Task 3.2P Answer Sheet

Name: Hai Nam Ngo
Student ID: 103488515

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

There are 2 Counter objects were 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?

When we declare a variable without using the "new" keyword, we are creating a reference variable that can hold an object's memory address. This variable does not create a new object in memory, but rather refers to one that already exists.

When we use the "new" keyword, we are creating a new object in memory. The "new" keyword initialises the object's instance variables and allocates memory for it. The variable is then assigned the memory location of this newly created object.

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

This is because myCounters[2] and myCounters[0] point to the same object.

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.

Dynamically allocated memory is memory that a program sets aside while it's running. The heap is where this memory is kept.

Heap Memory:

- Size: You can ask for different amounts of memory as needed.
- Lifetime: It can last beyond the function that created it.

Stack Memory:

- Size: It's fixed and determined when the program is written.
- Lifetime: It's automatically freed when the function using it ends.

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

Objects are allocated on the Heap and Local variables are allocated on the Stack

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

The new keyword is used to create a new instance of a class. When using the new keyword with a class, it allocates memory for a new object of that class and returns a reference to that object.

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?

When we declare a variable without initializing it, as in the code "Counter myCounter;", the variable is assigned a default value based on its type. The default value is null.
So, in this case, the variable myCounter would have a value of null, because it's a reference to an instance of the Counter class that hasn't been assigned an actual instance. This means myCounter doesn't currently point to any valid object in memory.

8. 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.