PROBLEM 1: ROBOT CLASS (10/10 points)

For the `Robot` class, decide what fields you will use and decide how the following operations are to be performed:

- Initializing the object
- Accessing the robot's position
- Accessing the robot's direction
- Setting the robot's position
- Setting the robot's direction

Complete the `Robot` class by implementing its methods in `ps2.py`.

Note: When a `Robot` is initialized, it should clean the first tile it is initialized on. Generally the model these Robots will follow is that after a robot lands on a given tile, we will mark the entire tile as clean. This might not make sense if you're thinking about really large tiles, but as we make the size of the tiles smaller and smaller, this does actually become a pretty good approximation.

Although this problem has many parts, it should not take long once you have chosen how you wish to represent your data. For reasonable representations, *a majority of the methods will require only a couple of lines of code*.)

Note:
The `Robot` class is an *abstract* class, which means that we will never make an instance of it. Read up on the Python docs on abstract classes at this link and if you want more examples on abstract classes, follow this link. If you took edX course 6.00.1x already, you've seen an abstract class - the `Trigger` class from the final problem set!

In the final implementation of `Robot`, not all methods will be implemented. Not to worry -- its subclass(es) will implement the method `updatePositionAndClean()` (this is similar to the `evaluate` method of the `Trigger` class from 6.00.1x).

Enter your code for classes `RectangularRoom` (from the previous problem) and `Robot` below.

```
        """
        self.tiles = [[1 for x in xrange(height)] for x in xrange(width)]
        #raise NotImplementedError

    def cleanTileAtPosition(self, pos):
        """
        Mark the tile under the position POS as cleaned.

        Assumes that POS represents a valid position inside this room.

        pos: a Position
        """
        self.tiles[int(pos.getX())][int(pos.getY())] = 0
        #raise NotImplementedError

    def isTileCleaned(self, m, n):
```

Correct

Test results

Hide output

CORRECT

```
Test: 1 class creation
```

Although Robot is an abstract class, we create instances of it for
the purposes of testing your code's correctness.
robot = Robot(RectangularRoom(1,2), 1.0)

Output:

```
```

```
Test: 2 test getRobotPosition
```

robot = Robot(RectangularRoom(5,8), 1.0)
robot.getRobotPosition()

Output:

```
Test passed
```

```
Test: 3 test getRobotDirection
```

robot = Robot(RectangularRoom(5,8), 1.0)
robot.getRobotDirection()

Output:

```
Test passed
```

```
Test: 4 test setRobotPosition
```

```
robot = Robot(RectangularRoom(5,8), 1.0)
robot.getRobotPosition()
loop 10 times:
  * Generate random x, y values
  * Check if Position(x,y) is in the room
    * If so, robot.setRobotPosition(Position(x, y))
    * robot.getRobotPosition()
```

Output:

```
Random position 0: (6.00, 9.00)
Random position 1: (4.00, 2.00)
(4.00, 2.00)
Random position 2: (6.00, 8.00)
Random position 3: (1.00, 3.00)
(1.00, 3.00)
Random position 4: (6.00, 7.00)
Random position 5: (4.00, 1.00)
(4.00, 1.00)
Random position 6: (6.00, 9.00)
Random position 7: (5.00, 1.00)
Random position 8: (0.00, 6.00)
(0.00, 6.00)
Random position 9: (2.00, 8.00)
```

Test: 5 test setRobotDirection

```
robot = Robot(RectangularRoom(5,8), 1.0)
robot.getRobotDirection()
loop 10 times:
   * Generate random direction value
   * robot.setRobotDirection(randDirection)
   * robot.getRobotDirection()
```

Output:

```
Test passed
Random direction: 297
297
Random direction: 130
130
Random direction: 80
80
Random direction: 104
104
Random direction: 338
338
Random direction: 256
256
Random direction: 311
311
Random direction: 177
177
Random direction: 138
138
Random direction: 201
201
```

Test: 6 test updatePositionAndClean

The abstract class Robot should not implement updatePositionAndClean.

Output:

```
NotImplementedError successfully raised.
```

Hide output

Check Save *You have used 1 of 30 submissions*

Show Discussion

New Post

EdX offers interactive online classes and MOOCs from the world's best universities. Online courses from MITx, HarvardX, BerkeleyX, UTx and many other universities. Topics include biology, business, chemistry, computer science, economics, finance, electronics, engineering, food and nutrition, history, humanities, law, literature, math, medicine, music, philosophy, physics, science, statistics and more. EdX is a non-profit online initiative created by founding partners Harvard and MIT.