# Task 3.2P Answer Sheet

Name: Hoang Bao Phuc Chau

Student ID: 103523966

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

There were 3 objects created.

## 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?

The variables with and without the “new” keyword (myCounters[0], myCounters[1] and myCounters[2]) both reference the objects in the class Counter. However, myCounters[0] and myCounter[2] refer to the same object, while myCounters[1] is separated from two variables above.

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

Because myCounters[2] and myCounters[0] both contain reference to the same object, so that any changes made to the object through myCounter[0] and myCounter[2] will be reflected in both variables.

## 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 on the heap is dynamically allocated during runtime. Compared to stack, heap holds memory which has variable size and life time manually controlled by the programmer. Also, the memory on the heap can be used to store data which has ability to access across the program.

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

The objects are stored on the heap, and the local variables are stored on the stack.

1. What does the new() method do when called for a particular class, and what does it return?  
     
   When the new() method is called, it will allocate the memory from the heap for that object. Also, it will return a pointer to the object created.
2. 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?

myCounter variable will have a value of “null” as the C# compiler will set variables without initialization to that value.

## 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.

Main

…

Stack

Heap