Project I: Software Implementation and Unit Testing

Assignment #1

Overview

This assignment should be done individually. Do your own work and do not share your work with others. Sharing work is an academic offense and is subject to penalty. Be aware that source code and documents are automatically checked by eConestoga against every other student's work in the course. Academic offenses will be reported to the College Registrar.

This Assignment serves as an introduction to the version control technology GitHub. It will require a bit of research into what GitHub is and how GitHub works. There are multiple ways to communicate and use GitHub:

- GitHub.com
- GitHub Bash Commands (https://github.com/git-guides/install-git)
- GitHub Desktop (https://desktop.github.com/)
- GitHub plug-in with Visual Studio (https://visualstudio.github.com/)

You may use any of these methods, however, the instructions in this document will focus on using the GitHub Desktop and Visual Studio approaches. Asynchronous training material can be found on GitHub and YouTube, I strongly suggest some personal training above and beyond what the professor walked through in class:

- https://github.com/
- https://docs.github.com/en
- https://www.youtube.com/results?search_query=github

What to do

Step 1: Create a GitHub Account (If required)

If you have not done so already, please create a personal account at https://github.com/. Feel free to use either your personal or eConestoga email address.

NOTE: You do not have to create a special account for this course.

Step 2: Clone the Assignment Repository

Once you have created a GitHub account your first step is to clone the CSCN71020_A1 project from your professors' repository. This can be done using GitHub.com as follows:

1. Log into your GitHub.com account

2. Select “Repositories”

3. Select “New”

4. Click on “Import a repository”

5. Under “Your old repository’s clone URL” cut and paste the following link:

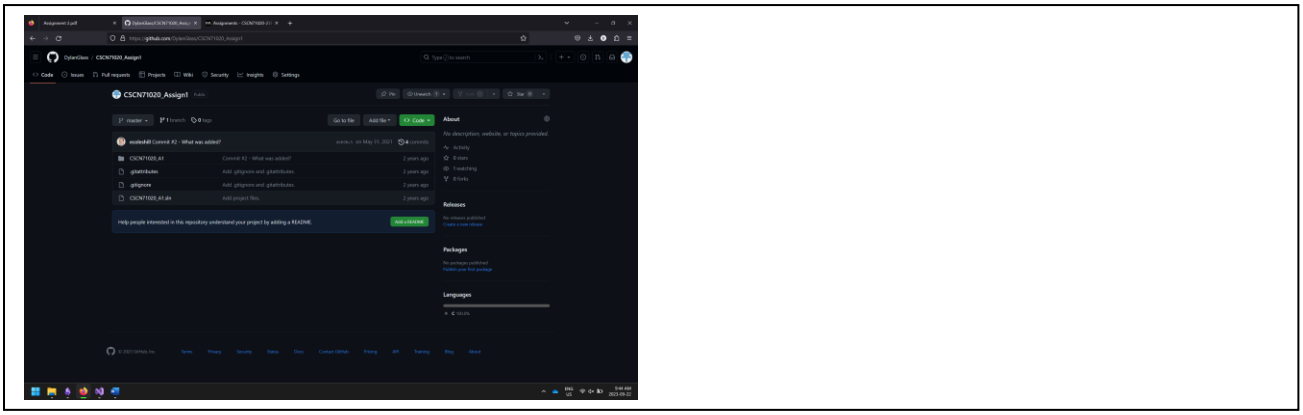
https://github.com/gurpreet-conestoga/gkaur1_CSCN71020_A1

6. Under “Your new repository details” give it the name “CSCN71020_Assign1”

7. Public/Private is up to you.

8. Click “Begin Import”

Take a screenshot of your GitHub.com showing the new repository created on your account. Upload that screenshot below.



Step #3: Create a Local Clone

Now that you have created a clone of the repository in your own GitHub profile you can begin working on it. The first step is to create a local clone in Visual Studio. This is done as follows:

1. Open Visual Studio 2019

2. Select “Clone a Repository”

3. Type in the URL to the path in your repository (NOT the professors). Here is how you find it:

    a. Log into GitHub.com

b. Click on “Repositories”

c. Select the “CSCN71020_Assign1” repository you just created

d. Click on “Code”

e. Click on the clipboard icon next to the URL

4. Select the “Path” where you wish the clone to be placed using the “…” button

5. Click “Clone”

This will create a directory in the selected “Path” with all the GitHub and Visual Studio solution files. You can now open the Visual Studio solution by double-clicking on the *.SLN file.

Step 3: Review History of Development

Open the project solution in Visual Studio 2019. You will notice there is a single source file called “Source.c”. This is a basic C programing file that contains a main and several supporting functions. The program simply asks the user to add/subtract two numbers and displays the answer.

Your first task is to view the history of the source code development. In Visual Studio view the Git history by selecting:

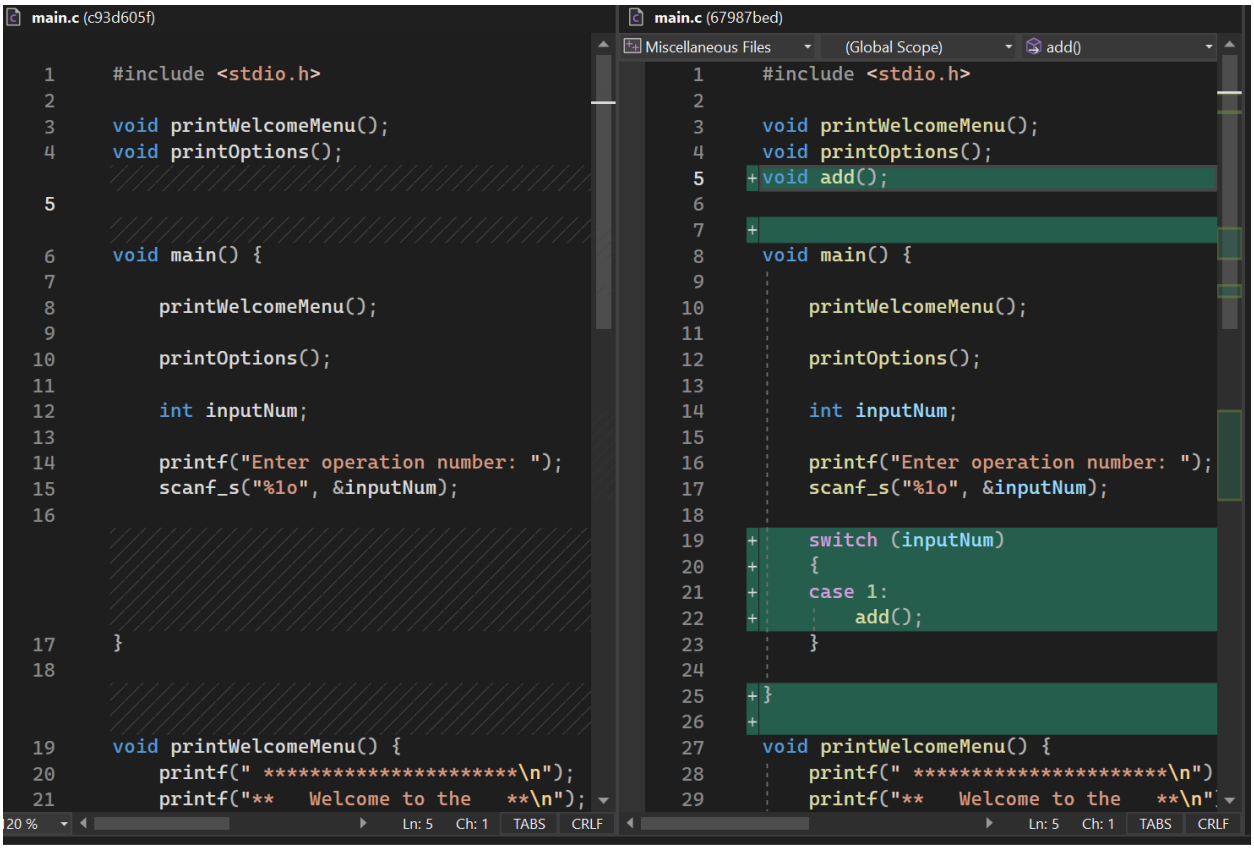
- “Git -> View Branch History”

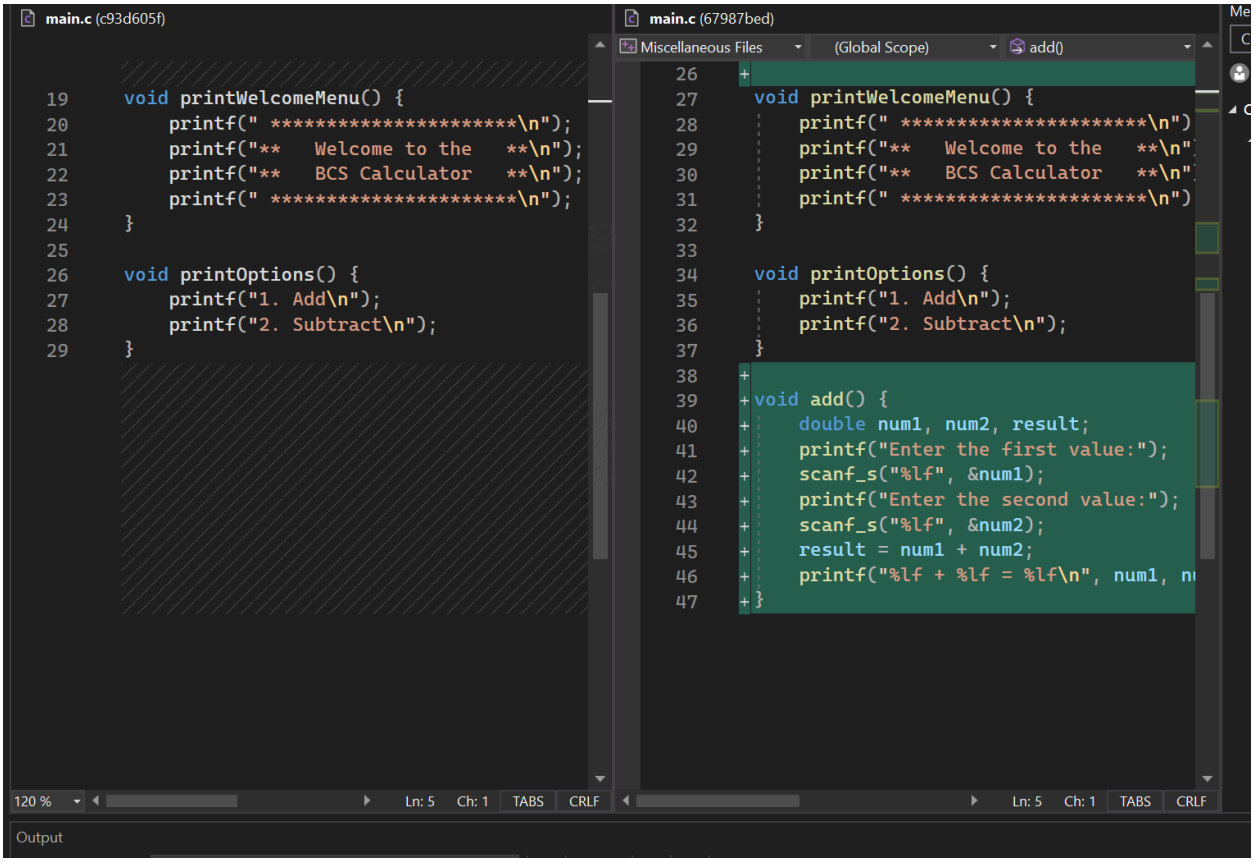
This will open a window that will show you the entire history of the master branch which you have cloned. There should be 4 commits in your history. In the text boxes below, write down what changes were made in the “Commit #1 – What is it?” and “Commit #2 – What was added?” pushes by performing the following:

- In this Branch History window right mouse-click on the commit you are interested in
- Select “View Commit Details”
    - o This will open a new tab window somewhere in the IDE (probably wherever your Solution Explorer is located)
    - o You will see under “Changes” a list of all the files that were modified by this commit. In this case there is only main.c
- Right mouse-click on main.c and select “Compare with Previous…”
    - o This will open a new comparison split window showing the old version, new version and highlighting the additions and deletions to the file

Commit #1 – What is it?

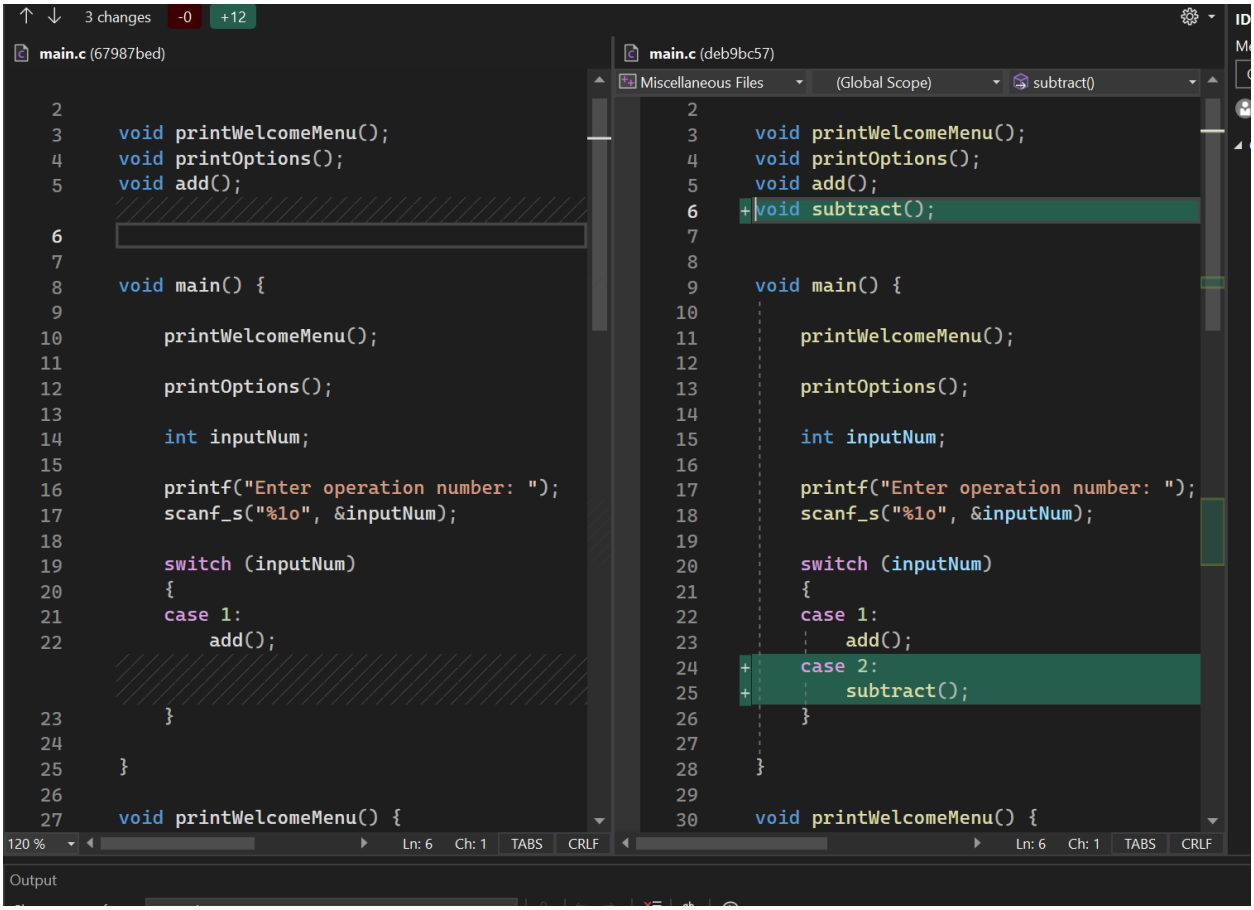
The addition function was added to the top of the file. It was then included in a switch to allow for its selection. It was then written out at the bottom to accept two numbers, add them, and output the full equation with the result.

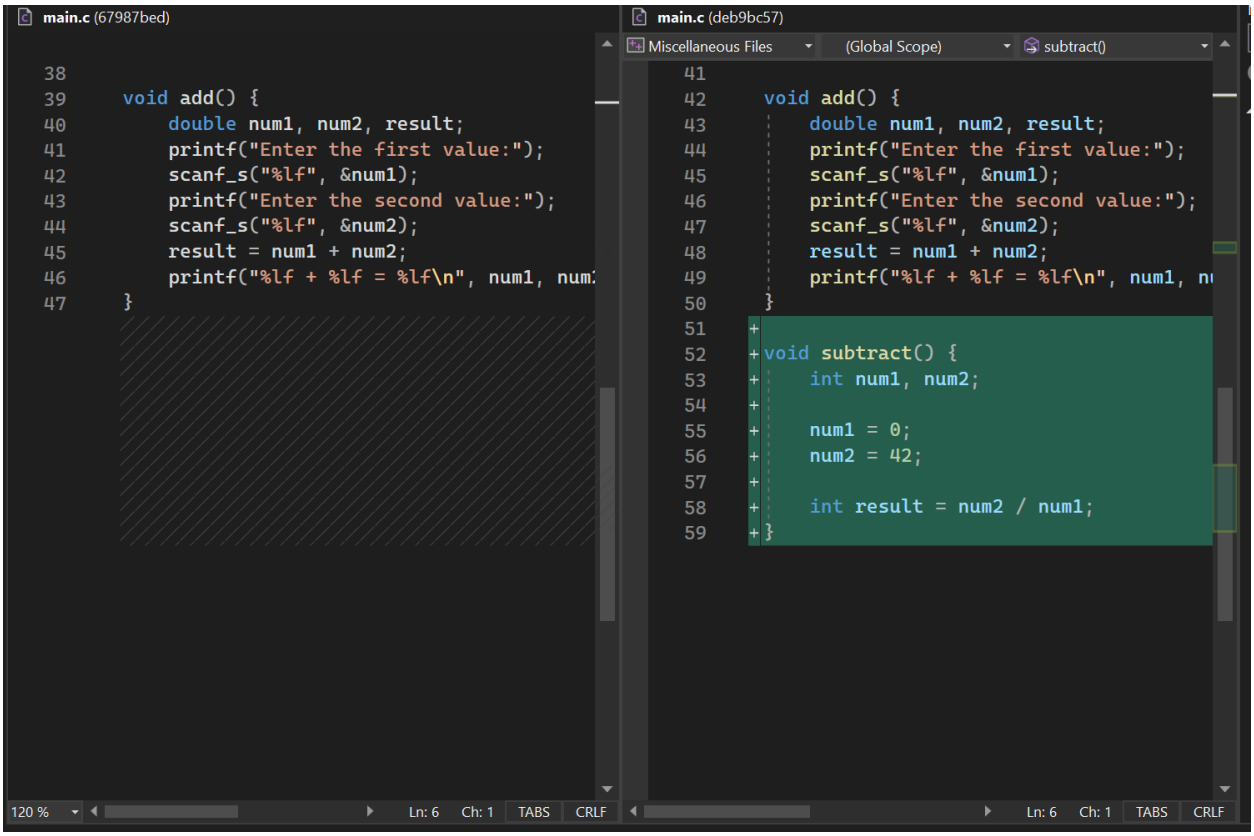




Commit #2 – What was added?

The subtraction function was added at the top. It was then called in the switch. Finally it was given functionality at the end of the script, however, it is currently using two hardcoded numbers and dividing these numbers.


3 changes -0 +12
main.c (67987bed)
main.c (deb9bc57)
Miscellaneous Files
(Global Scope)
subtract()
void printWelcomeMenu();
void printOptions();
void add();
void subtract();
void main() {
printWelcomeMenu();
printOptions();
int inputNum;
printf("Enter operation number: ");
scanf_s("%1o", &inputNum);
switch (inputNum)
{
case 1:
add();
case 2:
subtract();
}
}
void printWelcomeMenu() {
120 %
Ln: 6 Ch: 1 TABS CRLF
Output



Task 2 [2 Marks]

It appears some mistakes have occurred while working on this file. Which commit introduced error(s) [1 Mark]? What is or are the error(s) [1 Mark]?

Commit #2 - What was added? Has an error in the subtract function. This function currently divides the two numbers rather than subtracts them. The subtract function is also trying to divide by zero.

Step 4: Revert, Fix, Commit and Push Changes

Have a close look at the subtract() function. Is it wrong? The answer is yes. Your job is to revert the code, repair it (by writing it correctly, using the user's input) and commit your changes.

(a) Revert the source code

- To do this open the Git branch history window again and right mouse-click on the commit you wish to revert back to. In this case, "Commit #2 – What was added?"
- Select "Revert"

NOTE: At this point you will see you have 1 outgoing Push. That's okay. It is because Visual Studio and the Github plugin sees you have made changes.

(b) Re-Write subtract()

- Open the main.c file and you will see all the code related to subtract() has been removed
- Re-write the function properly. Make sure you update main() as well.
- Test your code and make sure everything is working

(c) Commit your changes

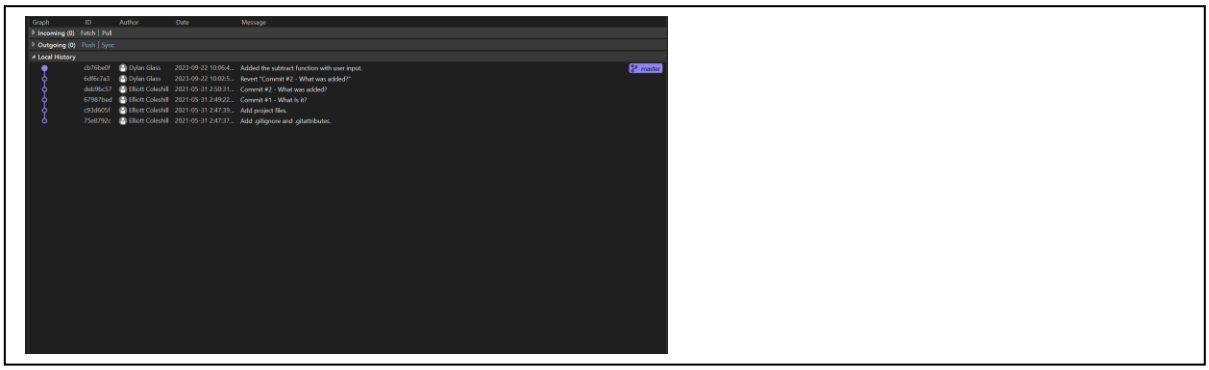
- Select “Git -> Commit or Stash”
- The Git Changes tab will appear again (look where your solution explorer is located)
- Type in a commit comment that makes sense for this fix
- Click “Commit All”

NOTE: At this point you will see you have 2 outgoing Push items.

(d) Push your changes to your GitHub account

- Click “Git View Branch History”
- You will see your new commits located in “Outgoing” and is showing above “master”
- Click on the “push” hyperlink to push (or publish) your changes to your GitHub.com master

(e) Take a screenshot of your Visual Studio Git Branch History window and upload the screenshot to the box below.



Step 5: There is one more bug

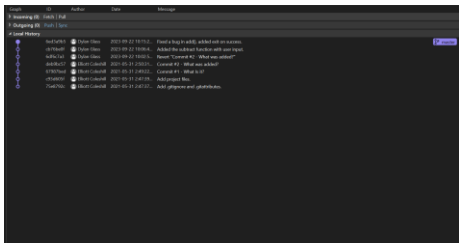
There is one more bug in this code. Run the code again, select add() and see what happens. You job is to fix it, commit the changes and push the changes back to your GitHub repository.

Review the code and in the box below describe what the problem is, and fix was.

Commit and push your changes to your GitHub repository.

Take a screenshot of your Visual Studio GitBranch History window (again) and upload the screenshot to the box below showing your committedand pushed changes.

When the user completed an addition action the program would then start the subtraction action instead of exiting.



Rubric

This assignment will be marked based on the following rubric information.

• Completion of the PDF form and submitted in PDF format (2 marks)

• Step 2 – successful clone and setup of visual studio IDE (3 marks)

• Step 3 – quality of your explanations of the history (5 marks)

• Step 4 – quality of subract() fix and Revert, Fix, Commit and Push actions (5 marks)

• Step 5 – Detection, solution and Fix, Commit and push actions (5 marks)

Total 20 marks.

What to Hand In

Upload the following files to eConestoga using the Assignment #1 link

• **Your main.c file**

• **A copy of this PDF form properly filled out**

Do not reformat this form in any way. Just fill it out, save it and upload it. Modifying this form will result in a 10% penalty.

Do not upload your Visual Studio solution space. Just the source code files listed above. Uploading

the entire Visual Studio solution space will result in a 10% penalty.