Q1 – Implementation of undo/redo pattern

Answer – In my design the undo/redo pattern is implemented with the help of an Array called history which is of type Command . Command is a deferred class which exists in Command subcluster under model cluster and it has 3 deferred feature which must be implemented by its children. The 3 deferred feature are execute, undo and redo which are implemented by pass, play, move and fire classes (children of Command class).

A pointer called cursor_pointer is used to move back and forth on history Array to execute undo and redo functionality. In case of undo the current Command is reverted and cursor_pointer is moved back by 1. In case of redo command, the curser_pointer is moved forward by 1 then the current command is executed again.

To print the message of undo and redo command I am using a Linked list called “write_last” which is storing the messages of different command being executed by the user. At the time of undo command, I decrement the cursor of write_last list then print that message. In case of redo command, I am incrementing the cursor of write_last list then printing the stored message to the user.

Hence with the help of history array I was able to implement the undo/redo functionality and with the help of write_last linked list I was able to print the corresponding message for undo and redo.

Q2

Occurrence of polymorphism –

The polymorphism occurs in undo/redo pattern implementation when a user inputs a command like fire then we create an instance of fire which is a child of Command class using its make constructor. In my history array of commands, the command variable is replaced by the child of deferred class Command at the runtime and based on that a particular command is undo or redo. Occurrence of dynamic binding –

Dynamic binding occurs in history array to execute undo/redo feature. Based on the cursor position I decide which command, s execute, undo, redo feature has to be called at the run time. When ever an undo command is called then the cursor position is decremented and current command as per the cursor pointer is reverted.

So, the decision of which command to execute or undo/redo it is decided by the cursor position at the run time.