circus of Plates  
Game

# patterns in our design:

## Singleton

|  |  |
| --- | --- |
|  | Clown, GameInfo:  class have its constructor as private and have a static instance of itself |

## Pool

|  |  |
| --- | --- |
|  |  |

## Iterator

|  |  |
| --- | --- |
|  | Iterator interface which narrates navigation method and a LevelStrategy interface which retruns the iterator. Concrete classes (Easy, Medium, Hard)implementing the LevelStrategy interface will be responsible to implement Iterator interface and use it |

## Dynamic Linkage

|  |  |
| --- | --- |
|  |  |

## Snapshot

|  |  |
| --- | --- |
|  |  |

## State

|  |  |
| --- | --- |
|  | State interface defining an action and concrete state classes (StateMiss, Statenopoint, Statepoint) implementing the State interface. Shape is a class which carries a State. |

## Strategy

|  |  |
| --- | --- |
|  | LevelStrategy interface defining an action and concrete strategy classes (Easy, Medium, Hard) implementing the LevelStrategy interface. GameInfo is a class which uses a Strategy. |

## Flyweight

|  |  |
| --- | --- |
|  | GameObject interface and concrete class Shape implementing the GameObject interface. A factory class ShapeFactory is defined as a next step. |

## Observer

|  |  |
| --- | --- |
|  | abstract class Observer and a concrete class (ObserverView, ObserverPoints, ObserverTime, ObserverLevel) that is extending class Observer. |

## Visitor

|  |  |
| --- | --- |
|  | EmptyStacks interface defining accept opearation. NoEmptyStacks, TwoEmptyStacks, LeftStackEmpty and RightStackEmpty are concrete classes implementing EmptyStacks interface. We will define another interface EmptyStacksVisitor which will define a visitor class CatchingObjects. |

BLABLABLA:

* blablabla
* blabla
* blablablabla

# Team members

We approve the project as described above.

|  |  |
| --- | --- |
| Name | ID |
| Yomna Gamal ELDin Mahmoud | 60 |
| Nada Salama |  |
| Eman Rafik | 11 |
| Toka Alaa |  |