Problem 4: Running the Simulation

In this problem you will write code that runs a complete robot simulation.

Recall that in each trial, the objective is to determine how many time-steps are on average needed before a specified fraction of the room has been cleaned. **Implement the following function:**

def runSimulation(num\_robots, speed, width, height, min\_coverage, num\_trials,

robot\_type):

"""

Runs NUM\_TRIALS trials of the simulation and returns the mean number of

time-steps needed to clean the fraction MIN\_COVERAGE of the room.

The simulation is run with NUM\_ROBOTS robots of type ROBOT\_TYPE, each with

speed SPEED, in a room of dimensions WIDTH x HEIGHT.

"""

The first six parameters should be self-explanatory. For the time being, you should pass in StandardRobot for the robot\_type parameter, like so:

avg = runSimulation(10, 1.0, 15, 20, 0.8, 30, **StandardRobot**)

Then, in runSimulation you should use robot\_type(...) instead of StandardRobot(...) whenever you wish to instantiate a robot. (This will allow us to easily adapt the simulation to run with different robot implementations, which you'll encounter in Problem 6.)

Feel free to write whatever helper functions you wish.

We have provided the getNewPosition method of Position, which you may find helpful:

class Position(object):

def getNewPosition(self, angle, speed):

"""

Computes and returns the new Position after a single clock-tick has

passed, with this object as the current position, and with the

specified angle and speed.

Does NOT test whether the returned position fits inside the room.

angle: integer representing angle in degrees, 0 <= angle < 360

speed: positive float representing speed

Returns: a Position object representing the new position.

"""

For your reference, here are some approximate room cleaning times. These times are with a robot speed of 1.0.

* One robot takes around 150 clock ticks to completely clean a 5x5 room.
* One robot takes around 190 clock ticks to clean 75% of a 10x10 room.
* One robot takes around 310 clock ticks to clean 90% of a 10x10 room.
* One robot takes around 3322 clock ticks to completely clean a 20x20 room.
* Three robots take around 1105 clock ticks to completely clean a 20x20 room.

(These are only intended as guidelines. Depending on the exact details of your implementation, you may get times slightly different from ours.)

You should also check your simulation's output for speeds other than 1.0. One way to do this is to take the above test cases, change the speeds, and make sure the results are sensible.

For further testing, see the next page in this problem set about the optional way to use visualization methods. Visualization will help you see what's going on in the simulation and may assist you in debugging your code.

Enter your code for the definition of runSimulation below.