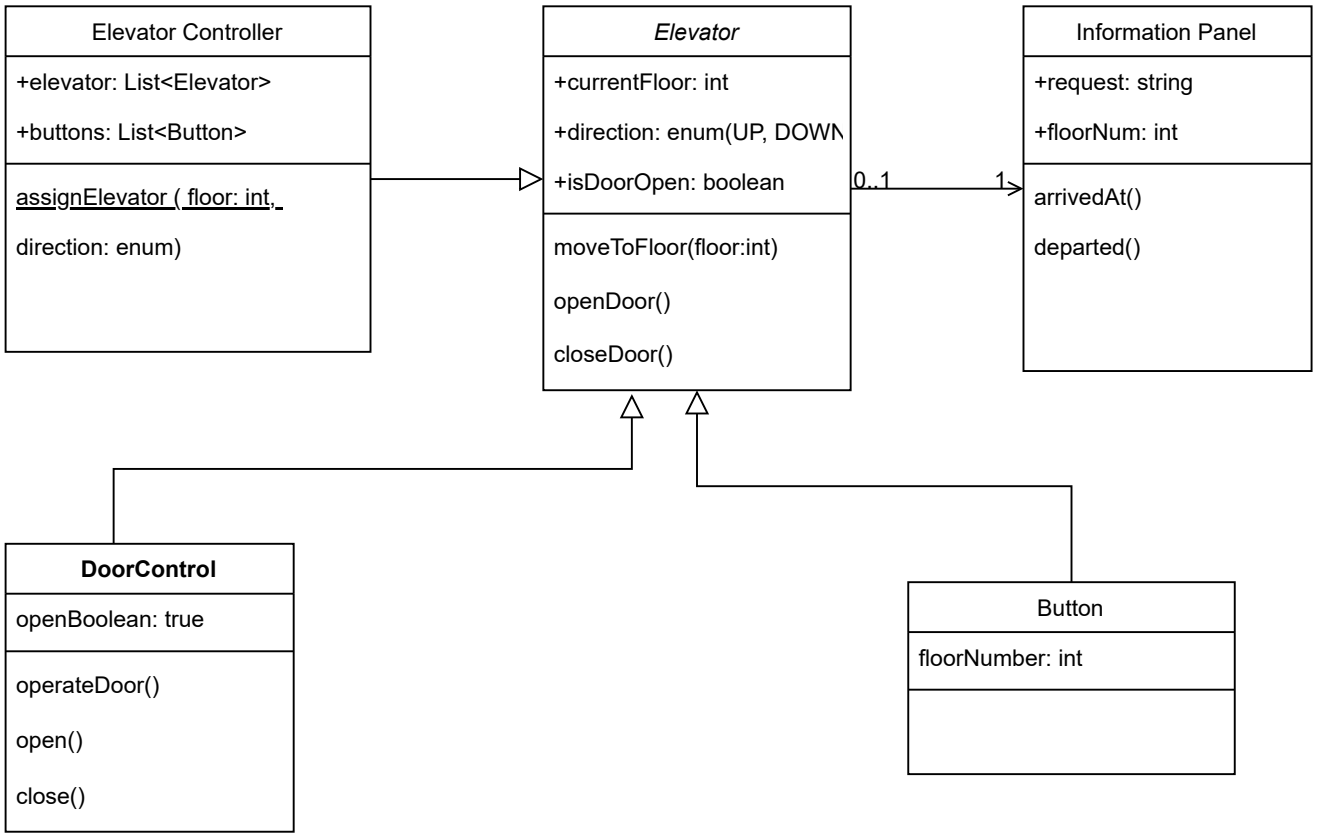
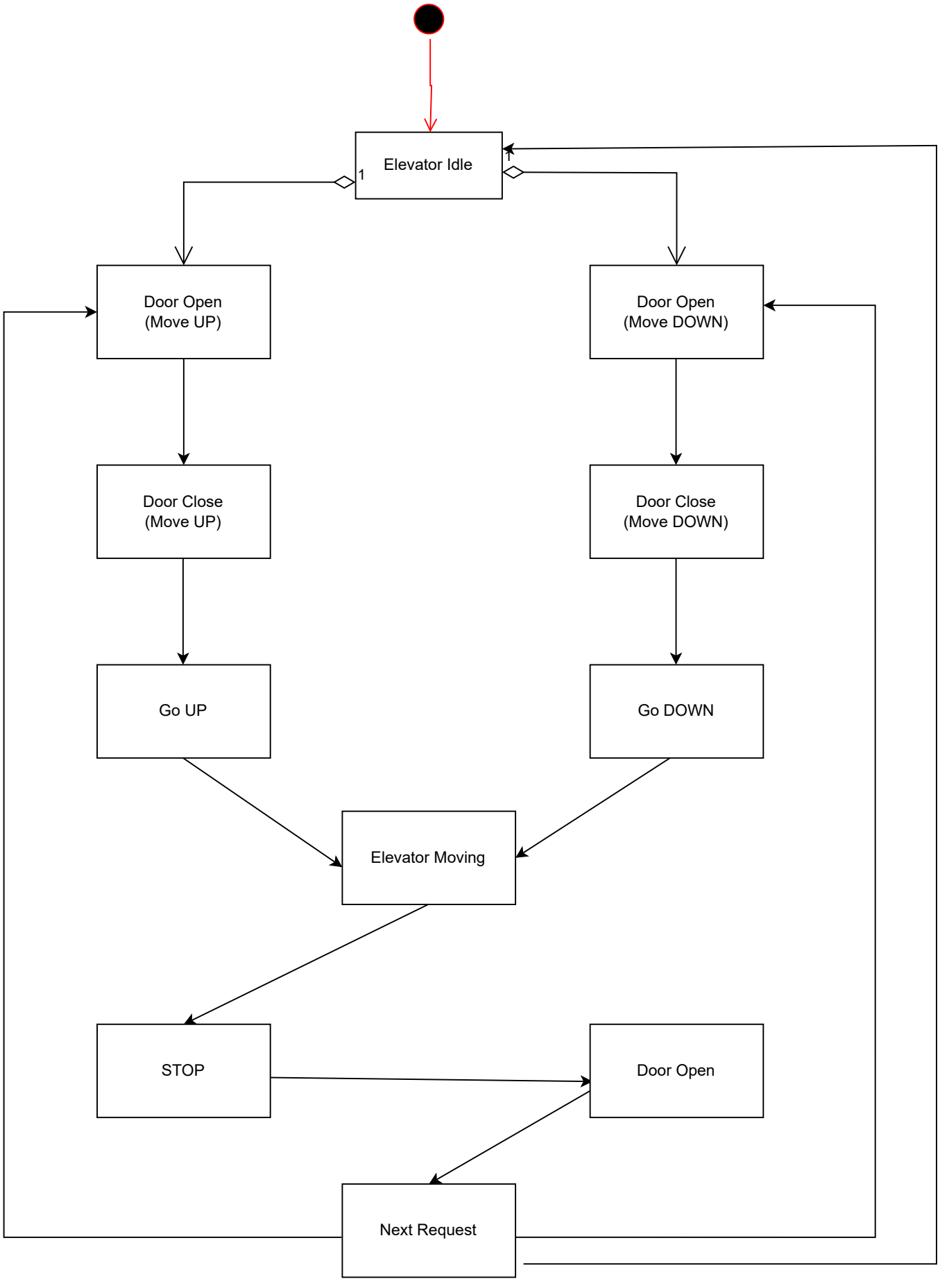
Design an "Elevator"
Name: Huy Tran SID: 017069104 Class: CS 151


Elevator Controller
+elevator: List<Elevator>
+buttons: List<Button>
assignElevator (floor: int,
direction: enum)
Elevator
+currentFloor: int
+direction: enum(UP, DOWN
+isDoorOpen: boolean
moveToFloor(floor:int)
openDoor()
closeDoor()
0..1
1
Information Panel
+request: string
+floorNum: int
arrivedAt()
departed()
DoorControl
openBoolean: true
operateDoor()
open()
close()
Button
floorNumber: int


Elevator Idle
1
Door Open
(Move UP)
Door Open
(Move DOWN)
Door Close
(Move UP)
Door Close
(Move DOWN)
Go UP
Go DOWN
Elevator Moving
STOP
Door Open
Next Request

```
public class Elevator {
public int currentFloor;
public boolean isdoorOpen;
public String request;
public int floorNumber;
}

public Elevator(int currentFloor, boolean isdoorOpen, String request, int floorNumber)
{
this.currentFloor = currentFloor;
this.isdoorOpen = isdoorOpen;
this.request = request;
this.floorNumber = floorNumber;
}

public int getcurrenFloor() {
return currentFloor;
}

public void setCurrentFloor(int currentFloor) {
this.currentFloor = currentFloor;
}
public String getRequest() {
return request;
}

public void setRequest(String request) {
this.request = request;
}

public boolean getIsdoorOpen() {
return isdoorOpen;
}

public void setIsdoorOpen(boolean isdoorOpen) {
this.isdoorOpen = isdoorOpen;
}

public int getFloorNumber() {
return floorNumber;
}

public void setFloorNumber(int floorNumber) {
this.floorNumber = floorNumber;
}
```

```
public class ElevatorTest {
private static char[] isdoorOpen;

public static void main(String[] args) {
Elevator elev = new Elevator(0, false, null, 0);

elev.setCurrentFloor();
elev.setIsdoorOpen(false);
elev.setRequest(null);
elev.setFloorNumber(0);

System.out.println("CurrentFloor is" + elev.getcurrenFloor());
System.out.println(isdoorOpen);
System.out.println("Request Floor" + elev.getRequest());
}
}
```