Question 1: In this assignment, you need to describe your own example of any design pattern of your choice and give answers to all of the parts of the question 1. (Part A to Part D – see below) But your scenario / example must be a non-programming example explaining any real-life scenario, like the one given below.

Observer Example

The Observer defines a one to many relationships, so that when one object changes state, the others are notified and updated automatically. Some auctions demonstrate this pattern. Each bidder possesses a numbered paddle that is used to indicate a bid. The auctioneer starts the bidding, and "observes" when a paddle is raised to accept the bid. The acceptance of the bid changes the bid price, which is broadcast to all of the bidders in the form of a new bid.

Part (A) Give your own example (textual + graphic representation):

Describe your own example of any designs pattern that we have covered in the class and give a substitute case of the design pattern being utilized in some other situation. (as shown in the above example) It very well may be a regular day to day example (e.g., how online goods are shipped, how to leave a place of employment and join another organization, ... whatever). In a manner similar to above example, briefly describe your example with a diagram and a brief textual description.

Important Note: The model you pick ought not be utilized in the GoF book (also minor varieties of the given examples are not allowed) or one that was utilized in the lecture slides or notes, neither it should be taken from any web article.

Part (B) The problem, The Intent, The motivation and The Applicability

Clarify how your case "coordinates" the example's essential thought, that is, the way your model is closely resembling the structure example's model. Typically, you should clarify how your model synchronizes with (1) the intent and (2) the problem being discuss. From the GoF patterns, it should coordinate the aim or intent, inspiration and applicability.

Part (C) UML

Draw a <u>class</u> diagram and <u>interaction</u> diagrams for the model you picked. What you sketch will rely upon the pattern you decided for your model. Utilize standard notations of UML.

Part (D) Consequences

Document a sensible situation that features one of the consequences (disadvantages) of the design pattern. That is, your case study must be feature an outcome that is a downside in the situation. Iyou're your situation should highlight how the solution failed to address your requirements. Else you should portray a situation wherein neglecting to utilize the pattern would bring about negative outcomes. For example, in strategy pattern there is a communication overhead between Strategy and its Context, you'll need tighter coupling between Strategy and Context.

Part(E) Write or Generate skeleton Code of class diagram in part (C).

Question 2 – Pattern identification:

Consider each of the scenarios below and identify the design pattern which is most directly addresses the problem described. Briefly explain your reasoning. (See what maps from GOF patterns and discuss and argue in favor of your thinking)

Part (A): You've developed a new implementation of the List interface and you'd like to test the behavior of your new data structure. You've written an algorithm to perform your speed tests, but the algorithm needs to make a great many instances of your list class. You want to test the performance of your list against the performance of ArrayList and LinkedList, but you don't want to have to write your algorithm three times in order for it to be able to create the right kind of list to test.

Part (B) You've completed a compiler for a new language! There are many parts to your compilation process: parsing, transformation, assembly code generation, and so forth. You'd like to allow other programmers to use an interface to compile their code without resorting to system calls or other command-line invocations – your compiler can just run in their processes – but you don't want those users to have to know how to bring all of the steps of compilation together in order to use your compiler.

Note:

1. Assignment must be submitted on Google forms (MS word Assignment report).
2. Last date of submission is ==**16-02-2024 11:00 AM (sharp)**==
3. Assignments will not be accepted after due date.
4. Only 1 submission is allowed, if you once submit it then you are allowed to update your submission.
5. You need to submit in your section only. If you submit in wrong section or both sections or if you violate any assignment instructions then expect negative marking.
6. Email submission not allowed. Emails queries asking hints for solution will result in negative marking.
7. If you upload empty or corrupted archive, you will get zero marks. Hence double check before uploading.
8. Plagiarism, if detected, will result in zero marks.
9. Do not use the sample shown in the reading assignment / text books or form internet.
10. Upload a Zip archive (name is as your_name_roll_number_section.ZIP) having following contents:
11. MS word document in /Doc directory
12. Contents of the MS Word report:
    a. Cover Page of Assignment must contain: Student name, Roll no, Date of submission.
    b. Attach screenshot of the question paper after cover page.
    c. Solutions to all the questions with proper headings, figures, source code, and textual descriptions.
13. Source code in /code directory
14. All UML diagrams in /UML directory