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** |
|  | 1.5 |
|  | 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: \interpreter\_controller.py, method do\_display\_graph() |  |
| * + - * 1. Main Logic: GraphView = \graph\graph.py |  |
| * + - * 1. Graph interface: \graph\igraph.py |  |
| * + - * Pattern 2 | 2 |
| * + - * 1. Client Code: \interpreter.py, methods serialize\_data\_arr(), save\_file, load\_file() |  |
| * + - * 1. Main Logic: Database = \database\database\database.py, sql\_database.py |  |
| * + - * 1. Database Interface: \Database\idatabase.py, database.py |  1. Client Code \interpreter.py methods 2. MainLogic: FileManagement\FileHandler.py 3. Graph Interface FileManagement\IFileHandler.py |  |
|  |  |
| * + 1. The name of the design pattern applied (2 \* N marks) | **4** |
| * + - * Design Pattern 1 = Builder | 2 |
| * + - * Design Pattern 2 = Factory | 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. Separate condensed graph file into smaller individual that generate one type of graph, reducing the complexity of the file as well as the length. Client can now generate graph easier using 1 function rather using multiple one |  |
| * + - * Design Pattern 2: | 2 |
| * + - * 1. Factory method for file handler so can differentiate the similar between the file save or load method. It helps easier to modify the base line how the file should be save rather than having everything in 1 function and can potentially break the code.         2. For database, it was choosen so their can have a template that will import or open any database and does not require to be the same as the previously given one |  |
|  |  |
| * + 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** |
|  | 1.5 |
|  | 1.5 |
|  |  |
| * + 1. Applying the design pattern proposed (10 \* N marks) | **16** |
| * + - * Pattern 1 Refer Code | 8 |
| * + - * Pattern 2 Refer Code | 8 |