

## Task 3.2P Answer Sheet

Name: Xuan Tuan Minh Nguyen

Student ID: 103819212

1. In 2.2P, how many Counter objects were created?
  - In total, there are two Counter objects were created, which 2 new Counter objects and one reference to the Counter at first index
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?
  - The variable that is created without the “new” keyword is considered as the reference variable (or pointer on some low-level languages such as C, C++, etc.) such when this variable changes the variable that this variable referenced to also changes. Whilst the one that is created with the “new” keyword is considered as creating an instance of a class, which involves memory allocation and construction of the object.
3. In 2.2P, explain why resetting the counter in myCounters[2] also changed the value of the counter in myCounters[0].
  - Since myCounters[2] is the reference of myCounters[0], or the pointer that points directly into myCounters[0]. So when the value of myCounters[2] changes, myCounters[0] also changes its value.
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 difference of stack and heap memory is that, while the stack has a fixed memory size (usually around 1-8MB) and will be deleted after the program is completed. The heap memory, just by the term “dynamically allocated memory” is not fixed and will depend on the use of memory of that program. The heap memory is the memory that programmer has to manually allocate and de-allocate the memory after using it, otherwise the program will encounter the error of “Memory leaks” (this only applied for some low-level programming languages (C++, C, Go, Rust, unsafe C#) that allows user to manually takes control over the memory. But in high-level programming language, there is a garbage collector that automatically deallocate and delete the unused or finished heaps).
5. Are objects allocated on the heap or the stack? What about local variables?
  - Objects are allocated on the heap, while the local variables are located on the stack.
6. What does the new() method do when called for a particular class, and what does it return?
  - When the new() method is called, the program will allocate the required memory of that particular class onto the heap memory, after that it will call

the constructor class and return the reference of 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?
  - The value that it could have when only DECLARING (Counter myCounter), not INITIALISE (Counter myCounter = new Counter()) is null. Because it does not refers to any objects or any valid memory location of the object.

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.

