

Name:			
Enrolment No:			
UPES End Semester Examination, December 2025 Course: Software Version Control Program: BTech CSE-All Programs Course Code: CSEG4013P_3 Semester: VII Time: 03 hrs Max. Marks: 100			
Instructions: Attempt all questions. Internal choices have been provided in some questions.			
SECTION A (5Qx4M=20Marks)			
S. No.		Marks	CO
Q 1	Differentiate between a directory and a repository. Write the git command to initialize a repository.	4	CO2
Q 2	A fast-paced, highly trusted development team releases updates multiple times a day. Developers need to integrate their changes frequently and safely, even when some features are incomplete. - Which Git workflow is most suitable for this project? - Give two reasons why this workflow is appropriate.	4	CO2
Q 3	Differentiate between SVN and Git.	4	CO4
Q 4	What does it mean to completely reset a local Git environment and why it's done? Write the steps or commands used to perform a complete reset.	4	CO1
Q 5	Define a changeset in Mercurial and write its role in version control. Write the names of three phases of changesets in Mercurial.	4	CO3
SECTION B (4Qx10M= 40 Marks)			
Q 6	Explain the history of Git in short. Include: - The first release of Git. - Why Linus Torvalds created it. Git uses cryptographic hashes to ensure the integrity of commits. Answer the following with respect to this: - Differentiate between SHA-1 and SHA-256, and explain why SHA-256 may be preferred. - When can a developer choose to use SHA-256 while creating a project, and write the Git command used to enable SHA-256.	10	CO3
Q 7	Explain the multiple-repository (multi-repo) model used in software development. Discuss its advantages and challenges compared to a monorepo approach. In your answer, also describe the purpose of the .gitignore file within a multi-repo setup, and provide two examples of files or directories that are typically added to .gitignore.	10	CO4

Q 8	Define Continuous Integration (CI) and explain it using a neat diagram. Also, explain the major functionalities of a CI server. Name any three popular CI servers used in the industry.	10	CO1
Q 9	Explain the differences between git reset --soft, git reset --mixed, and git reset --hard with usage, effects, and one example of each. OR What is rebasing and its usage? Also explain: - How rebasing works in the following two cases: - Rebasing a feature branch onto the main branch - Rebasing the main branch onto a feature branch - Discuss which rebasing approach is safe and which should generally be avoided, along with clear reasons. Support your explanation with Git commands, examples, or diagrams wherever necessary.	10	CO2, CO3
SECTION-C (2Qx20M=40 Marks)			
Q 10	Explain the Agile methodology in software development. In your answer, describe its core principles and name three frameworks commonly used in Agile. Then, explain Scrum in detail, including its key roles, artifacts, and events.	20	CO1
Q 11	Draw a Gitflow workflow diagram including: main, develop, feature, release, hotfix. Explain purpose, creation, merging of each. Write the exact Git commands used to handle an urgent production update by creating a hotfix branch and merging it later with the required branches. In this scenario, you are required to make a one-line code change and commit (instead of fixing a bug, just add a .txt file and commit). OR <pre> A --- B --- C --- D (main) \ E --- F --- G (feature/login) </pre> In the above example, the main branch has four commits: A, B, C, and D. The feature/login branch has five commits: A, B, E, F, and G. Perform the following operations on the given commit structure: - Merge the feature/login branch into main. Write the exact Git commands and draw the updated commit graph after the merge for both branches separately. - For the same initial example before merging, rewrite history using git rebase so that the feature/login branch appears as if it was originally created after commit D. Write the required commands and draw the updated commit graph after rebasing. - Explain merge conflict. When a conflict occurs during a merge, describe how it is resolved and write the Git commands required to complete the merge after resolving the conflict.	20	CO2, CO3