Q2

May 17, 2023

[44]:
```python
class Elevator:
    def __init__(self,max_capacity,num_floors):
        self.state = 0                    # Ground Floor
        self.capacity = 0
        self.max_capacity = max_capacity
        self.moving = True
        self.num_floors = num_floors

    def move_to_floor(self,floor_number):
        if(self.state == floor_number):
            return
        elif(self.state > floor_number and self.moving == True):
            while self.state != floor_number:
                self.state -= 1
        elif(self.moving == True):
            while self.state != floor_number:
                self.state += 1

    def check_capacity(self):
        if(self.capacity > self.max_capacity):
            return False
        else:
            return True

    def add_passenger(self):
        if(self.check_capacity() == True):
            self.capacity += 1
        else:
            print("Capacity Full")

    def remove_passenger(self):
        self.capacity -= 1

    def emergency_stop(self):
        self.moving = False

    def back_to_normal(self):
```

```
        self.moving = True

    def oper(self):
        print("Press E for emergency stop:")
        print("Press F for elevator being functional again:")
        print("Press any floor Number in range 0 to",self.num_floors)
        print("Press N for new pessenger:")
        print("Press R to remove pessenger:")
        print("Press C to show current Floor")
        inp = input("Press:")
        try:
            inp = int(inp)
            if(inp > self.num_floors or inp < 0):
                print("Wrong Floor Entered")
                return False
            else:
                self.move_to_floor(inp)
                return True
        except:
            if(inp == 'E' or inp == 'e'):
                self.emergency_stop()
                return True
            elif(inp == 'F' or inp == 'f'):
                self.back_to_normal()
                return True
            elif(inp == 'N' or inp == 'n'):
                self.add_passenger()
                return True
            elif(inp == 'R' or inp == 'r'):
                self.remove_passenger()
                return True
            elif(inp == 'C' or inp == 'c'):
                print(self.state)
                return True
            else:
                return False
```

[45]:
```
e = Elevator(10,10)
```

[46]:
```
while e.oper():
    continue
```

```
Press E for emergency stop:
Press F for elevator being functional again:
Press any floor Number in range 0 to 10
Press N for new pessenger:
Press R to remove pessenger:
```

```
Press C to show current Floor

Press: 10

Press E for emergency stop:
Press F for elevator being functional again:
Press any floor Number in range 0 to 10
Press N for new pessenger:
Press R to remove pessenger:
Press C to show current Floor

Press: c

10
Press E for emergency stop:
Press F for elevator being functional again:
Press any floor Number in range 0 to 10
Press N for new pessenger:
Press R to remove pessenger:
Press C to show current Floor

Press: n

Press E for emergency stop:
Press F for elevator being functional again:
Press any floor Number in range 0 to 10
Press N for new pessenger:
Press R to remove pessenger:
Press C to show current Floor

Press: r

Press E for emergency stop:
Press F for elevator being functional again:
Press any floor Number in range 0 to 10
Press N for new pessenger:
Press R to remove pessenger:
Press C to show current Floor

Press: a
```

[47]:
```
e.capacity
```

[47]:
```
0
```

[ ]:
```

```