**ASSIGNMENT 1 FRONT SHEET**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | |
| **Unit number and title** | Unit 20: Advanced Programming | | |
| **Submission date** |  | **Date Received 1st submission** |  |
| **Re-submission Date** |  | **Date Received 2nd submission** |  |
| **Student Name** |  | **Student ID** |  |
| **Class** | GCH0705 | **Assessor name** | Doan Trung Tung |
| **Student declaration**  I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice. | | | |
|  |  | **Student’s signature** |  |

**Grading grid**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| P1 | P2 | M1 | M2 | D1 | D2 |
|  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **❒ Summative Feedback: ❒ Resubmission Feedback:** | | |
| **Grade:** | **Assessor Signature:** | **Date:** |
| **Lecturer Signature:** | | |

1. Introduction

In this document, I will introduce the concept of object-oriented programming using the C# programming language. This will cover design patterns, their definitions and intended applications, as well as scenarios illustrating their usage. Additionally, class diagrams and user diagrams will be provided to enhance understanding.

2. OOP general concepts

2.1. What is OOP

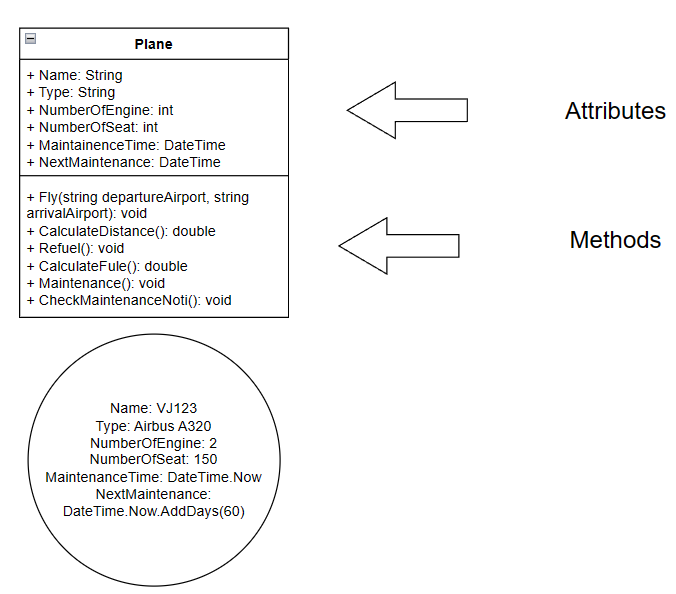
Object-Oriented Programming (OOP) is a programming paradigm based on the concept of “objects”, which can contain data and code. The data is in the form of fields (often known as attributes or properties), and the code is in the form of procedures (often known as methods). In OOP, computer programs are designed by making them out of objects that interact with one another. Any contemporary Object-Oriented Programming language is built around classes. A class must be created in OOP languages in order to represent data. A class is the blueprint for an object and includes functions to manipulate the data as well as variables to store it. A class is merely a logical representation of data because it will not take up any memory. Simply use the term "class" followed by the class name to create a class (Nyakundi, 2023)

2.2. Structure

2.3 Class

In object-oriented programming (OOP), a class is a pattern or blueprint to define the specific objects that we will create in the program. The class acts as a "template" for creating objects, defining how they work, and managing their data and behavior. Among the important ideas about classes are:

* A class can have subclasses that can inherit all or some of the characteristics of the class. In relation to each subclass, the class becomes the superclass.
* Subclasses can also define their own methods and variables that are not part of their superclass.
* The structure of a class and its subclasses is called the class hierarchy.



This class diagram represents an plane consisting of the following components (Attributes):

* Name: is the name of plane with data type string
* Type: is plane’s type (passenger plane, cargo plane, combat aircrafts,…) with data type is string
* NumberOfEngine: the number of plane engines used with data type is int
* NumberOfSeat: the number of plane seat with data type is int
* MaintainceTime: Represents the aircraft's most recent maintenance date wih data type is DateTime

Plane’s operation (Methods):

Fly(string departureAirport, string arrivalAirport): take off and land at airports

Refuel(): appropriate refueling for each flight

Maintenance(): Maintenance and set maintenance date

3.OOP scenario

ABC company is a wood company have a plan to create an application to make a table by customer self. The product is mixed by many other source which have difference attribute and price