

Homework 08

In this assignment, you will implement a simulation of an elevator that serves a building that has 5 floors. Each floor has a random number of passengers [0-5].

When the elevator starts, it will start from floor 1 and goes up. The elevator stops and opens its door to get passengers in only if there is at least one passenger on the floor or if it's checking out a passenger to the requested floor. Otherwise the elevator keeps moving. The elevator goes up to the fifth floor and then goes down to the first floor. By the end of the trip, the elevator should serve all the people who requested the service. Follows a breakdown of the simulation program.

Description

1. Initialize building: First, you need to initialize the five floors. Each floor has a random number of passengers. Each passenger is requesting to go to another floor (other than the floor she is currently in).
 - (A) The number of passengers on each floor is a random number [0-5] (see 1 in Figure1)
 - (B) Passenger: Every passenger object has the following properties:
 - (a) Passenger ID: This is an integer that starts from 1 and increment by one when every new object is instantiated.
 - (b) Floor: This is the current floor the passenger is currently in (see 3 in Figure1)
 - (c) Destination Floor: This is the floor the passenger is going to. (see 4 in Figure1). Note that a destination floor is a random number but at the same time this number can not be the same number as the current floor.
2. Print Building Status: In this function, you need to print out the building status as shown in Figure1.
3. Start Elevator: In this function, the elevator will start moving from the first floor. While the elevator is moving you need to print out every stop of the elevator and how many passengers left/ get in the elevator.
 - (A) Starting from floor 1. The elevator picks all passengers on the first floor. You need to add the passenger objects to the elevator and also print out the current transaction (see 1 and 2 in Figure2)
 - (B) You need to print the elevator movement direction. Up or down. (see 3 in Figure2)
 - (C) If there is no passenger in the elevator is going to a certain floor AND there is also no passengers waiting on that floor then the elevator should keep moving. For example, no passenger is going to floor 2 and no passenger is waiting so elevator should keep moving the third floor (see 4 in Figure2)
 - (D) After reaching the fifth floor the elevator goes down. As you can see the elevator does not pick new passengers. Only serving current passengers in the elevator (see Figure3)
 - (E) Note that when the elevator reaches back the first floor the number of passengers in the elevator should **ALWAYS** be equal to zero. (see Figure3)

Building has 5 floors

```
=====
Floor 1 (1) passengers waiting:
Passenger{passengerID=1, floor=1, destinationFloor=3}
=====
Floor 2 (0) passengers waiting:
=====
Floor 3 (1) passengers waiting:
Passenger{passengerID=2, floor=3, destinationFloor=5}
=====
Floor 4 (2) passengers waiting:
Passenger{passengerID=3, floor=4, destinationFloor=2}
Passenger{passengerID=4, floor=4, destinationFloor=1}
=====
Floor 5 (2) passengers waiting:
Passenger{passengerID=5, floor=5, destinationFloor=4}
Passenger{passengerID=6, floor=5, destinationFloor=4}
```

Figure 1: Printing The Building Status

```
Elevator started
At floor 1, elevator picked 1 passengers
Going up
=====
Stopped at floor 3, elevator picked 1 passengers, # of passengers now 2
Elevator served 1 passengers, # of passengers now 1
Going up
=====
Stopped at floor 4, elevator picked 2 passengers, # of passengers now 3
Elevator served 0 passengers, # of passengers now 3
Going up
=====
Stopped at floor 5, elevator picked 2 passengers, # of passengers now 5
Elevator served 1 passengers, # of passengers now 4
Going down
```

Figure 2: Elevator Moving - part1

Stopped at floor 4. Elevator served 2 passengers, # of passengers now 2
Going down

=====

=====

Stopped at floor 2. Elevator served 1 passengers, # of passengers now 1
Going down

=====

Stopped at floor 1. Elevator served 1 passengers, # of passengers now 0

Figure 3: Elevator Moving - part2

Hints:

1. Use an object of type Random to generate random numbers
2. Elevator is a list. You can add passengers and remove specific passengers regardless of when they were added to the list but based on the values of the passengers' data.
3. Each floor is a list of passengers. The size of the list is dynamic (random number) and the values of the destination floor are random as well.
4. Divide your code into three functions:
 1. Initialize Building
 2. Print Building Status
 3. Start Elevator