Swinburne University of Technology

COS20007 Object Oriented Programming

The Stack and Heap

PDF generated at 12:01 on Friday 25th August, 2023

Produced by OOP Submission Formatter

Task 3.2P Answer Sheet

Name: Tran Vu Duc
Student ID: 104175614

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

   There are 2 Counter objects

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 call "new" for myCounters[0] and myCounters[1], it creates new object of "Counter" class and return references to them.
   myCounter[2] is initialized with the value of myCounter[0]. Therefore, myCounter[2] will point to the same object of myCOunter[0].
   The relationship between these is both ways reference the same object in memory.

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

   Because both is pointing 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.

Memory on the heap	Memory on the stack
Allocated at runtime. The size is not fixed, can be changed during the runtime. Data exists until the program is terminated. Lifetime can be determined at runtime.	Allocated at the compile time. The size is fixed. Lifetime is limited of function or block code that created it.

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

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

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

   It allocates a number of memory for an object of the class based on the size of the

data needed, returns the address of that object for the variable to store. It also ensures that the constructor is called to initialize the project

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 would be null. There’s no object which has been created of assigned to myCOunter

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.