**CIS 4930 Artificial Intelligence**

**Assignment 1**

Due date: September (Sunday at 11:59 pm)

**Problem 1 (6 points)**

Answer the following questions:

1. Give your own definition of Artificial Intelligence

Answer:

AI is the combination of math, science, and algorithms implemented in code by computing machines to solve non-trivial problems important to humans and society. The word “artificial” implies that the intelligence is not organic but crated by human intelligence. The Turing test challenges computer scientists to develop intelligent code and computing machines that cannot be distinguished from real humans. AI solves important engineering problems and also learns and adapts to the environment. In short, AI is the science of making machines that pass the Turing test.

1. Is AI a science, or is it engineering? Or neither or both? Explain.

Answer:

For me, science is more the study of a problem or theory while engineering is the application of that study to perform a task or design a machine to solve a problem. Engineering is applied science. AI could be either a science or engineering depending on the student. For example, a computer scientist may study the various aspects of an area of AI and create new knowledge in that area where an engineer may read about that area but implement a design to solve a real world problem. To be more specific, imagine a scientist working on a natural language processing approach that improves the efficiency of translating one language to another. Imagine the engineer in the same field understanding the current art but also exploiting that to develop a real product used by the military or consumers.

1. Imagine that the speed and memory of computers became 1000 larger. Which AI problems become trivial due to this increase, and which would not get any easier?

Answer: Most asymptotic problems that are of exponential order quickly become unmanageable due to the growth curve. Even at 1000x improvement, these problems would not get much better. I can see text to speech and speech to text improving significantly with this type of increase. Facial recognition and image processing would show noticeable improvements but the bigger problems that suffer from non-linear processing times would continue to be an issue. For example, the traveling salesman problem would not go away with faster computers.

**Problem 2 (4 points)**

Answer the following questions:

1. Define in your own words the following terms: state, state space, search tree, search node, goal, action, transition model, and branching factor

Answer:

**State**: The current status of an object or agent

**State Space:** The combination of starting state, actions, and transition that form the graph. How the world is at a point in time.

**Search Tree:** The representation of all the possible choices that radiate from one decision to another. The search tree is where the algorithm calculates the possibilities.

**Search Node:** A specific point in the search tree. The node has parent, children, and state/

**Goal:** The final state or objective trying to be met through the analysis.

**Action:** The moves or move possible from the node in the search or decision tree.

**Transition model:** The set of possible moves to change from one state to another.

Branching Factor: A number that represents how many children a parent node may have. I some cases this is 2 which leads to a binary tree. Other times the branching may be 3, or more children. If there are many children an average may be taken.

1. What is the difference between a world state, a state description, and a search node? Why is this distinction useful

Answer:

**World State:** The total description of the real world. This is the actual world that is not simplified by a model. It includes every thing in the real world. This is the infinitely large world.

**State Description:** The information that describes a state at a point in time.

Search Node: A node is a bookkeeping data structure used to represent the search tree. A State corresponds to a configuration of the world. (Pg. 79, Russell,Norvig)

**Problem 3 (10 points)**

Implement a program that inputs a weighted directed graph, and finds the shortest path between two given vertices of this graph. Your program must read a graph from a given file, prompt the user to specify two vertices, and output the shortest path between them. The format of the graph encoding is as follows:

<vertex> <vertex> <weight>

<vertex> <vertex> <weight>

<vertex> <vertex> <weight>

…

Each line encodes an edge, which points from the first to the second vertex, and the weight of this edge. The vertices of the graph are denoted by natural numbers, which may not be consecutive; for example, the vertex numbers may be 0, 2, and 8. All edge weights are positive, and they are also encoded by natural numbers. For example, the following graph consists of three vertices (denoted 0, 2, and 8) and four edges:

0 2 25

15

25

10

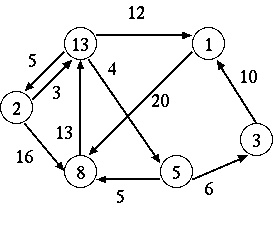
20

2 8 15

8 0 10

8 2 20

Problem 3 on Assignment 1 involves writing a program to solve the shortest path problem on a graph. In order to illustrate this problem further, sample file and graph are provided here.



Solutions:

This graph contains six vertices and ten edges. An example of a shortest path through this graph is that from vertex 2 to vertex 8.

Starting vertex: 2   
Ending vertex: 8   
The shortest path is:

Vertex 2 to vertex 13 (edge weight of 3)   
Vertex 13 to vertex 5 (edge weight of 4)   
Vertex 5 to vertex 8 (edge weight of 5)

Note that the shortest path in this case was not the path with the fewest edges, but the least total edge weight. For instance, there is an edge that directly connects vertices 2 and 8, but this edge has a weight of 16, which means it is *not* the shortest path. There is also another path (from 2-13-1-8), but this has an even greater total weight of 35.

Total weight: 12

**Submitting your assignment**

* Submission via Canvas Assignment.
  + It is your responsibility to submit these assignments in a timely fashion.
* All files should be zipped together.
* There should be a readme file explaining in detail the exact steps to be taken to compile and execute the code files and the title page
* In case of any code errors, partial credit may be offered based on the code and documentation.

**Late Submission Policy**

* Late work will be not accepted.

**Grading Criteria:**

* Minus 10 points if code does not compile.
* Minus 7 points if it compiles but does not run.
* Further deduction will be depended on your code