|  |  |
| --- | --- |
| **Testing** | **Evidence** |
| In version one I set up the input of the integers and making sure I could store and print them without any issues. |  |
| In version two, I added the other half of the input – the menu options. I tested this to make sure that it was storing and printing the input correctly. |  |
| In version three, I took the option input and made the program perform the relevant calculation and output the result.  I also added some comments to explain the code, since it was starting to get a bit more complex. |  |
| In version four, I added an infinite while True loop that only gets broken if the user enters “q” to quit. This means they can perform multiple calculations with the same integers without having to restart the program each time. |  |
| In version five I put certain parts of the code into functions. This included the choice of menu option, as well as all of the operators (addition, etc). Even though this takes up more lines, it’s easier to understand and clearer, and if I wanted to add something like a history list of past calculations, I could append the variable to a list.  I tested the menu input and the different options to make sure that everything was still working properly. |  |
| In version six I started with error handling by creating a function that forces the user to enter an integer. If they try to enter anything else (string, float, etc.) it requests that they enter it again and will continue to do so until the user enters a valid input. This gets called twice – once for each integer. |  |
| In version seven I added error handling for when the user inputs which menu option they want. Originally I was going to do this by putting if opt\_choice in MENU\_OPTIONS == True, however this would not work because MENU\_OPTIONS is a 2D list. So, I did this in the while loop instead. If the input is not one of the options in the if/elif/else branch, then it must be invalid. Therefore the prompt is printed and the program returns to the start of the while loop, where they are asked to enter the input again. |  |
|  |  |
|  |  |
|  |  |