Marking guide:

|  |  |
| --- | --- |
| **There are 25 \* N marks in total in this assignment where N = 2. Marks are awarded based on correctness.** | **Marks** |
|  |  |
| 1. Identifying **N** design patterns used in the frameworks in this assignment (5 \* N marks) | 0 |
| * 1. The name of the design pattern identified (N marks) | 0 |
| * 1. The locations of code involved in the framework (i.e., file name & line numbers) (2 \* N marks) | 0 |
| * 1. The class diagram of the design pattern identified. All the components in the design pattern class diagram provided in our textbook should be explicitly labelled in your class diagram. (2\*N marks) | 0 |
|  |  |
| 1. Applying **N** design patterns in your assignment 2 solution (20 \* N marks) | **34/40** |
|  |  |
| * + 1. The class diagram before your modification (2 \* N marks) | **3** |
| * + - * [Design Pattern 1 Before](part%202%20Builder%20before.vsdx) | 1.5 |
| * + - * [Design Pattern 2 Before](part%202%20Template%20Method%20Before.vsdx) | 1.5 |
|  |  |
| * + 1. The locations of code involved (i.e., code reference) in your assignment 2 solution (2 \* N marks) | **4** |
| * + - * Pattern 1 | 2 |
| * + - * 1. Client Code: \source\interpreter\_controller.py, method do\_show() |  |
| * + - * 1. Main Logic: GraphView = \source\view\graph\_view.py |  |
| * + - * 1. Graph interface: \source\view\i\_graph\_view.py |  |
| * + - * Pattern 2 | 2 |
| * + - * 1. Client Code: \source\model\interpreter.py, methods serialize\_data\_arr(), save\_file, load\_file() |  |
| * + - * 1. Main Logic: FileHandler = \source\model\file\_handler\file\_handler.py |  |
| * + - * 1. FileHandler Interface: \source\model\file\_handler\i\_file\_handler.py |  |
|  |  |
| * + 1. The name of the design pattern applied (2 \* N marks) | **4** |
| * + - * Design Pattern 1 = Builder | 2 |
| * + - * Design Pattern 1 = Template Method | 2 |
|  |  |
| * + 1. The reasons why applying this design pattern is suitable; the reasons MUST be specified for the particular situation you try to apply, i.e., do not just give general reasons why using that design pattern is good. (2 \* N marks) | **4** |
| * + - * Design Pattern 1: | 2 |
| * + - * 1. The builder pattern is used to separate complex objects from the creation process. In this case I separated the various graphs as being the complex objects from the creation process of creating a graph. This is good because it gives the client a common interface for generating graphs. |  |
| * + - * Design Pattern 2: | 2 |
| * + - * 1. I used to the template method to consolidate the similarities primarily in serialize\_data\_arr(), save\_file() methods. Having the common code held in the template method and calling to the primitive methods to handle the aspects that are different. This is good as it helps reduce duplicate code now and if future development. E.g. adding another file type or implementing load data for shelve files. |  |
|  |  |
| * + 1. The class diagram after your modification; all the components in the design pattern class diagram provided in our textbook should be explicitly labelled in your class diagram. (2 \* N marks) | **3** |
| * + - * [Design Pattern 1 After](part%202%20Builder%20after.vsdx) | 1.5 |
| * + - * [Design Pattern 2 After](part%202%20Template%20Method%20after.vsdx) | 1.5 |
|  |  |
| * + 1. Applying the design pattern proposed (10 \* N marks) | **16** |
| * + - * Pattern 1 Refer Code | 8 |
| * + - * Pattern 2 Refer Code | 8 |