COS20007

Object-Oriented Programming

Learning Summary Report

STUDENT NAME: Nguyen Duc Thang
STUDENT ID: 104776473

Self-Assessment Details

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

In addition to the checklists, please append the following supporting appendices to your portfolio:

- **Appendix I:** A list of screenshots showing each task title and the feedback from your tutor upon submission.

This assignment does not count toward the final grade.

1.1P - Preparing for OOP

New Attempt

Due 9 Aug by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf **Available** after 24 Jul at 0:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips **and** Academic integrity

1.1P - Preparing for Object Oriented Programming.pdf

Minimise file preview

Page 1 of 2 ZOOM

Submission

Submitted!

7 Aug at 13:57

Submission details

Download 1.1P-1.pdf

Comments:

MINOR REVISION.
Section B.2.Float example is not clear in which unit measurement. Please consider another example.
The current screenshots have quite low resolution.
Next time, try to use these following approach
https://swinburne.instructure.com/courses/6 pdfs-of-your-code-files-for-task-submissions?module_item_id=4245689

Viet Vo, 7 Aug at 10:10

The submission is acknowledged.
You can include it in your final portfolio summary report

Viet Vo, 7 Aug at 14:00

This assignment does not count toward the final grade.

1.2P - Personalized Hello World program

New Attempt

Due 9 Aug by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 24 Jul at 0:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips **and** Academic integrity

1.2P - Object Oriented - Personalized Hello World.pdf

Minimise file preview

Submission

Submitted!

1 Aug at 17:41

Submission details

Download 1.2P.pdf

Comments:

The submission is acknowledged.
You can include this one in your final portfolio summary report.

Viet Vo, 7 Aug at 13:47

This assignment does not count toward the final grade.

2.1P - *In Person* Check-in 1 - Checked-in tools and Prior Experience Survey

New Attempt

Due 16 Aug by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 1 Aug at 0:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

This is the in-person check-in task. You must meet your tutor in person. Please read the Deadline management tips **and** Academic integrity

Check-in tasks require a discussion with your tutor during your lab or at the help desk before they can be signed off for submission acknowledgment. If you fail to show up for the check-in tasks, you must provide special consideration, such as a medical certificate, to the convenor for approval within two weeks of the task announcement. If you fail to show up and fail to provide special consideration, your portfolio will not be eligible for assessment.

2.1P - In Person Check-in 1 - Tools-2.pdf

Minimise file preview

Submission

Submitted!

7 Aug at 13:39

Submission details

Download 2.1P - In Person Check-in 1 - Tools Answer Sheet.pdf

Comments:

The submission is acknowledged.
You can include this submission in your final portfolio summary report.

Viet Vo, 7 Aug at 13:41

This assignment does not count toward the final grade.

2.2P - Counter Class and Arithmetic Overflow-checking

New Attempt

Due 16 Aug by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 1 Aug at 0:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips **and** Academic integrity

2.2P - Counter Class and Arithmetic Overflow.pdf

Minimise file preview

Page 1 of 7 ZOOM

Submission

Submitted!
15 Aug at 20:54
Submission details
Download 2.2_latest-1.pdf

Comments:
MINOR REVISION
Please assign the count to be the Max Value in the int type and use the checked and unchecked statements as highlighted in the Hints. of the Assgiment Specs. Then provide your observation and screenshots after you increase the _count.

The result should work to display the negative value. You can watch the last 10 minutes of the lecture recording in week 2.
Viet Vo, 15 Aug at 6:29

Submission is acknowledged.
You now can include this task submission and the screenshots of the feedback in your final portfolio summary report.
Viet Vo, 21 Aug at 10:30

This assignment does not count toward the final grade.

2.3P - Drawing Basic Shapes with your own attributes

New Attempt

Due 16 Aug by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 1 Aug at 0:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips **and** Academic integrity

2.3P - Drawing Program - A Basic Shape with Your Own Attributes.pdf

Minimise file preview

Submission

Submitted!
4 Aug at 2:02
Submission details
Download 2.3P.pdf

Comments:
The submission is acknowledged.
You can include this task submission and the screenshots of the feedback into your final portfolio summary report.
Viet Vo, 15 Aug at 6:53

This assignment does not count toward the final grade.

2.4P - Identifiable Object with your ID

New Attempt

Due 16 Aug by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 1 Aug at 0:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips **and** Academic integrity

2.4P - Case Study Iteration 1 - Identifiable Object-1.pdf

Minimise file preview

Submission

Submitted!
12 Aug at 4:18
Submission details
Download 2.4P.pdf

Comments:
The submission is acknowledged.
You can include this task submission and the screenshots of the feedback into your final portfolio summary report.
Viet Vo, 15 Aug at 7:03

This assignment does not count toward the final grade.

3.1P - Clock Class with your own hour format

New Attempt

Due 23 Aug by 23:59 **Points** 0 **Submitting** a media recording or a file upload **File types** pdf and cs
Available after 8 Aug at 0:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips **and** Academic integrity

3.1P - Clock Class - your own format-2.pdf

Minimise file preview

Submission

Submitted!
15 Aug at 22:08
Submission details
Download 3.1P_new.pdf

Comments:
Submission for task 3.1P is acknowledged..
You can now include the current submission and the screenshots of the feedback into your final portfolio summary report.
Viet Vo, 21 Aug at 23:48

This assignment does not count toward the final grade.

3.2P - Stack and Heap and Null Pointer

New Attempt

Due 23 Aug by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 8 Aug at 0:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

3.2P - The Stack and Heap and Null Pointer.pdf

Minimise file preview

Submission

Submitted!
20 Aug at 17:31
Submission details
Download 3.2P - Answers Null Pointer-2.pdf

Comments:
Submission for task 3.2P is acknowledged.. You can now start building up your portfolio by adding this task submission and the screenshots of the feedback into your final portfolio summary report.
Viet Vo, 21 Aug at 23:57

This assignment does not count toward the final grade.

3.3P - Drawing Program - Drawing Class with your own attributes

New Attempt

Due 23 Aug by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 8 Aug at 0:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

3.3P - Drawing Program - A Drawing Class-2.pdf

Minimise file preview

Submission

Submitted!
20 Aug at 17:38
Submission details
Download 3.3P.pdf

Comments:
Submission for task 3.3P is acknowledged.. You can now start building up your portfolio by adding this task submission and the screenshots of the feedback into your final portfolio summary report.
Viet Vo, 22 Aug at 0:20

This assignment does not count toward the final grade.

4.1P - Drawing Program - Multiple shape kinds with your own attributes

New Attempt

Due 30 Aug by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 15 Aug at 2:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

4.1P - Drawing Program - Multiple Shape Kinds-1-1.pdf

Minimise file preview

Submission

Submitted!
26 Aug at 13:35
Submission details
Download 4.1P.pdf

Comments:
Submission for task 4.1P is acknowledged.. You can now start building up your portfolio by adding this task submission and the screenshots of the feedback into your final portfolio summary report.
Viet Vo, 29 Aug at 1:09

This assignment does not count toward the final grade.

4.2P - Case Study - Iteration 2: Custom Players, Items, and Inventory

New Attempt

Due 30 Aug by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 15 Aug at 2:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

4.2P - Case Study - Iteration 2 - Players, Items, and Inventory-1.pdf

Minimise file preview

Submission

Submitted!
28 Aug at 1:50
Submission details
Download 4.2P.pdf

Comments:
Submission for task 4.2P is acknowledged.. You can now start building up your portfolio by adding this task submission and the screenshots of the feedback into your final portfolio summary report.
Viet Vo, 29 Aug at 8:00

This assignment does not count toward the final grade.

5.1P - *In Person* Check-in 2 -Customized Drawing Program

New Attempt

Due 6 Sep by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 22 Aug at 23:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

This is the in-person check-in task. You must meet your tutor in person.

Please read the Deadline management tips and Academic integrity

5.1P - In Person Check-in 2 - Drawing Program.pdf

Minimise file preview

Submission

Submitted!
4 Sep at 10:02
Submission details
Download 5.1P - In Person Check-in 2 - Drawing Program Answer Sheet-1-1.pdf

Comments:
The submission for tasKs 5.1 is acknowledged.
You can include this submission and the screenshots in your final portfolio summary report.
Viet Vo, 4 Sep at 14:30

This assignment does not count toward the final grade.

5.2P - Case Study -- Interation 3: Bags

New Attempt

Due 6 Sep by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 22 Aug at 23:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

A demonstration of the completed test case design for this task can be viewed in this video.

Submission

Submitted!
4 Sep at 7:53
Submission details
Download 5.2P.pdf

Comments:
Submission for task 5.2P is acknowledged..
You can now start building up your portfolio by adding this task submission and the screenshots of the feedback into your final portfolio summary report.
Viet Vo, 4 Sep at 10:09

This assignment does not count toward the final grade.

5.3C - Drawing Program — Saving and Loading with Customized Payload

New Attempt

Due 6 Sep by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 22 Aug at 23:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

See the video demonstration of the completed task here:

Submission

Submitted!
4 Sep at 7:54
Submission details
Download 5.3C.pdf

Comments:
this task needs live demo
Viet Vo, 4 Sep at 10:26
The submission is acknowledged for task 5.3C
You can include this task submission and the feedback in the final portfolio summary report.
Viet Vo, 4 Sep at 15:22

This assignment does not count toward the final grade.

6.1P - Case Study - Iteration 4: Look Command with customization

New Attempt

Due 20 Sep by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 29 Aug at 23:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

6.1P - Case Study - Iteration 4 - Look Command.pdf

Minimise file preview

Submission

Submitted!
19 Sep at 23:36
Submission details
Download 6.1P-3.pdf

Comments:
MINOR REVISION

in IHaveInventory FetchContainer(Player p, string containerId), need to double check p.Locate(containerId) is null or not

Viet Vo, 18 Sep at 17:23
Submission for task 6.1P is acknowledged..
You can now start building up your portfolio by adding this task submission and the screenshots of the feedback into your final portfolio summary report.
Viet Vo, 29 Sep at 9:55

This assignment does not count toward the final grade.

6.2P - Key Object Oriented Concepts and Self-Reflection

New Attempt

Due 20 Sep by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 29 Aug at 23:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

6.2P - Key Object Oriented Concepts.pdf

Minimise file preview

Submission

Submitted!

18 Sep at 9:05

Submission details

Download Key Object Oriented Concepts Report.pdf

Comments:

Submission for task 6.2P is acknowledged.. You can now start building up your portfolio by adding this task submission and the screenshots of the feedback into your final portfolio summary report.

Viet Vo, 18 Sep at 22:27

This assignment does not count toward the final grade.

6.3D - Custom Program Initial Plan

New Attempt

Due 28 Sep by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf, docx, and cs
Available after 29 Aug at 23:59

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

6.3D - D Level Custom Program Initial Plan.pdf

Minimise file preview

Submission

Submitted!

8 Oct at 23:42 (late)

Submission details

Download D Level Custom Program Design-1.pdf

Comments:

This require in-person discussion to clarify your custom program.

At least 2-3 design patterns should be considered for D level to make the custom program more practical.

Viet Vo, 1 Oct at 16:40

This assignment does not count toward the final grade.

6.4D - D Level Custom Program : UML Design Level, Implementation, and User Documents

New Attempt

Due 25 Oct by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf, cs, zip, mp4, and mkv
Available after 29 Aug at 23:59

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

6.4D - D Level Custom Program.pdf

Minimise file preview

Submission

Submitted!

25 Oct at 22:26

Submission details

Download Custom_program_demonstration.zip

Download Custom_program_report_final.pdf

Comments:

No comments

This assignment does not count toward the final grade.

6.5HD - HD Level Custom Program Initial Plan

New Attempt

Due 28 Sep by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available after 29 Aug at 23:59

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

6.5HD - HD Level Custom Program Initial Plan.pdf

Minimise file preview

Submission

Submitted!

8 Oct at 23:43 (late)

Submission details

Download HD Level Custom Program Design-4.pdf

Comments:

Please discuss in person during my helpdesk or in the class.

We acquire at least 5 design patterns and exceptional functionalities.

Viet Vo, 1 Oct at 12:15

This assignment does not count toward the final grade.

6.6HD - HD Level Custom Program

New Attempt

Due 25 Oct by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf, cs, zip, mp4, and mkv
Available after 29 Aug at 23:59

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

6.6HD - HD Level Custom Program.pdf

Minimise file preview

Submission

Submitted!
25 Oct at 22:27
Submission details
Download Custom_program_demonstration-1.zip
Download Custom_program_report_final-1.pdf

Comments:
No comments

This assignment does not count toward the final grade.

7.1P - Case Study - Iteration 5 - Console Application typing prior iterations

New Attempt

Due 4 Oct by 23:59 **Points** 0 **Submitting** a file upload **File types** cs and pdf
Available after 12 Sep at 0:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

7.1P - Case Study - Iteration 5 - Tying it Together-1.pages.pdf

Minimise file preview

Submission

Submitted!
2 Oct at 9:10
Submission details
Download Program.pdf

Comments:
Submission for task 7.1P is acknowledged.

Viet Vo, 2 Oct at 16:35

This assignment does not count toward the final grade.

7.2C - Case Study — Advanced Iteration 6: Locations

New Attempt

Due 4 Oct by 23:59 **Points** 0 **Submitting** a file upload **File types** cs and pdf
Available after 12 Sep at 0:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

7.2C - Case Study - Iteration 6 - Locations.pages.pdf

Minimise file preview

Submission

Submitted!
2 Oct at 16:09
Submission details
Download 7.2C.pdf

Comments:
The submission is acknowledged for task 7.2C

Viet Vo, 2 Oct at 16:39

This assignment does not count toward the final grade.

T1 - Semester Test

Due 27 Sep by 10:30 **Points** 0 **Submitting** a file upload **File types** pdf and cs
Available 25 Sep at 9:00 - 28 Sep at 0:05

This assignment was locked 28 Sep at 0:05.

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

T1 - Semester Test 2024_final_2.pdf

Minimise file preview

Page 1 of 4 ZOOM

Submission

Submitted!
27 Sep at 2:38
Submission details
Download Mid semester test-2.pdf
Download File-2.cs
Download FileSystem-2.cs
Download Folder-2.cs
Download Program-2.cs
Download Thing-2.cs

Comments:
MINOR REVISION

fix and resubmit

In the UML, The arrow relation among classes are incorrect. Please distinguish the aggregation relation notation and composition relation notation

The concepts of the recursive nature where a folder contains files and folders are also missing..

Viet Vo, 1 Oct at 17:59

This assignment does not count toward the final grade.

T1-1 - Semester Test Fix and Resubmit

Due 17 Oct by 23:59 **Points** 0 **Submitting** a file upload **Available** 3 Oct at 23:30 - 18 Oct at 0:30

This assignment was locked 18 Oct at 0:30.

This assignment is only for students who received a MINOR or MAJOR REVISION outcome on T1 - Semester Test. Therefore, you are required to **FIX AND RESUBMIT** your work.

Submission

Submitted!

8 Oct at 10:33

Submission details

Download Mid semester test-3.pdf

Comments:

Submission acknowledged for T1-1

Viet Vo, 26 Oct at 1:17

This assignment does not count toward the final grade.

9.1P - *In Person* Check-in 3 - Case Study wrap up

New Attempt

Due 11 Oct by 23:59 **Points** 0 **Submitting** a file upload **File types** cs and pdf
Available after 27 Sep at 1:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

This is the in-person check-in task. You must meet your tutor in person.

Please read the Deadline management tips and Academic integrity

9.1P - In Person Check-in 3 - Case Study-1.pages.pdf

Minimise file preview

Submission

Submitted!

9 Oct at 15:58

Submission details

Download 9.1P - In Person Check-in 3 - Case Study Answer Sheet.pdf

Comments:

Submission is acknowledged for task 9.1P

Viet Vo, 9 Oct at 16:16

This assignment does not count toward the final grade.

9.2C - Case Study - Advanced Iteration 7: Paths

New Attempt

Due 11 Oct by 23:59 **Points** 0 **Submitting** a file upload **File types** cs and pdf
Available after 27 Sep at 1:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

9.2C - Case Study - Iteration 7 - Paths.pages.pdf

Minimise file preview

Submission

Submitted!

9 Oct at 16:53

Submission details

Download 9.2C.pdf

Comments:

The submissin is acknowledged 9.2C is acknowledged.

Viet Vo, 9 Oct at 17:46

This assignment does not count toward the final grade.

10.1C - Case Study - Advanced Iteration 8 - Command Processor

New Attempt

Due 18 Oct by 23:59 **Points** 0 **Submitting** a file upload **File types** cs and pdf
Available after 3 Oct at 10:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

10.1C - Case Study - Iteration 8 - Command Processor.pages.pdf

Minimise file preview

Submission

Submitted!

16 Oct at 13:41

Submission details

Download 10.1C-3.pdf

Comments:

Acknowledged 10.1C is acknowledged.

Viet Vo, 16 Oct at 14:17

This assignment does not count toward the final grade.

11.1P - Clock in Another Language with Practical Benchmarks

New Attempt

Due 25 Oct by 23:59 **Points** 0 **Submitting** a file upload **File types** pdf
Available after 11 Oct at 0:00

Note: ignore the text above saying that this assignment does not count towards the final grade. This is a misleading message automatically added by Canvas.

Please read the Deadline management tips and Academic integrity

11.1P - Clock in Another Language.pages.pdf

Minimise file preview

Submission

Submitted!

21 Oct at 9:50

Submission details

Download 11.1P.pdf

Comments:

This task is acknowledged for 11.P.

Viet Vo, 23 Oct at 13:32

This task is acknowledged for 11.P.

Viet Vo, 23 Oct at 13:34

- **Appendix II:** A summary of task corrections. For each task, if you have made any amendments or corrections that have not been previously assessed by your tutor, please summarize these changes. If there are no corrections or updates, indicate that the task submission from previous weeks (up to Week 12) remains unchanged.

Task 6.3D – Custom Program Initial Plan:

- Added design patterns list and explanation where the design patterns used in the UML

Class Diagram

Design patterns used:

- Observer Pattern (IOserver interface implemented by medical staff)
- Factory Pattern (MedicalStaffFactory creates various types of medical staff that will make the system more maintainable and also scalable)
- Strategy Pattern (Allow different treatment plans which allow flexibility in treatment strategies)

Task 6.5HD – HD Level Custom Program Initial Plan:

- Edit the UML diagram so that it have 5 design patterns

- **Appendix III:** A list of your up-to-date **corrected** task submissions in PDF format. These should reflect any changes mentioned in Appendix II. If your submission has already been assessed by your tutor on a weekly basis and there have been no changes, you do not need to resubmit the PDFs, as we already have them.

Task 6.3D – Custom Program Initial Plan:

Design Overview for Hospital Management System

Name: Nguyen Duc Thang
Student ID: 104776473

Summary of Program

Describe what you want the program to do… one or two paragraphs.

Include a sketch of sample output to illustrate your idea.

This program will illustrate the functionality of how Hospital manage the client and also the staff inside that hospital.

Sketch:

```
Please enter your details:
Name: Thang
Date of Birth: 30/11/2005
Contact: 0123456789
Symptoms: "Headaches", "Exhaustion", "Lack of sleep"

You will be assigned to a psychologist

After examination you are having: stress
You should take enough sleep, have a healthier diet and do some exercises

Your record has been stored in the hospital database
```

Required Roles

Describe each of the classes, interfaces, and any enumerations you will create. Use a different table to describe each role you will have, using the following table templates.

Table 1: <<role name>> details – duplicate

Responsibility	Type Details	Notes
_medical_staff:	MedicalStaff	Include information about doctors and nurse
_patient:	Patient	Include patient information and symtoms
_receptionist:	Receptionist	Include information and roles of a receptionist
_patient_record:	PatientRecord	Include: date and description given to the patient
_issue_date:	DateTime	Contains the day, month and year
_is_valid:	Bool	Check if the record is from a medical staff or not
_name:	String	Name of patient or medical staff
_contact:	String	Phone number of the person

_dob:	DateTime	Date of birth of the person
_symptoms:	string[]	Problem the patient are having
_working_hours:	string	Time that a medical staff will be on
_experience:	string	Number of years in the job
Hospital()	void	Contains patient and medical staff data
Manage()	void	Control how many medical staff are available and enable patient record and find it
PatientRecord()	void	Contains feedback given from the medical staff
Receptionist()	void	Can add and remove a patient record
Person()	void	Abstract class represent a human information
Patient()	void	Include patient's personal information and symtoms
Nurse()	void	A medical staff that will update the patient record
SurgicalNurse()	void	A type of nurse will update feedback from a surgeon
PediaricNurse()	void	A type of nurse will update feedback from a pediatric
EmergencyNurse()	void	A type of nurse will update feedback from others doctor type
MedicalStaff()	void	Abstract class inherit from person class
Surgeon()	void	Will give feedback to the patient
Pediatric()	void	Will give feedback to the patient
Cardiologist()	void	Will give feedback to the patient
Psychologist()	void	Will give feedback to the patient
MedicalStaffFactory	void	Create a new Medical Staff

Table 2: <<enumeration name>> details

Value	Notes

Class Diagram

Design patterns used:

- Observer Pattern (IOserver interface implemented by medical staff)
- Factory Pattern (MedicalStaffFactory creates various types of medical staff that will make the system more maintainable and also scalable)
- Strategy Pattern (Allow different treatment plans which allow flexibility in treatment strategies)

Patient Records
+ _issue_date: DateTime
+ _is_valid: bool
+ PatientRecords()
+ IssueDate(): string
+ IsValid():
+ Description(): string
Receptionist
+ Receptionist()
+ Name(): string
+ Contact(): string
+ AddPatientRecord()
+ RemovePatientRecord()
Surgeon
+ Surgeon()
+ Name(): string
+ Contact(): string
+ ProvideFeedback(): <<overriden>>
Pediatric
+ Pediatric()
+ Name(): string
+ Contact(): string
+ ProvideFeedback(): <<overriden>>
Use
Manage
+ _patient_record: PatientRecord
+ Mange()
+ AddPatientRecord() <<abstract>>
+ RemovePatientRecord() <<abstract>>
+ AddDoctor() <<property>>
+ RemoveDoctor() <<property>>
+ AddNurse() <<property>>
+ RemoveNurse() <<property>>
+ SearchPatient() <<property>>
+ EnableBooking() <<property>>
+ GetInstance(): Manage
<<abstract>>
Person
+ _name: string
+ _contact: string
+ _dob: DateTime
+ Person()
+ Name(): string
+ Contact(): string
+ DOB(): DateTime
Use
Medical Staff
+ _working_hours: string
+ _experience: string
+ Staff()
+ WorkingHours(): string
+ Experience(): string
+ Diagnose(): <<property>>
+ ProvideFeedback(): <<abstract>>
Doctor
+ Doctor()
+SetTreatment(TreatmentPlan plan)
Cardiologist
+ Cardiologist()
+ Name(): string
+ Contact(): string
+ ProvideFeedback(): <<overriden>>
<<Interface>>
IObserver
+ IObserver()
Patient
+ _symptoms: string[]
+ _observer: IObserver<>
+ Patient()
+ Name(): string
+ DOB(): string
+ Contact(): string
+ Symptoms(): string[]
+ Attach(observer: IObserver<>)
+ Detach(observer: IObserver<>)
+ Notify()
Use
<<Interface>>
Treatment Plan
+ TreatmentPlan()
+ ExecuteTreatment(Patient patient)
Psychologist
+ Neurologist()
+ Name(): string
+ Contact(): string
+ ProvideFeedback(): <<overriden>>
Hospital
+ _medical_staff: MedicalStaff
+ _patient: Patient
+ _receptionist: Receptionist
+ Hospital()
+ PatientList: List<Patient>
+ DoctorList: List<MedicalStaff>
+ NurseList: List<MedicalStaff>
+ ReceptionistList: List<Receptionist>
+ GetInstance(): Hospital
<<abstract>>
Nurse
+ Nurse()
+ UpdatePatientRecord() <<abstract>>
MedicalStaffFactory
+ field: Type
+ MedicalStaffFactory()
+ CreateStaff(): MedicalStaff
Mild Treatment
+ ExecuteMildTreatment()
Aggressive Treatment
+ ExecuteAggressiveTreatment()
Surgical Nurse
+ PediatricNurse()
+ Name(): string
+ Contact(): string
+ UpdatePatientRecord()
<<property, overriden>>
Pediatric Nurse
+ PediatricNurse()
+ Name(): string
+ Contact(): string
+ UpdatePatientRecord()
<<property, overriden>>
Emergency Nurse
+ EmergencyNurse()
+ Name(): string
+ Contact(): string
+ UpdatePatientRecord()
<<property, overriden>>

Sequence Diagram

Task 6.5HD – HD Level Custom Program Initial Plan:

Design Overview for Hospital Management System

Name: Nguyen Duc Thang
Student ID: 104776473

Summary of Program

Describe what you want the program to do… one or two paragraphs.

Include a sketch of sample output to illustrate your idea.

This program will illustrate the functionality of how Hospital manage the client and also the staff inside that hospital.

Sketch:

```
Please enter your details:
Name: Thang
Date of Birth: 30/11/2005
Contact: 0123456789
Symptoms: "Headaches", "Exhaustion", "Lack of sleep"

You will be assigned to a psychologist

After examination you are having: stress
You should take enough sleep, have a healthier diet and do some exercises

Your record has been stored in the hospital database
```

1. Design Patterns implemented in the program:
    - Singleton Pattern
    - Factory Pattern
    - Observer Pattern
    - Strategy Pattern
    - Decorator Pattern
2. Complexity and additional functionalities:
   In this custom program I have used the knowledge learnt from previous task as well as do some research about the design patterns and Responsibility Driven Design to implement the UML diagram which will have the flexibility to change it's core function due to the design patterns implemented.
3. In real world medical and health are essential fields for society, with this Health Management System I implemented, It will help enhance the efficiency of the Hospital system and also can be extended to different model of hospital due to it's versatile using the design patterns implemented into the program.

4. This is not simple compared to the D level Custom Program because it requires you to not just think about the solution for one problem but the custom program have to expand and involve for example it can be used in different library others than the dotnet framework or it can change it's functionality depends on the user's requirements.

Required Roles

Describe each of the classes, interfaces, and any enumerations you will create. Use a different table to describe each role you will have, using the following table templates.

Table 1: <<role name>> details – duplicate

Responsibility	Type Details	Notes
_medical_staff:	MedicalStaff	Include information about doctors and nurse
_patient:	Patient	Include patient information and symtoms
_receptionist:	Receptionist	Include information and roles of a receptionist
_patient_record:	PatientRecord	Include: date and description given to the patient
_issue_date:	DateTime	Contains the day, month and year
_is_valid:	Bool	Check if the record is from a medical staff or not
_name:	String	Name of patient or medical staff
_contact:	String	Phone number of the person
_dob:	DateTime	Date of birth of the person
_symptoms:	string[]	Problem the patient are having
_working_hours:	string	Time that a medical staff will be on
_experience:	string	Number of years in the job
Hospital()	void	Contains patient and medical staff data
Manage()	void	Control how many medical staff are available and enable patient record and find it
PatientRecord()	void	Contains feedback given from the medical staff
Receptionist()	void	Can add and remove a patient record
Person()	void	Abstract class represent a human information
Patient()	void	Include patient's personal information and symtoms
Nurse()	void	A medical staff that will update the patient record
SurgicalNurse()	void	A type of nurse will update feedback from a surgeon
PediaricNurse()	void	A type of nurse will update feedback from a pediatric
EmergencyNurse()	void	A type of nurse will update feedback from others doctor type
MedicalStaff()	void	Abstract class inherit from person class
Surgeon()	void	Will give feedback to the patient
Pediatric()	void	Will give feedback to the patient
Cardiologist()	void	Will give feedback to the patient
Psychologist()	void	Will give feedback to the patient
MedicalStaffFactory	void	Create a new Medical Staff
TreatmentPlan	void	Give the solution based on patient's condition

MildTreatment	void	Give a mild treatment to the medical staff
AggressiveTreatment	void	Give a aggressive treatment to the medical staff
MedicalStaffFactory	void	Create a medical staff
IObserver()	void	Update the data based on the input
Doctor()	void	Set treatment for patient
MedicalStaffDecorator()	void	Allow flexible addition to the responsibility to medical staff
SpecialDecorator()	void	Use for rare patient condition that need special consultation
EmergencyDecorator()	void	Use for patient with emergency issue

Table 2: <<enumeration name>> details

Value	Notes

Class Diagram

Sequence Diagram

Doctor
TreatmentPlan
MildTreatment
AggressiveTreatment
SetTreatment(MildTreatment)
ExecuteTreatment(Patient)
SetTreatment(AggressiveTreatment)
ExecuteTreatment(Patient)

- **Appendix IV:** The source code of all your previous task submissions in compressed zip format. Regardless of whether you made corrections or not, you must include all source code for your task submissions. Compress the source code into one or more zip files and submit them to Canvas along with your portfolio report. For example, the first .zip file can contain all the source code for the Hello-World program, The Counter, and Clock projects. The second .zip file should contain all the C# source code (.cs files) and test case implementations for your task submissions related to the Shape Drawing project. Finally, the third .zip file should include all the C# source code (.cs files) and test case implementations for your task submissions related to the Swin-Adventure case study.

- **Appendix V.** If you repeatedly receive minor/major revision feedback after the T1-1 - Semester Test Fix and Resubmit, you can still resubmit your corrections in this appendix. You need to summarize how you addressed the feedback and submit your full test solution again. However, this will result in a mark deduction from your final grade.
- Alternatively, if you failed to submit the T1-1 - Semester Test Fix and Resubmit by the deadline, you can still submit it in this appendix. However, this will result in a **FAIL grade** for the unit if your test resubmission is incorrect.

Remarks:

Failure to provide the source code for any task submission will result in that task not being assessed, even if the source code is printed and included in the PDF submission. The teaching team needs the source code to execute it and verify correctness. Additionally, the source code will be used for plagiarism detection and academic integrity checks.

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. Failure to meet this requirement will result in a failing grade for the unit.
Failure to provide the source code for any task submission will result in that task not being assessed, even if the task is included in PDF format.

Signature: **Nguyen Duc Thang**

Portfolio Overview

This portfolio includes work that demonstrates that I have achieve all Unit Learning Outcomes for COS20007 Unit Title to a **High Distinction** level.
I have completed all the task and received acknowledgement for the P and C marks, I also submitted the initial plans for my D and HD custom program and my custom programs contains 5 design patterns, also my file saving system for the patient records are able to save and update the saved data without deleting the whole file information and replace with the new data, this feature extend beyond from the C task of just saving and deleting the data and repeat the process which is inefficient.

Task Summary

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

- 1.1P: Completed
- 1.2P: Completed
- 2.1P: Completed
- 2.2P: Completed
- 2.3P: Completed
- 2.4P: Completed
- 3.1P: Completed
- 3.2P: Completed
- 3.3P: Completed
- 4.1P: Completed
- 4.2P: Completed
- 5.1P: Completed
- 5.2P: Completed
- 5.3C: Completed
- 6.1P: Completed
- 6.2P: Completed
- 6.3D: Submitted and received comments and resubmitted

  Comments:

  This require in-person discussion to clarify your custom program.

  At least 2-3 design patterns should be considered for D level to make the custom program more practical.

  Viet Vo, 1 Oct at 16:40

- 6.4D: Submitted
- 6.5HD: Submitted and received comments and resubmitted

Comments:

Please discuss in person during my helpdesk or in the class.

We acquire at least 5 design patterns and exceptional functionalities.

Viet Vo, 1 Oct at 12:15

- 6.6HD: Submitted
- 7.1P: Completed
- 7.2C: Completed
- T1 –Semester Test: Submitted and received minor revision
- T1 –Semester Test Fix and Resubmit: Completed
- 9.1P: Completed
- 9.2C: Completed
- 10.1C: Completed
- 11.1P: Completed

Reflection

The most important things I learnt:

In this unit I think I've learnt a lot of valuable knowledge about OOP such as the 4 basic concept in OOP (Polymorphism, inheritance, abstraction and encapsulation), how to develop a good coding project using that concepts and also extend them by using design patterns to make the solution more practical and scalable.

The things that helped me most were:

I think the online lecture every week and the materials in Canvas module helps me a lot in the first few weeks when starting this unit because this is the first time I've never done my submission in Adobe Acrobat and also setting up and use Splashkit.

I found the following topics particularly challenging:

When starting to learn about the OOP concepts I have some struggles in understanding the 4 basics concept in OOP (inheritance, polymorphism, abstraction and encapsulation) and interpret those concepts so that everyone can understand but with the help of my tutor Viet Vo, I started to do some deeper research about the concepts in different coding languages focused more in OOP such as Java and C++, now I think I have a better understanding about OOP and how it applies in making an optimal solution to solve real life problems in businesses.

I found the following topics particularly interesting:

To me in this unit I think the design patterns is the topic that makes me interest the most because the design patterns help the coding solution more flexible and scalable in solving customer different requirements. For instance you can easily change your coding project depends on the customer criteria without much edit to your coding project.

I feel I learnt these topics, concepts, and/or tools really well:

I think I learnt well about UML sequence diagram, UML Class Diagrams and Design Patterns because I was able to do iteration 7 and 8 where you need to implement your own UML diagram and sequence diagram to make the Path and Command processor for the game. Additionally I also create my own UML diagram and sequence diagram with 5 design patterns for my custom program.

I still need to work on the following areas:

I think I still need some improvement in General Responsibility Assignment Software Patterns (GRASP) in OO design because my UML class connections in the Semester Test wasn't optimal enough so I have to fix the connection arrows between the classes.

My progress in this unit was …:

During the semester, I tried to submit my weekly task at the start of that week or before my class with the tutor so that I maintain the work flow throughout the semester and also ensure that if there any problems I need to fix then I still have time to adjust my errors. I also discuss and ask about the knowledge that I don't understand in that week content and next week content with my tutor in class so that I can ensure about my task for that week and plan for next week tasks. I think scheduling your work helps me a lot in the unit and future units because it gives me more controls of my work and I can adjust if others unit need more time to study.

This unit will help me in the future:

This unit will be the foundation for studying future units and also future works because it helps me understand how to create a coding projects using OOP and how are the criteria to make a good coding solution that will help solve business problems.

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

If there's a chance to do this unit again, I might plan my work schedule better, start things as early as possible and seek as many helps as possible. During my study from week 6 and from week 9-12 I wasn't able to focus fully focus on designing the custom program because I wasn't predicted correctly the workload of other unit such as Network and Switching and Cloud Computing architecture so I have to split the work for others unit assignments.

Other…:

[Add any other reflections you think help you demonstrate your learning]