Task 3.2P Answer Sheet

Name: Luan Nguyen
Student ID:

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

A total of 2 counters

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?

In the main method in 2.2P, 3 variables are declared but 2 of them are the myCounter[0] and myCounters[1] are assigned to as new instances of the *Counter*. However, myCounters[2] was assigned to the same instances as the myCounters[0]. So, any changes that are made to the myCounters[0] or myCounters[2] will affect the other. But, changes to myCounter[1] will not affect other counters

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

As mentioned above, myCounters[2] was assign to the myCounters[0] so any changes to myCounter[0] will change the myCounter[2]

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 the heap is used for dynamically allocated memory, which allows programs to allocate and deallocate memory as needed during runtime, without being limited by the fixed size of the stack. The size of allocations on the heap can vary and their lifetime is determined by the program's code, but they persist until explicitly freed.

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

Objects are allocated on the heap, while local variables are typically allocated on the stack.

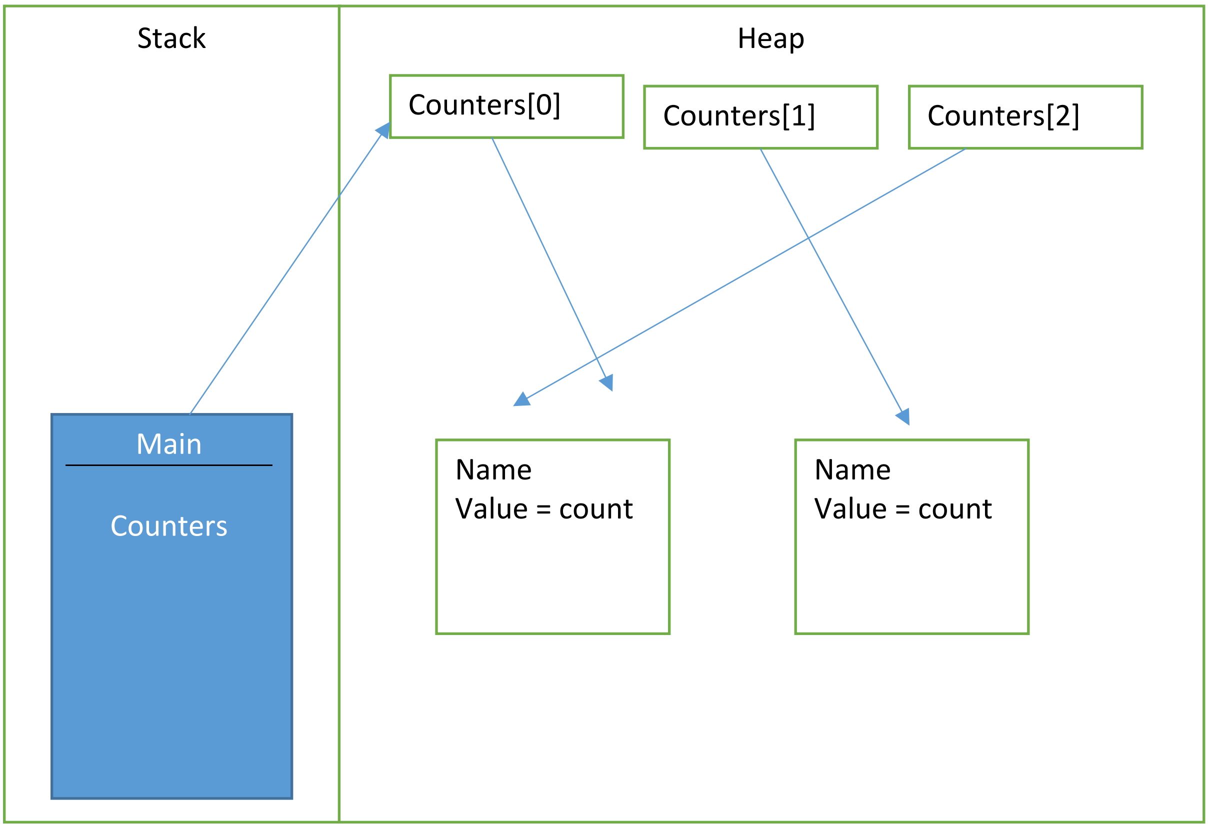
6. What does the new() method do when called for a particular class, and what does it return?

The new() method is a constructor that creates a new object on the heap, initializes its fields to default values, and returns a reference to the 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?

So, without “=” , no value would be assigned to myCounters which makes it become null and can't be used until it's assigned a valid value.

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
Counters[0]
Counters[1]
Counters[2]
Main
Counters
Name
Value = count
Name
Value = count