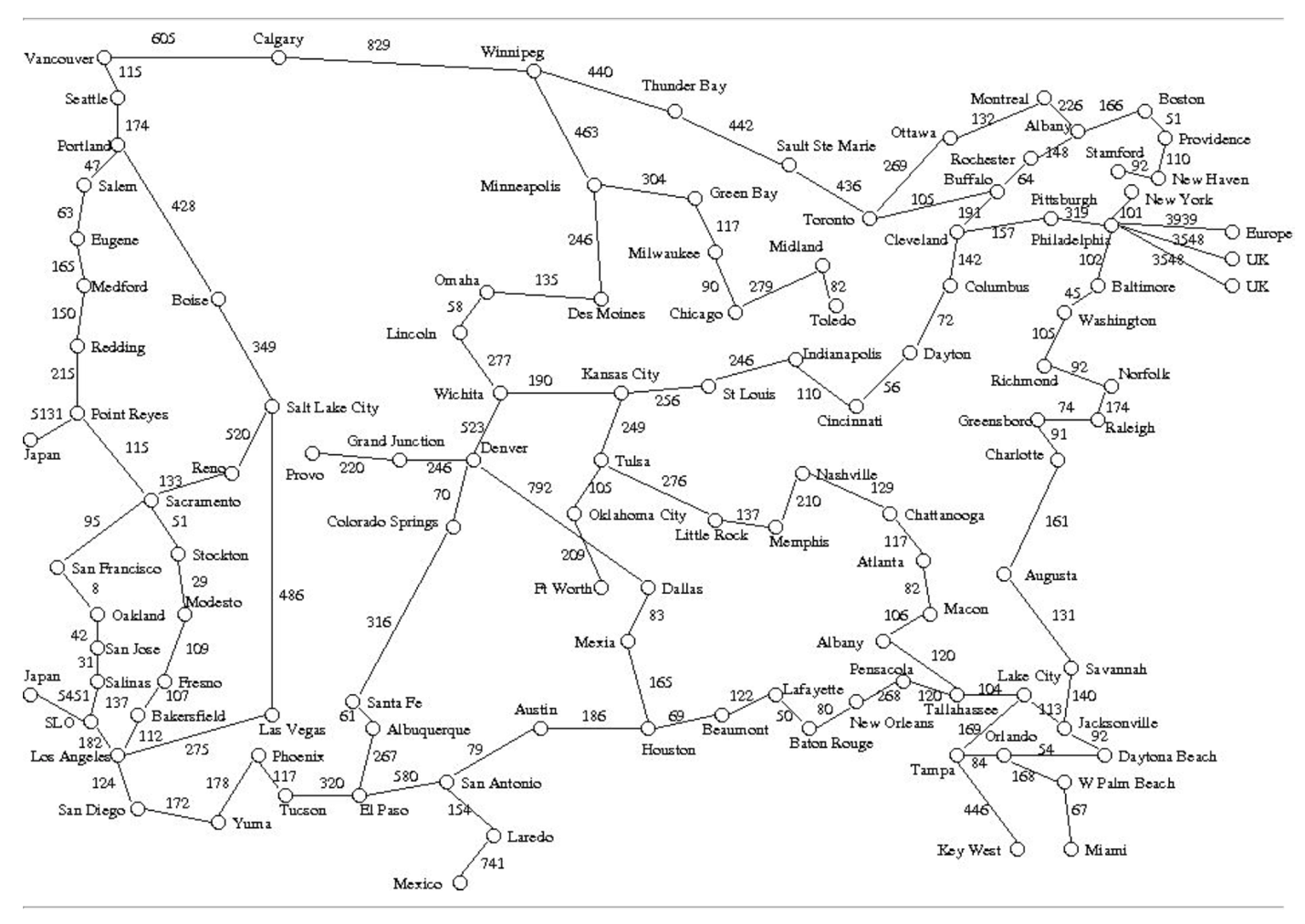
**Introduction to Artificial Intelligence: Assignment 2**

1. This question concerns route-finding, with comparison of several search algorithms. This time, we're in the U.S. Here's jpg of the map below. The solution consists of the series of cities a network packet must pass through, each city connected to one or more others by network links of the indicated length. There are no other network links.



* 1. Experiment with executing your implementation of A\* to find various paths, until you understand the meaning of the output. Are there any pairs of cities (A,B) for which the algorithm finds a different path from B to A than from A to B? Are there any pairs of cities (A,B) for which the algorithm expands a different total number of nodes from B to A than from A to B?

Solution: Japan to DesMoines has a different path from Japan to DesMoines and back.

Also, they have different number of expanded nodes while going from Japan to DesMoines and back.

* 1. Change your code so as to implement **greedy** search, as discussed in the web notes.

Solution: Code provided as an attachment

* 1. Do enough exploration to find at least one path that is longer using greedy search than that found using A\*, or to satisfy yourself  that there are no such paths. Find at least one path that is found by expanding more nodes than the comparable path using A\*, or satisfy yourself that there are no such paths. If there is such a path, list the nodes in the path and the total distance.

Solution: **Greedy** path from **Rochester – Ottawa** is longer than the **astar** path from **Rochester-Ottawa**.

For node expansion, the number of nodes expanded between **Wichita - Vancouver** for greedy search is more than for astar.

1.4 Change your code so as to implement **uniform cost** search, as discussed in the web notes.

Solution: Code provided as an attachment.

1.5 Do enough exploration to find at least one path that is longer using uniform cost than that found using A\*, or to satisfy yourself  that there are no such paths. Find at least one path that is found by expanding more nodes than the comparable path using A\*, or satisfy yourself that there are no such paths. If there is such a path, list the nodes in the path and the total distance.

Solution: No paths could be found for which uniform cost is longer than A\*. Number of nodes expanded for **uniform cost search** from **Rochester to Ottawa** came out to be more than the **A\*** search from **Rochester to Ottawa**.

1.6 As part of your answer, compare the solution paths and explain what happened, especially any weird behavior you might detect.

Solution: The observations were found that Uniform is the most optimal path in terms of path-cost when compared with A\* and greedy.

Also, amongst the 3 Greedy search results in the highest path cost because it doesn’t take into consideration the path cost and hence lacks in optimality.

2. [20 points] Tic-tac-toe (also known as Noughts and Crosses) is a two-person, zero-sum game, in which player X and player O alternate placing their symbols in one of the blank spaces in a 3-by-3 grid that looks like this:

The first player to place three of his symbols in a row -- horizontally, vertically, or diagonally -- wins.

* 1. Beginning from the position, with X's turn to move, construct by hand the game-tree for the rest of the game. Assume the search horizon is the end of the game on all branches.
  2. Suppose the static evaluation function scores a win for O as +1, a draw as 0, and a loss for O as -1. At each level of your sketch of  the game tree, indicate the value of each node based on its children. Indicate the best next move for X.
  3. Now use α-β pruning. At each level of your sketch of the game tree, indicate the value of the nodes and indicate any nodes that  would be pruned. Explicitly indicate the final value of the α-β interval is for each node. Indicate the best next move for X.

Solution: Please find the scanned pdf attached by the name ticTacToe.pdf.

3. [12 points] Use Propositional Logic to determine whether or not the following set of requirements is logically consistent. In other words, represent the following sentences in Propositional Logic, convert to Conjunctive Normal Form, and run Resolution until a contradiction is derived, or else show a model of all the expressions showing that no contradiction exists.

The system is in multiuser state if and only if it is operating normally. If the system is operating normally, the kernel is functioning. Either the kernel is not functioning or the system is in interrupt mode. If the system is not in multiuser state, then it is in interrupt mode. The system is not in interrupt mode.

Use the following lexicon:

Propositional symbols:

m -- The system is in multiuser state.

n -- The system is operating normally.

k -- The kernel is functioning.

i -- The system is in interrupt mode.

Solution: Propositional Logic(PL):

1. m ⇔ n --🡪 m ⇒ n and n ⇒ m
2. n ⇒ k
3. m ⇒ i

Conjunctive Normal Form(CNF):

1. i

-------------

-------------

1. m using 6 + 5
2. n using 7 + 1
3. k using 8 + 3
4. i using 4 + 9
5. using 10 + 6

4. Consider the following English sentences.

All cats are mammals. The head of any cat is the head of a mammal.

* 1. [6 points] Convert the sentences into first-order predicate logic. Be extremely careful about quantification; because not everything in the universe is a mammal, or a cat, or a head, you will need both kinds of quantification. Use the following lexicon:

Predicates: cat(X) -- X is a cat.

* 1. mammal(X) -- X is a mammal.
  2. headOf(H,X) -- H is the head of X.
  3. [4 points] Convert the logic statements into CNF. HINT: With this lexicon, you will need two Skolem constants if you're doing this correctly.
  4. [6 points] Using resolution and the 4-part heuristic presented in class, prove using FOPL resolution that the second of the original sentences follows from the first. Number your clauses, and indicate explicitly step-by-step what resolves together, under what substitution.
  5. Show a shorter proof that doesn't use the heuristic, if you can find one.

Solution:

Solemnizing with a

Solemnizing with b

1. Conjunctive Normal Form(CNF):

Negating CNF (~CNF):

1. 1.

2. cat(X)

3. Head(b,X)

4.

----------------------------------------------------------------------------------------

5. 4 + 1 and X/a

6. 5 + 2 and X/a

7. 6 + 3 and X/a

Here, it is observed that upon resolution the result comes out to be a NULL set, the problem set is consistent.

1. There is no shorter proof to be found.

------------------------------------------------------------------------------------------------------------