# Comp 4603

# Advanced C++

|  |  |  |  |
| --- | --- | --- | --- |
| Assignment | 8 | Part | 1 |

Student Name: Alisher Shamayev

BCIT ID: A01182685

To design Five C++ Behavioral design patterns with their names and problems to solve

Task 1: Your example of Iterator with UML diagram

|  |
| --- |
| A piece of paper with writing on it  Description automatically generated  In this case we made as class of school example so we have implemented Iterator design pattern which illustrates as StudentIterator interface, and VectorStudentIterator implementation of interface and last but not least Student concrete class. So in future we can make StudentList class and implement VectorStudentIterator method iterating through a list. Without changing in future of any concrete class or getting anything from concrete class. |

Task 2: Your example of Visitor with UML diagram

|  |
| --- |
| A piece of paper with writing on it  Description automatically generated  Here we have abstract class Visitor and two sub classes Client and Driver that will implement Visitor class method visit. Also we have Taxi concrete class that will call visit method in its own method. |

Task 3: Your example of Observer with UML diagram

|  |
| --- |
| A piece of paper with writing on it  Description automatically generated  Observer is abstract class and has own sub class Query. Other concrete base class is Database it has own several methods however it will call update method in its class through Query class and so we will see when and where the updates occurs through this design pattern. |

Task 4: Your example of Template Method with UML diagram

|  |
| --- |
| A piece of paper with writing on it  Description automatically generated  This scenario we can understand as Movie’s in cinema and lets say different movie types have different ticket prices. So we made a base class Movie and two child classes ComedyMovie and HorrorMovie so base class has to method and price will be usual regular price, however if Comedy and Horror have different price it will be overridden and implemented in child class. |

Task 5: Your example of State with UML diagram

|  |
| --- |
| Text, letter  Description automatically generated  Pretty simple case we made a delivery system for this pattern so basically state class have two child so one is Transit and another is Delivery and basically whenever package is delivered or in transit state can be updated, settled or changed through this design. |