Task 3.2P Answer Sheet

Name: Shah Nawaz Chowdhury
Student ID: 103830682

1. In 2.2P, how many Counter objects were created?
   3

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?
   Variables initialised with a new keyword creates a new object, which is allocated a new memory in the heap. Variables initiating without a new keyword uses an already existing object in the heap.

3. In 2.2P, explain why resetting the counter in myCounters[2] also changed the value of the counter in myCounters[0].
   Because both counters are referring 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.
   The memory allocated to any object can constantly change its size according to the code. The memory is also allocated when a function that requires it is called, and deallocated when the function exits.

5. Are objects allocated on the heap or the stack? What about local variables?
   Objects are in the heap, local variables are in the stack and if the local variable is an identifier for an object, then it simply references the object in the heap.

6. What does the new() method do when called for a particular class, and what does it return?
   It creates a new object using the blueprint of that class and it returns an object that has access to the public variables and methods, as well as its properties.

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?
   Null, because the variable has only been initialised with the data type 'Counter' without assigning a value yet.

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.

Stack	Heap

Stack:

Main
_____

myCounters

Heap:

Counter
----
name = counter 1
count = 0

Counter
----
name = counter 2
count = 0

Array of counters

[counter1, counter 2, counter 3]