COS20007

Object Oriented Programming

Learning Summary Report

Marco Giacoppo

104071453

Self-Assessment Details

The following checklists provide an overview of my self-assessment for this unit.

Self-Assessment Statement

	Pass (D)	Credit (C)	Distinction (B)	High Distinction (A)
Self-Assessment		✓		

Minimum Pass Checklist

	Included
Learning Summary Report	✓
Test is Complete	✓
C# programs that demonstrate coverage of core concepts	✓
Explanation of OO principles	✓
All Pass Tasks are Complete	✓

Minimum Credit Checklist (in addition to Pass Checklist)

	Included
All Credit Tasks are Complete	✓

Minimum Distinction Checklist (in addition to Credit Checklist)

	Included
Custom program meets Distinction criteria & Interview booked	
Design report has UML diagrams and screenshots of program	

Minimum Low-Band (80 – 89) High Distinction Checklist (in addition to Distinction Checklist)

	Included
Custom project meets HD requirements	

Minimum High-Band (90 – 100) High Distinction Checklist (in addition to Low-Band High Distinction Checklist)

	Included
Research project meets requirements	

Declaration

I declare that this portfolio is my individual work. I have not copied from any other student's work or from any other source except where due acknowledgment is made explicitly in the text, nor has any part of this submission been written for me by another person.

Signature: **Marco Giacoppo**

Portfolio Overview

This portfolio includes work that demonstrates that I have achieve all Unit Learning Outcomes for COS20007 Unit Title to a **Credit** level.

I'm writing to provide a detailed explanation of why I believe this unit deserves a good grade, highlighting my consistent efforts and achievements throughout the course.

First and foremost, I have displayed a thorough understanding of the core principles of object-oriented programming (OOP) throughout the unit. Through the successful completion of tasks and assignments, I have effectively demonstrated my comprehension of crucial concepts such as encapsulation, inheritance, and polymorphism. By applying these principles in practice, I have showcased my ability to design and implement well-structured and reusable code. I also used SplashKit to create drawing program which allows user to create multiple kinds of shapes. These programs demonstrate my ability to apply OOP concepts to provide engaging and visually appealing user experiences. Another aspect worth mentioning, I've also drawn UML (Unified Modeling Language) and Sequence diagrams throughout tasks. Through the creation of the well-structured diagrams, I have effectively communicated relationships, interactions, and flow of information within various software systems.

In addition, I have devoted significant attention to testing and debugging my object-oriented programs to ensure their accuracy and reliability. Employing various testing strategies, including thorough unit testing, I have meticulously validated the functionality of my code. This approach reflects my commitment to delivering high-quality solutions that can withstand rigorous testing scenarios.

Moreover, I have consistently met all deadlines for submitting tasks and assignments throughout the course. Despite encountering challenges and complexities in my coding projects, I have remained determined to overcome errors and inaccuracies, ensuring that my work is completed and submitted on time. This demonstrates my strong work ethic, resilience, and effective time management skills.

In conclusion, I firmly believe that I have met all the Unit Learning Outcomes to a Credit level in the object-oriented programming unit. My consistent efforts, deep understanding of OOP principles, effective application of testing and debugging strategies, and timely completion of tasks and assignments all contribute to the strong case for a good grade. I have actively engaged with the material, seeking to enhance my knowledge and skills in object-oriented programming, and my dedication to the unit's objectives is evident throughout my work.

Task Summary

To demonstrate my learning in this unit, I would like the following tasks to be considered part of my portfolio:

- **1.1P** – Preparing for Object Oriented Programming   **COMPLETE**

- **1.2P** – Object Oriented Hello World   **COMPLETE**

- **2.1P** – In Person Check-in 1 – Tools   **COMPLETE**

- **2.2P** – Counter Class   **COMPLETE**

- **2.3P** – Drawing Program – A Basic Shape   **COMPLETE**

- **2.4P** – Case Study Iteration 1 – Identifiable Object   **COMPLETE**

- **3.1P** – Clock Class   **COMPLETE**

- **3.2P** – The Stack and Heap   **COMPLETE**

- **3.3P** – Drawing Program – A Drawing Class   **COMPLETE**

- **4.1P** – Drawing Program – Multiple Shape Kinds   **COMPLETE**

- **4.2P** – Case Study – Iteration 2 – Players Items and Inventory   **COMPLETE**

- **5.1P** – In Person Check-in 2 – Drawing Program   **COMPLETE**

- **5.2P** – Case Study – Iteration 3 – Bags   **COMPLETE**

- **5.3C** – Drawing Program – Saving and Loading   **COMPLETE**

- **6.1P** – Key Object Oriented Concepts   **COMPLETE**

- **7.1P** – Case Study – Iteration 5 – Tying it Together   **COMPLETE**

- **7.2C** – Case Study – Iteration 6 – Locations   **COMPLETE**

- **9.1P** – In Person Check-in 3 – Case Study   **COMPLETE**

- **9.2C** – Case Study – Iteration 7 – Paths   **COMPLETE**

- **10.1C** – Case Study – Iteration 8 – Command Processor   **COMPLETE**

- **11.1P** – Clock in Another Language   **COMPLETE**

- **T1-1** – Semester Test Fix and Resubmit   **COMPLETE**

Reflection

The most important things I learnt:
- Understanding OOP principles: concepts such as classes, objects, inheritance, polymorphism, and encapsulation
- Designing and implementing classes
- Testing and debugging OOP code

The things that helped me most were:
- Asking my friends to teach me thing I don't quite understand
- The feedback from my tutor when submitting tasks
- Searching on the internet for tutorials

I found the following topics particularly challenging:
- Inheritance and Polymorphism: especially when dealing with multiple levels of inheritance and implementing polymorphic behavior effectively
- Creating Sequence diagram can be a bit confusing
- Abstraction and Interfaces: Understanding abstraction and interfaces and knowing when and how to use them effectively can be challenging

I found the following topics particularly interesting:
- Using SplashKit was interesting for me because it's a new thing for me and it was fun to learn
- Creating games (SwinAdventure) using the core principles of OOP was also exciting, even though some of the higher iteration gave me confusions

I feel I learnt these topics, concepts, and/or tools really well:
- Visual Studio: This was my first time using visual studio, but I feel like I've learned a lot by practicing every day to complete my tasks
- Classes and Objects: I've figured out how to define classes with attributes and methods and create instances (objects) of those classes
- NUnit testing: I feel like I've got the hang of it from creating lots of test cases for my study case tasks

I still need to work on the following areas:
- The overall concept of Inheritance, Polymorphism, Abstraction and Encapsulation
- Applying this into real world scenario
- Practicing on Sequence Diagrams

My progress in this unit was …:
I consistently submit my tasks on time and sometimes engage with the tutor. I feel like I'd need to engage more in the future. Maybe the reason why I didn't engage more often was because I'm a bit scared to ask stupid questions. I had a bit of struggle towards the end of the semester where I couldn't get the code running but I feel like I've done it now. I would continue to apply my techniques to submit tasks on time and use this in the future.

This unit will help me in the future:

Studying Object-Oriented Programming (OOP) will provide a solid foundation for future programming concepts, collaborative development, and industry relevance, enhancing my abilities to build complex software systems and adapt to evolving technologies.

If I did this unit again I would do the following things differently:

- I would try to finish my tasks even faster to make sure I get everything done on time and can focus more on the D or HD tasks
- I would engage more with the tutor and question things that I find confusing

Other…:

Overall, I enjoyed my time studying OOP this semester. I learned a lot of new things here and I'm hoping I can use them in future projects and career opportunities.