**INF 502: Software Development Methodologies**

**Homework 1**

**Part 1:**

**Question 1:**

* **Git branch:** Here are the branches on `hands-on` folder.

**A black screen with green text

AI-generated content may be incorrect.**

* **Git checkout & Git log:** Here are the branches and one line structure of commits for each branch.

**A screenshot of a computer program

AI-generated content may be incorrect.**

**Question 2:**

`git log –graph –all` commit tree for `feature/calculate\_this` branch:

A screenshot of a computer program

AI-generated content may be incorrect.

`git log –graph –all` commit tree for `main` branch:

A screenshot of a computer program

AI-generated content may be incorrect.

The `git log --graph –all` visualization revealed the branching and merging structure of the repository. From the output, I could clearly see how the `feature/calculate\_this` branch diverged from main at a specific commit, allowing independent development. The commit tree also illustrated the linear history of main alongside the parallel development on the feature branch. The graph provided a clear view of commit ancestry, making it easy to trace where branches split and how they might later be merged, which is valuable for understanding project history and collaboration workflows.

**Question 3:**

Here is the git diff statement between `main` and `feature/calculate\_this` branches. Here is the summary of the diff which we see between the two branches:  
  
The changes introduced in the feature/calculate\_this branch includes modifying the A.py file so that the function now prints the two input numbers along with their computed sum, replacing the earlier version that only printed the numbers side by side. Additionally, a previously existing comment about calling the function was removed. In the B.py file, a new comment line was added to demonstrate the use of the decorated log command (git log --graph --decorate --all). Overall, these updates enhance the functionality of A.py by providing clearer output and add documentation in B.py for demonstrating Git features.

A computer screen with text

AI-generated content may be incorrect.

**Question 4:**

Command sequence to merge `feature/calculate\_this` to `main` branch.

1. Go into the `feature/calculate\_this` branch first by using `git checkout feature/calculate\_this`.
2. Rebase the feature branch with `main` branch using `git rebase main`. You might face some merge conflicts, resolve them accordingly.
3. Now go back to `main` branch using `git checkout main`.
4. Merge the feature branch into main branch using `git merge feature/calculate\_this`.

**Question 5:**

Command sequence to create a new branch called math.

1. Check if you are in the main branch by using `git status`.
2. Make sure you have the latest main branch code by using `git pull origin main`.
3. Now create and checkout to the new branch using this command `git checkout -b math`

**Question 6:**

Editing B.py file to add the source code from homework pdf.

A screenshot of a computer

AI-generated content may be incorrect.

**Question 7:**

Command Sequence to commit my changes to B.py file.

1. Check the status of changes in your math branch by using `git status`.
2. Commit my changes to math branch using `git commit -a -m “Adding changes to B.py file”.

**Question 8:**

Command sequence to move to main branch and make changes to B.py file and commit my changes.

1. Check status to check if there are no un staged changes math branch by using `git status`.
2. Move back to main branch by using `git checkout main`.
3. Make the changes as mentioned in the homework pdf.
4. Commit the new changes to main branch by using `git commit -a -m “ Adding changes to B.py file in main branch”`.

A screenshot of a computer program

AI-generated content may be incorrect.

**Question 9:**

Command sequence to merge the math branch into main branch.

1. Check if there are any un staged changes in main branch by using `git status`.
2. Merge math branch into main branch by using `git merge math`.
3. We see merge conflicts since we made changes to B.py file in both the main and math branches.

A screen shot of a computer

AI-generated content may be incorrect.

**Question 10:**

Command sequence to abort the merge.

1. We can abort the merge while having merge conflicts by using ` git merge –abort`.

A screenshot of a computer

AI-generated content may be incorrect.

**Question 11:**

Steps to manually merge B.py file, once we are at the stage of merge conflict.

1. Find the lines like <<<<, ====, >>>> which shows local commits and incoming commits, like the screenshot shown below.

A screen shot of a computer

AI-generated content may be incorrect.

1. Once you remove the unnecessary code lines to keep what is necessary, commit the file changes with `git commit -a -m “Merge branch ‘main’”`.
2. You can verify the commit and changes with `git status` and `git log`.

**Question 12:**

Once the above commit is made and log is checked, you can also check the graph using `git log –graph –all`, here is the output for the graph:

```  
sarojk@Mac ~/D/N/F/I/H/hands-on (main)> git log --graph --all

\* commit fe45368ef07bb91a39522a684cd9f7e84a3f3d4f (HEAD -> main)

|\ Merge: 2c3adeb fe8dfaa

| | Author: Saroj Kaashyap Pokkula <sarojk@incentsoft.com>

| | Date: Mon Sep 15 23:12:21 2025 -0700

| |

| | Merge branch 'math'

| |

| \* commit fe8dfaa66df2ee15ae74892ca3d2835963b6151c (math)

| | Author: Saroj Kaashyap Pokkula <sarojk@incentsoft.com>

| | Date: Mon Sep 15 23:02:04 2025 -0700

| |

| | Adding changes to B.py file

| |

\* | commit 2c3adebf5d6d114f49f6e8fe11cba9e7da1cb9f6

|/ Author: Saroj Kaashyap Pokkula <sarojk@incentsoft.com>

| Date: Mon Sep 15 23:04:25 2025 -0700

|

| Adding changes to B.py file in main branch

|

\* commit 9c746d108ee6d265954bafaa8c40d3aadc12b660

| Author: Kyle N. Winfree <kyle.winfree@nau.edu>

| Date: Mon Sep 15 14:13:41 2025 -0700

|

| made a change to B.py

|

| \* commit 9e4af1e0c2199291ecb3142cdb425028c44e7c1e (feature/calculate\_this)

| | Author: Kyle N. Winfree <kyle.winfree@nau.edu>

| | Date: Mon Sep 15 14:13:11 2025 -0700

| |

| | And this features is done!

| |

| \* commit 528f888b7b15f5055253140684d6868824afface

| | Author: Kyle N. Winfree <kyle.winfree@nau.edu>

| | Date: Mon Sep 15 14:13:01 2025 -0700

| |

| | this should print the inputs AND print the sum of the two

| |

| \* commit e7ecc393b9fb370b4f3a498254e87d4edf5c6e37

|/ Author: Kyle N. Winfree <kyle.winfree@nau.edu>

| Date: Mon Sep 15 14:12:51 2025 -0700

|

| this should print the inputs

|

\* commit ceed1eb23236449a9cc0193b4442e68f86f685a9

| Author: Kyle N. Winfree <kyle.winfree@nau.edu>

| Date: Mon Sep 15 14:12:31 2025 -0700

|

| a few more basics...

|

\* commit 482dd19b6069f0d4d49d77d2f226377b22f4c4db

| Author: Kyle N. Winfree <kyle.winfree@nau.edu>

| Date: Mon Sep 15 14:12:21 2025 -0700

|

| I added some comments to, at least somewhat, outline the eventual contents of each python file.

|

\* commit 01ab7226c6b71a35ee453d132eb5032dfa1904be

| Author: Kyle N. Winfree <kyle.winfree@nau.edu>

| Date: Mon Sep 15 14:12:11 2025 -0700

|

| And now for the skeleton of B too. Granted, these could have both been added in a single commit, right?

|

\* commit b1b9e94d5a3d69de824f3c0fc369c43200a5e95b

Author: Kyle N. Winfree <kyle.winfree@nau.edu>

Date: Mon Sep 15 14:12:01 2025 -0700

This is just a skeleton file of A first.

```