2022-2023 Fall CMPE480 Homework 3
Deadline: 26.12.2022

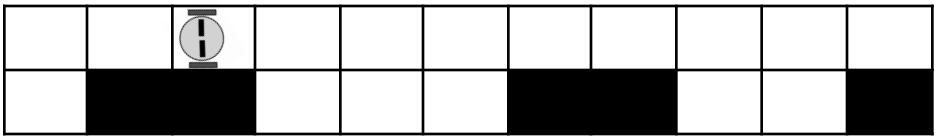
- Submit 1 file (not zipped!): last-name.py (not ipynb!)
- last-name.py file contents will be copied to Google Colab notebook.
- Make sure your program runs with version Python 3.7 in Google Colab environment
    - (run !python --version).

- You need to use and update the following code segment!

```
walls="x xx  xx x"
sensors=["on","on","off","on"]
# your code starts

# your code ends
print('The most likely current position of the robot is',robot_pos,'with probability',robot_pos_prob)
```

Assume a robot in a grid world with two rows and arbitrary number of columns.



- Initially, the robot locates in an arbitrary cell in the upper row.
- Each grid in the lower row might include an obstacle (shown with black box).
- The upper row does not contain any obstacles.
- The map of the environment is known a-priori.
- The initial location of the robot is not known.
- But it is known that the probability of being in any grid in the upper row is equally likely initially.
- The locations of the obstacles in the lower row are given in your python code in the first row.

For example, line:
`walls="x xx  xx x"`
corresponds to the following environment:

- At each time-step, the robot
    - Senses its environment.
    - Attempts to move one grid to the right
- However, there is uncertainty both in sensing and actuation.
- The sensor corresponds to a range sensor that measures wall information on the bottom. It is supposed to report on if there is a wall and off if there is no wall. However, sometimes it makes errors. Sometimes Sensor=on even though there is no wall and sometimes Sensor=off even though there is a wall below. The corresponding emission probability is known and given below:

P(Sensor|Wall)	**on**	**off**
+wall	0.7	0.3
-wall	0.2	0.8

- There is also uncertainty in actions. At each time-step, the robot might move right or stay in the same location with the following probabilities:
    - If the current location index is an even number, the probability to move right is 0.8.
    - If the current location index is an odd number, the probability to move right is 0.6
- If the robot is at the end of the corridor, it of course cannot move to the right anymore.
- Location indices start from 1 from the left-most grid.

- The sensor information is also provided:

```
sensors=["on","on","off","on"]
```

1. The sensor measured ***on*** at t=1.
2. The robot attempted the right action.
3. The sensor measured ***on*** at t=2.
4. The robot attempted the right action.
5. The sensor measured ***off*** at t=3.
6. The robot attempted the right action.
7. The sensor measured ***on*** at t=4.
8. The robot attempted the right action.

- Now you need to calculate the most likely location of the robot after its last right action and report both the grid index and the probability of being in that grid. You need to assign the corresponding variables for this:

```
print('The most likely current position of the robot is',robot_pos,'with probability',robot_pos_prob)
```