List all algorithms and methods that we have covered in this course. Write 3 sentences to describe what each algorithm and method solves etc.

1)Depth First Search(DFS) :

In the depth first search the first most node is considered as root node. The name itself tells us the depth first search which means we should expand the deepest node first. It starts executing the left most branch first and explores all its nodes. It goes from left to right. It follows last in first out principle.

2)Breadth First Search(BFS) :

In BFS the first node is considered as a parent node which will be root node. It is top down approach. It explores the shallow node first. In BFS, first it explores all its neighbor nodes in the current level and next it explores the nodes in the lower level in the same way.

3)Uniform Cost Search :

In uniform cost search it first explores the node which has the minimum cost. This uses priority queue. Here the nodes are explored by considering its cost from the root node. The node which has the lowest cost will be explored first. It follows the same way and explores all the nodes in the order of having the least cost.

4) Greedy Search :

Here we will expand the node which will be closest to the goal state. Here we find the nearest node to the current node and will expand that node. The problem with greedy is sometimes it may not be the optimal or shortest path.

5) A\* Search :

In A\* search we use heuristics. We will calculate the cost for each node which is f(n). Here f(n) = g(n) [backward cost] + h(n) [forward cost]. It considers the start state and by using the f(n) function for each node it expands to the neighboring node for which f(n) is smallest and reach the goal state.

 6)Dijkstra's algorithm :

It is generally used to find the shortest path between the start state and goal state. Here we consider the start state and calculate the distance to its neighbors and then it explores the path having the least distance from the current state and finally reaches the goal state by doing the same process at every state.

7) Kruskal's Algorithm :

It is also the minimum spanning tree where it considers the edge having the least possible weight. We should consider a start state and goal state and then add edges in the order of increasing weights and we should take care that it does not forms a cycle.

8)Prim's Algorithm :

In prims algorithm we generally consider the start state and it finds the subsets of the edges which includes all the vertices where total weight of all the edges in the tree will be minimized. IT also used to find minimum spanning tree for weighted undirected graph.

9)Minimax Algorithm :

Mini max algorithm is generally used in game designing. It has two agents. For each step we try to get the best possible solution for the next move and then we get the worst possible solution that can made the opponent for us and we perform this process continuously until we reach the base case.

10) Expectimax Algorithm :

In Expectimax, we should compute the average score under optimal policy. We should consider the max nodes as in minimax search. Here the chance nodes will be the minimum nodes and the output will not be certain and we should calculate the expected utilities by considering the weighted average of children.

11) Markov Decision Process :

Generally, MDP is defined using set of states, action and transition function, reward function, start state and terminal state. We can solve the problem using expectimax search. Here we require optimal plan to travel from stat to goal where policy gives action for each state and optimal policy maximizes the utilities. Explicit policy will define the reflex agent.

12) Reinforcement Learning :

It is kind of dynamic programming which trains the algorithm using system of rewards and punishments. In this algorithm the agent reacts with the environment and receive rewards if it performs correctly and penalties if it performs incorrectly. By using the rewards and penalties the agents train itself and tries to improve for the upcoming iterations and gives us the best possible output.

13) Q Learning :

Q learning is type of reinforcement learning algorithm, Q learning utilizes a Q function and gives a reward based on state and action. Based on the reward, the function agent will learn and update its own reward.

14) Alpha Beta Pruning :

In Alpha beta pruning it is also used for designing video games, it is also used for two player games. Here we don’t search the whole tree, if we get the desired one, we can avoid to search that part of the tree. It is similar to minimax where alpha is the minimal score where max player gets it and beta will be the maximum score that player hope to get. If the value is greater than alpha those nodes are not explored. This is alpha beta pruning.