**Northeastern University**

CS 5100 Foundations of Artificial Intelligence

**Homework#9** [300 points]

***PART A: Theory and Algorithms*** *[100 points] \* See PART B Prog Assignment on Page 3*

Please - clearly write your **full name** on the first page. Submit a single PDF file.

Please provide brief but complete explanations, using diagrams where necessary, and suitably using your own words. While presenting calculations, explain the variables and.

Study Chapter 24 and 25 of Russel AI textbook – selected sections only, plus notes provided. Answer the below:

1. **From textbook p.969** [40 points]
2. Problem 24.4
3. Problem 24.7
4. **Based on Ch. 25 from textbook pg. 1015** [60 points]
5. Problem 25.3
6. Problem 25.7a
7. Problem 25.9a

***PART B: Implementation – Programming Assignment (PA)*** *(200 points)*

In this assignment, we’ll focus on Ch. 24 and 25.

**Section 1**: **Computer Vision - OpenCV** [60 points]

Install OpenCV. Using a suitable image (not the same in the tutorial),implement and demonstrate below operations: <https://www.pyimagesearch.com/2018/07/19/opencv-tutorial-a-guide-to-learn-opencv/>

1. Rotating an image
2. Smoothing an image
3. Counting objects
4. Converting an image to grayscale
5. Edge detection
6. Detecting and Drawing Contours

Briefly explain the results, what the operation is and how it is done in the code.

**Section 2**: **Build a Balancing Bot (Robotics** using **PyBullet)** *see the references provided.*

**Assignment:** *build a Balancing Bot* [140 points]

For this assignment you need to build a Balancing Bot. Similar to this:

<https://backyardrobotics.eu/2017/11/27/build-a-balancing-bot-with-openai-gym-pt-i-setting-up/>

Your bot should be able to balance on its own after a few minutes of training. Take a screenshot of the Bot and attach it along with your source code in the submission file. You can use the above code as a starting point. But, perturb the object (knock it down or move it) at least once or twice randomly using your own lines of code. Build a version of this with a slight, creative modification as your own.

Gym is a toolkit for developing and comparing reinforcement learning algorithms. It makes no assumptions about the structure of your agent, and is compatible with any numerical computation library, such as TensorFlow or Theano. The [gym](https://github.com/openai/gym) library is a collection of test problems: environments that you can use to work out your reinforcement learning algorithms. These environments have a shared interface, allowing you to write general algorithms. The gym comes prepackaged with many environments. It is this common API around many environments that makes Gym so great. Here we will list additional environments that do not come prepacked with the gym. Submit another to this list via a pull-request.

**PyBullet Robotics Environments**

3D physics environments like the Mujoco environments but uses the Bullet physics engine and does not require a commercial license. Works on Mac/Linux/Windows.

This tutorial will guide you in using PyBullet:

<https://alexanderfabisch.github.io/pybullet.html>

**References:** <https://docs.google.com/document/d/10sXEhzFRSnvFcl3XxNGhnD4N2SedqwdAvK3dsihxVUA/edit#heading=h.wz5to0x8kqmr>

**Pybullet QuickStart Guide:**

<https://usermanual.wiki/Document/pybullet20quickstart20guide.479068914/html>

<https://medium.com/@gerardmaggiolino/creating-openai-gym-environments-with-pybullet-part-1-13895a622b24>