Full Name: _____

Lab 7: Implementation v.2

Note: This work must be completed by the end of the lab in the week of June 6.

Goals:

1. Be able to select a suitable ADT in Java.

Tasks:

- Implement the following for your elevator program, using appropriate data structures:
  - the elevator scheduling algorithm (to keep things simple, keep only one list of target floors, rather than distinguish between up and down requests)
  - multiple, concurrent elevator calls
- Remove cyclic dependencies from your code. For example, if a Floor can turn on/off the light of a CallButton, and a CallButton can talk to a Floor to call an elevator, this is a cyclic dependency. Either use a message queue to facilitate communication or pass a CallFloorInterface, implemented by the Floor, to the CallButton.

```
public interface CallFloorInterface {
    void callElevator(Direction.DIRECTION dir);
}

public class Floor implements CallFloorInterface {
    CallButton callButton;
...
    public Floor(CallElevatorSystemInterface system, int id) {
        callButton = new CallButton(this);
...
}

public class CallButton {
...
    public CallButton(CallFloorInterface floor) {
...
}
```

Submission:

- Please submit:
  - ☐ a printout of all your changed non-test source code (2-up is ok). Please highlight the changes made. See the dropbox for details re: correctness, documentation, and structure. Tests do not need to be documented.
  - ☐ an electronic copy of all your source code; use zip to compress the entire source directory
  - ☐ Labs 1, 2, and 6.
- ☐ Also demonstrate your test results using the supplied test code; see EventSimulatorInterfaceTest2.java

_____
Instructor Stamp