SIT102 – Introduction to Programming

Answers for 9.MP: P Final Milestone Checkpoint with Self-reflection

Final Milestone for your Introduction to Programming journey

Part A – A quick self-check on your Pass tasks completeness

Have you addressed the given feedback from your OnTrack tutor to your on-time submissions of your weeks 9 **Pass** tasks – 9.1P and 9.2P?

Yes

Part B – Self reflection

It is the ***final milestone*** task in relation to 9.1P task (covered weeks 1-8 topics) and 9.2P about pointers. However, this ***does not*** imply the following weeks 10, 11's topics of SIT102 are less important in building your programming skills. Though there are no practical programming Pass tasks for Week 10 Secure Coding and Week 11 Conclusion – Programming Paradigms, they include essential knowledge wrapping up the programming concepts and artefacts covered in weeks 1 to 9.

It is always a good habit to learn, explore, and consolidate your knowledge and skills regularly. Your learning outcomes will also be fully reflected in your portfolio with this P milestone task and your learning summary report (including your justifications on your achievements for all weeks' contents).

Note: Your reflection in each milestone task are the building blocks for your learning summary report for your portfolio submission.

Provide your reflection in 80 – 100 words (or more) to elaborate your understanding on the associate concepts in relation to **Data management** learnt in this unit.

- What is your understanding on the differences between using stack and using heap region in memory management for storing data in a program?
- What are the programming artefacts and techniques for modelling and managing different data structures and storage for variables? Recall what have you experienced till now in this unit – what have you done in tasks to demonstrate your understanding on them?
- What is your understanding on how pointers work in a C++ program to reference and dereference data? What have you demonstrated in your 9.2P Data Pass by Pointers task?
- Think about how the different discussed way(s) in SIT102 of managing data during software development contribute(s) to a secure software application.

1 The stack is a linear data structure and the heap is a hierarchical data structure. Stack memory is not fragmented, but heap memory can be fragmented when a block of memory is first allocated and then deallocated. The stack only accesses local variables, while the heap provides global access to variables.

2 By using enumerated and structured custom data types and using static and dynamic arrays to store lists of values, the data can be accessed in different ways by value/reference/pointers after drunkenness.

3 A pointer in C++ is a variable that holds the memory address of another variable. A reference is an alias for an already existing variable. Once a reference is initialized to a

variable, it cannot be changed to refer to another variable. Hence, a reference is similar to a const pointer. **Task 9.2, Part A, uses the reference method to get the address of the variable it points to. And in part B, use dereferencing to access the data pointed to by the pointer for data exchange.**
4 Using data validation helps to ensure that the input data is reasonable and valid, for example, using integer, then the input number cannot have a decimal point and must be an integer. And using memory management will avoid buffer data overflow. Setting user access rights will prohibit unauthorized access to the data.

End of document (2 parts)