Gremlin Lifts

Gremlin Lifts, a division of CBG Inc[1], has installed an elevator in a seven-story building. Due to their European heritage, Gremlin Lifts has numbered the floors 0 – 6 and refuse to re-number them, claiming that the request is "out of scope". Gremlin Lifts, having been built by actual gremlins, don't necessarily work the way one might hope.

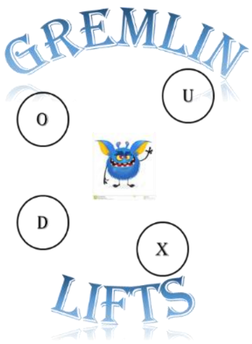


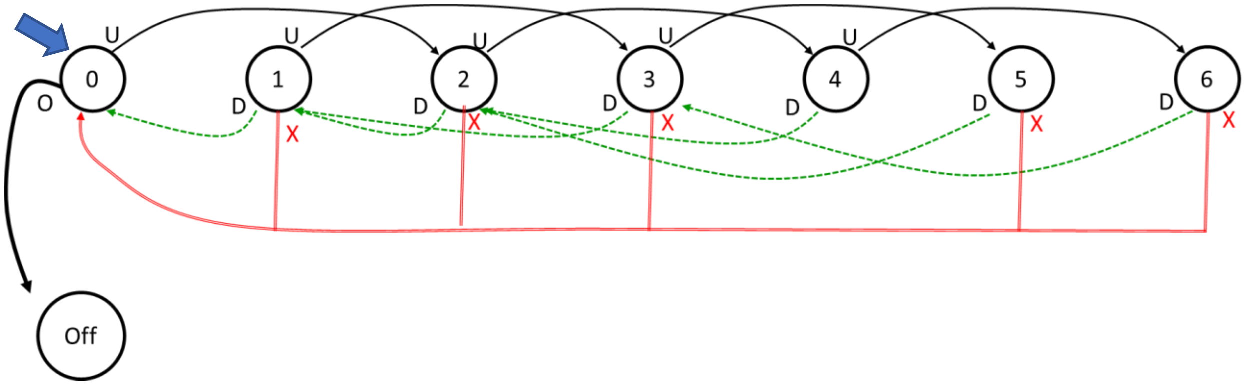
Figure 1 Control Panel

[2]

The Gremlin Lift elevator has the same four 4 buttons in the elevator and on each floor: Up, Down, eXpress and Off. Using a proprietary technology, the elevator only responds to a button if it is already at a floor. Buttons pressed while the elevator is in motion are ignored.

Their transition from floor to floor is governed by the following state machine … please note that Off is really the ground floor with the elevator turned off.

Figure 2 Elevator State Machine



Buttons

- Pushing any button in the Off state has no effect—there is no on button. At installation the elevator was in the "0" state.
- The up button always causes the elevator to go up 2 floors, if it cannot go up 2 floors it does nothing.
- Pushing the down button causes the elevator to go down the half the number of floors it is on (rounded up).
- The express button causes the elevator to go to the ground (0) floor, except a system glitch means the button does not work when on floor "4".
- The off button only works if on the ground (0) floor.

[1] Cheap But Glitchy Incorporated

[2] https://www.dreamstime.com/scary-cartoon-monster-gremlin-big-mouth-waving-hand-halloween-vector-illustration-image109880061

It may be easier to visualize this a table. Blanks indicate that the elevator does not transition state (i.e. change floors) if that button is pushed.

	Up	Down	Xpress	Off
⇨ 0	2			Off
1	3	0	0	
2	4	1	0	
3	5	1	0	
4	6	2		
5		2	0	
6		3	0	
off				

You are provided starter code in Gremlins_lift_stater.py. You are to implement the Gremlins Lift state machine in Python. Some guidance on implementation and some additional requirements are as follows:

- The variable `floor` represents the state.
- Use the value `-1` to represent the “off” state. Represent the other states by integers.
- You need to change the initial value of `floor` to the appropriate value.
- You need to write the loop condition so the state machine responds to buttons when not off.
- The rest of you code goes inside the provided while loop, below the comment.
- In general, each Loop iteration should:
  - Determine what button is pushed;
  - Based on current state, make a transition to a new state when appropriate.
- Your program should not crash on any input, including a blank line.
  - A blank line has a len() == 0 or is an empty string ""
- Your program should accept capital and lowercase inputs for button pushes (“U” or “u” for ‘up’). The inputs for button pushes will match the letters in the control panel.
- Your final submission may not have any extra print statements, only the two provided in the starter code. You may, of course, use print statements to help you debug, just delete them before submitting.
- You will also need to
  - Type your answers to the 2 questions at the bottom of the starter file.
  - Fix the header doc-string (your name, section, short description).

Submission

Submit a file, using your name, as LastnameFirstname _Section_HW2.py”. This file must run from the command line, using Python 3 to receive any credit.