Task 3.2P Answer Sheet

Name: Lorraine Becker
Student ID: 104584773

1. In 2.2P, how many Counter objects were created?

There were two Counter Objects created

2. Variables declared without the "new" keyword are different to the objects created when we call "new". Referring to the main method in task 2.2P, what is the relationship between the variables initialised with and without the "new" keyword?

   With: For creating a 'new' instance of 'Counter' object
   Without: For declaring the variable, but it does not assign any value to it.

3. In 2.2P, explain why resetting the counter in myCounters[2] also changed the value of the counter in myCounters[0].

Because both myCounters[0] and myCounters[2] are on the same Counter instance. This means that any changes made through one reference will be seen through the other reference as well.

4. The key difference between memory on the heap and memory on the stack is that the heap holds "dynamically allocated memory". What does this mean? In your answer, focus on the size and lifetime of the allocations.

It means that memory on the heap is larger in size and is more flexible than memory on the stack. In terms of lifetime, it is less predictable because it is the programmer's responsibility to release memory after use. Otherwise, it leads to memory leaks.

5. Are objects allocated on the heap or the stack? What about local variables?

Objects are allocated on the heap in order to hold the object's data, while local variables are allocated on the stack.

2. 6. What does the new() method do when called for a particular class, and what does it return?

   When new() method is called for a particular class, it allocates the required memory on the disk and calls for the constructor and returns the reference to the objects

7. Assuming the class Counter exists in my project, if I wrote the code "Counter myCounter;" (note there is no "="), what value would myCounter have? Why?

It would have the value 'Null' because it has not been assigned an actual instance of the Counter class.

1. Based on the code you wrote in task 2.2P, draw a diagram showing the locations of the variables and objects in main and their relationships to one another.