National University of Computer & Emerging Sciences, Karachi
Spring -2023
AI-2002 Artificial Intelligence
Assignment#1

Due Date: 22. Feb.2023 **Max Marks**: 10 points

Q1: Read Turing's original paper on AI (Turing, 1950). In the paper, he discusses several potential objections to his proposed enterprise and his test for intelligence. Which objections still carry some weight? Are his refutations valid? Can you think of new objections arising from developments since he wrote the paper? In the paper, he predicts that by the year 2000, a computer will have a 30% chance of passing a five-minute Turing Test with an unskilled interrogator. Do you think this is reasonable?

Q2: Examine the AI literature to discover whether or not the following tasks can currently be solved by computers. **(Any five)**

1. Playing a decent game of table tennis (ping-pong).
2. Driving in the centre of Karachi.
3. Playing a decent game of bridge at a competitive level.
4. Discovering and proving new mathematical theorems.
5. Writing an intentionally funny story.
6. Giving competent legal advice in a specialized area of law.
7. Translating spoken English into spoken Urdu in real time.

For the currently infeasible tasks, try to find out what the difficulties are?

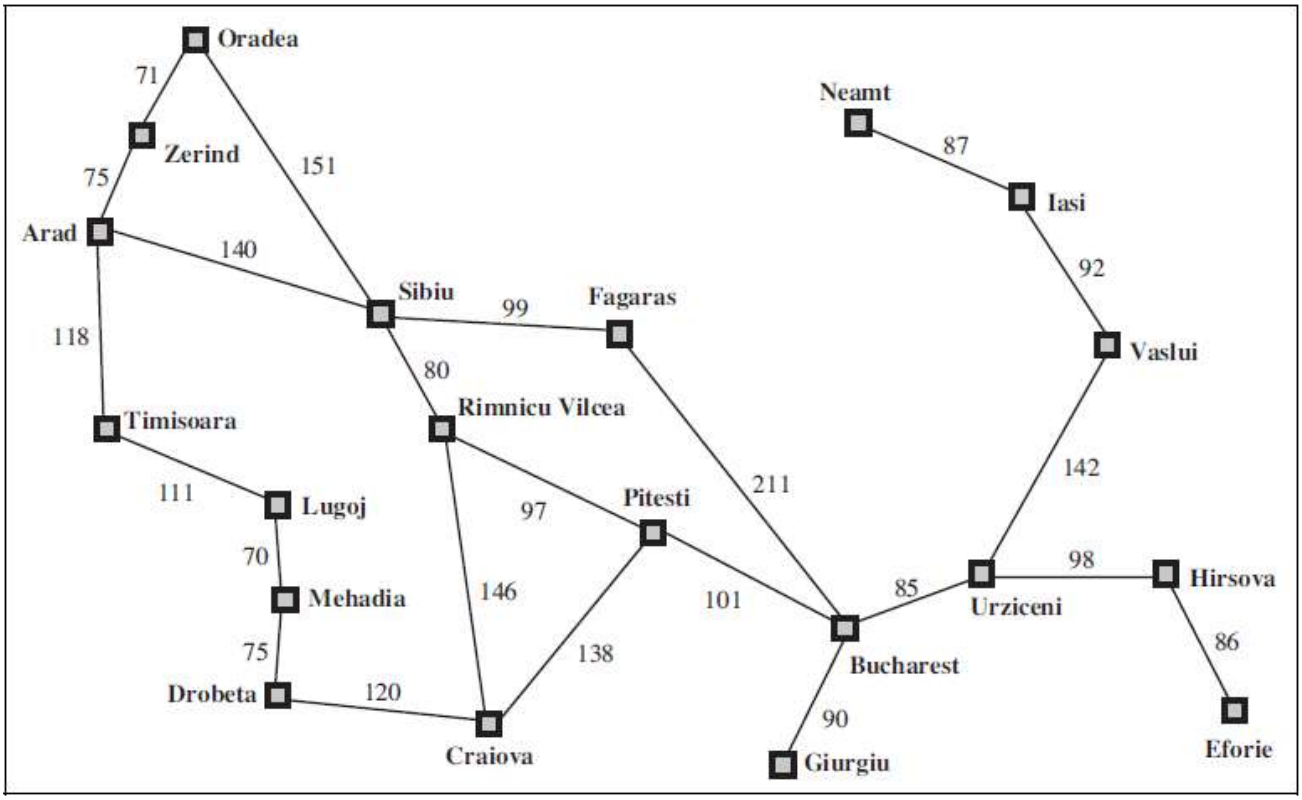
Q3: Choose a domain that you are familiar with, and write a PAGE description of an agent for the environment. Characterize the environment as being accessible, deterministic, episodic, static, and continuous or not. What agent architecture is best for this domain?

Q4: For each of the following activities, give a PEAS description of the task environment. **(Any five)**
- Playing soccer.
- Exploring the subsurface of Arabian Sea.
- Shopping for used AI books on the Internet.
- Playing a tennis match.
- Practicing tennis against a wall.
- Performing a high jump.
- Knitting a sweater.
- Bidding on an item at an auction.

Q5: For each of the following assertions, say whether it is true or false and support your answer with examples or counter examples where appropriate. **(Any five)**
1. An agent that senses only partial information about the state cannot be perfectly rational.
2. There exist task environments in which no pure reflex agent can behave rationally.
3. There exists a task environment in which every agent is rational.
4. The input to an agent program is the same as the input to the agent function.
5. Every agent function is implementable by some program/machine combination.
6. Suppose an agent selects its action uniformly at random from the set of possible actions. There exists a deterministic task environment in which this agent is rational.
7. It is possible for a given agent to be perfectly rational in two distinct task environments.

P.S. You are free to use any particular language for Q6 and Q7 (Coding Based).

Q6: You are clear about uninformed and informed strategies now. The example discussed in the class regarding Romania map has to get implemented in this assignment.



Arad	366	Mehadia	241
Bucharest	0	Neamt	234
Craiova	160	Oradea	380
Drobeta	242	Pitesti	100
Eforie	161	Rimnicu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Hirsova	151	Urziceni	80
Iasi	226	Vaslui	199
Lugoj	244	Zerind	374

The map is directly taken up from your book together with the heuristics table. The task is to reach from a particular source to destination using different strategies. This means that user will be facilitated with the option of choosing any random source and destination point at run time. Following are the strategies to be implemented.

a) Breadth first search
b) Uniform cost search
c) Greedy best first search
d) Iterative deepening depth first search

A comparison of these four needs to be done. Complete list of pathway and path cost of each algorithm has to be calculated so that it shows clearly that which algorithm is best out of all in ascending order

Q7: N-Queens problem: There is an n x n grid where the value of n is 4≤n≤8 (user shall be asked at run time for the value of n he wants to keep). Your task is to place n queens on this board. As per the rules of chess, a queen should have no other queen in its respective column, neither should it have any other queen in its row nor should it have any within its diagonal cells. You can consider this case as placing each queen individually per column such that it does not violate any of the constraints mentioned. It's quite easy to find solution manually but your task is now to code it and find the correct positions for the queens.

Q1	x	x	x
X	x		
X		x	
x			X

This problem has to be solved through the concept of backtracking. So initially you will place the first queen randomly at any location within column 1. With respect to its location, now next n-1 queens' domain i.e. the places where they can be placed might shrink up.
E.g. in this case 4 queens have to placed and Q1 is placed on (0,0) so x positions represent the illegal places now where other queens cannot be placed due to Q1 placement. This means for rest of the queens some positions have been considered as illegal.
Further up, when we will move forward, there might be a point where domain gets empty for any particular queen, at that instance apply backtrack concept, which means that location of previous queen will have to get changed.
Advise: Start building up your logic first on 4 queen problem so that you may exactly understand the flow then go for making a generalized code version of n queens' placement.

Comments your code where necessary to make it clean and clear.