commands that enable Bob to see Alice’s changes in his own repository? State the **actor** (Alice or Bob) and the **git command** (without any command-line arguments). Leave blank any of the answer lines that you do not need.

(a) _____ runs **git** _____ .

(b) _____ runs **git** _____ .

(c) _____ runs **git** _____ .

(d) _____ runs **git** _____ .

(e) _____ runs **git** _____ .

(f) _____ runs **git** _____ .

5 Short answer

23. (4 points) What is the difference between verification and validation?

24. (4 points) What is the difference between git and GitHub?

25. (4 points) What is a software design pattern?

26. (6 points) Give 3 general approaches, that are as different as possible, to obtain information from customers when gathering requirements.

(a) _____

(b) _____

(c) _____

27. (4 points) Define “technical debt”.

6 Git bisect

For all of this section, assume that the version control history is linear, containing n commits.

28. (2 points) What is the best-case complexity of `git bisect`, in big- O notation?

29. (5 points) Give a modification to the `git bisect` algorithm that improves the best-case complexity without degrading the average-case or worst-case complexity.

No credit is given below unless credit was given for this question. (We don't want to reward guessing below.)

30. (1 point) What is the new best-case complexity? _____

31. (1 point) What is the new worst-case complexity? _____

32. (1 point) What is the new average-case complexity? _____

33. (3 points) Why haven't the git maintainers implemented this improvement?

