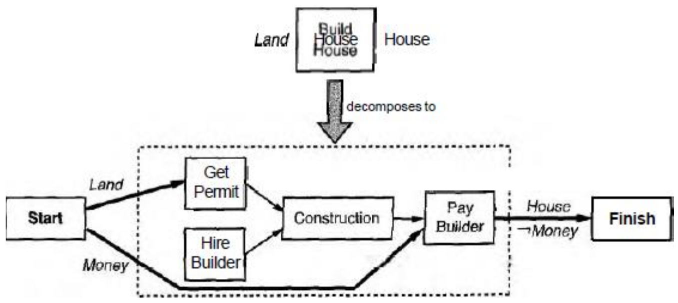
1.	List the importance of hierarchical task networks This section describes a planning method based on **hierarchical task networks** orHTNs. The approach we take combines ideas from both partial-order planning and the area known as "HTN planning." In HTN planning, the initial plan, which describesthe problem, is viewed as a very high-level description of what is to be done-eg building a house.
2.	Give four planning methods for handling indeterminacy. • Sensorless planning • Conditional planning • Execution monitoring and replanning • Continuous planning
3.	What is state set and belief set? The simplest way to model this situation is to say that the initial state belongs to a **state set;** the state set is away of describing the agent's initial **belief state.**
4.	Draw the One possible decomposition for the BuildHouseaction.  Land Build House House decomposes to Land Money Start Get Permit Hire Builder Construction Pay Builder House ¬Money Finish
5.	Define job shop Scheduling. Time is of the essence in the general family of appllications called **job shop scheduling.** Such tasks require completing a set of jobs, each of whi.ch consists of a sequence of actions, where each action has a given duration and might require some resources
6.	Which gives the expected error in decision trees? Give a note on it. **Gini index** $2p^{\cdot}(1-p^{\cdot})$ – this is the expected error if we label examples in the leaf randomly: positive with probability $p^{\cdot}$ and negative with probability $1-p^{\cdot}$. The probability of a false positive is then $p^{\cdot}(1-p^{\cdot})$ and the probability of a false negative$(1-p^{\cdot})p^{\cdot}$
7.	Differentiate passive learning and active learning. **Passive learning,** where the agent's policy is fixed and the task is to learn the utilities of states (or state-action pairs); **active learning,** where the agent must also EXPLORATION learn what to do.
8.	Write the Widrow-Hoff rule or delta rule.

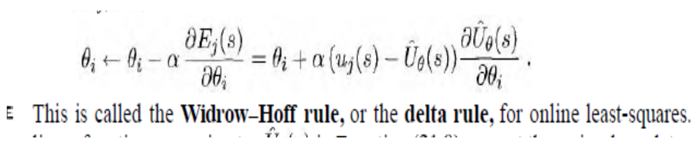
	$$\theta_i \leftarrow \theta_i - \alpha \frac{\partial E_j(s)}{\partial \theta_i} = \theta_i + \alpha\,(u_j(s) - \hat{U}_\theta(s))\frac{\partial \hat{U}_\theta(s)}{\partial \theta_i}\,.$$ This is called the **Widrow–Hoff rule,** or the **delta rule,** for online least-squares.
9.	How does policy search methods operate? **Policy search** methods operate directly on a representation of the policy, attemptingto improve it based on observed performance. The variance in the performance in astochastic domain is a serious problem; for simulated dlomains this can be overcome byfixing the randomness in advance
10.	State the algorithm for an exploratory Q-learning agent. **function** Q-LEARNING-AGENT(*percept*) **returns** an action **inputs:** *percept,* a percept indicating the current state s' and reward signal r' **static:** Q, a table of action values index by state and action N_{sa}, a table of frequencies for state-action pairs s, a, r, the previous state, action, and reward, initially null **if** s is not null **then do** increment $N_{sa}[s, a]$ $Q[a,s] \leftarrow Q[a,s] + \alpha(N_{sa}[s,a])(r + \gamma \max_{a'} Q[a',s'] - Q[a,s])$ **if** TERMINAL?[s'] **then** $s, a, r \leftarrow$ null **else** $s, a, r \leftarrow s', \operatorname{argmax}_{a'} f(Q[a',s'], N_{sa}[s',a']), r'$ **return** a
11.	What is critical path? The critical path is that path whose total duration is longest; the path is "critical" because it determines the duration of the entire plan
12.	Mention the importance of hierarchical task networks This section describes a planning method based on hierarchical task networks orHTNs. The approach we take combines ideas from both partial-order planning and the area known as "HTN planning." In HTN planning, the initial plan, which describesthe problem, is viewed as a very high-level description of what is to be done-eg building a house.
13.	List down the different types of nodes in Decision Trees. The Decision Tree consists of the following different types of nodes: 1. Root node: It is the top-most node of the Tree from where the Tree starts.

	2. Decision nodes: One or more Decision nodes that result in the splitting of data into multiple data segments and our main goal is to have the children nodes with maximum homogeneity or purity. 3. Leaf nodes: These nodes represent the data section having the highest homogeneity.
14.	Which should be preferred among Gini impurity and Entropy? In reality, most of the time it does not make a big difference: they lead to almost similar Trees. Gini impurity is a good default while implementing in sklearn since it is slightly faster to compute. However, when they work in a different way, then Gini impurity tends to isolate the most frequent class in its own branch of the Tree, while entropy tends to produce slightly more balanced Trees.
15.	Define job shop Scheduling. Time is of the essence in the general family of appllications called job shop scheduling. Such tasks require completing a set of jobs, each of whi.ch consists of a sequence of actions, where each action has a given duration and might require some resources
16.	List out the disadvantages of the Decision Trees 1. Overfitting: This is the major problem associated with the Decision Trees. It generally leads to overfitting of the data which ultimately leads to wrong predictions for testing data points. it keeps generating new nodes in order to fit the data including even noisy data and ultimately the Tree becomes too complex to interpret. In this way, it loses its generalization capabilities. Therefore, it performs well on the training dataset but starts making a lot of mistakes on the test dataset. 2. High variance: As mentioned, a Decision Tree generally leads to the overfitting of data. Due to the overfitting, there is more likely a chance of high variance in the output which leads to many errors in the final predictions and shows high inaccuracy in the results. So, in order to achieve zero bias (overfitting), it leads to high variance due to the bias-variance tradeoff. 3. Unstable: When we add new data points it can lead to regeneration of the overall Tree. Therefore, all nodes need to be recalculated and reconstructed. 4. Not suitable for large datasets: If the data size is large, then one single Tree may grow complex and lead to overfitting. So in this case, we should use Random Forest instead, an ensemble technique of a single Decision Tree.

17. List out the difference between passive reinforcement learning and active reinforcement learning.

Passive learning, where the agent's policy is fixed and the task is to learn the utilities of states (or state-action pairs); active learning, where the agent must also EXPLORATION learn what to do.

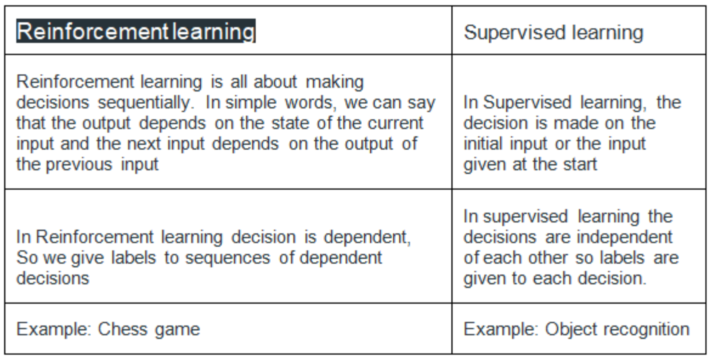
18. Write the Widrow-Hoff rule or delta rule.



$$\theta_i \leftarrow \theta_i - \alpha \frac{\partial E_j(s)}{\partial \theta_i} = \theta_i + \alpha\,(u_j(s) - \hat{U}_\theta(s))\frac{\partial \hat{U}_\theta(s)}{\partial \theta_i}\,.$$

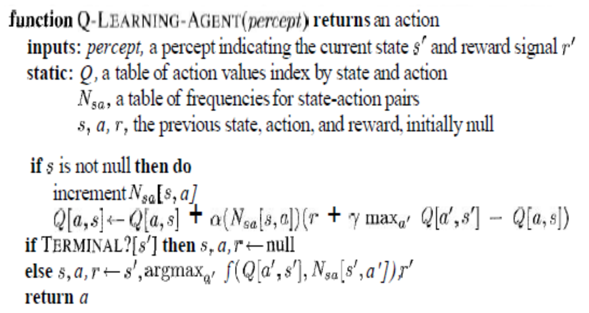
This is called the **Widrow–Hoff rule,** or the **delta rule,** for online least-squares.

19. Mention the difference between Reinforcement learning and Supervised learning:



Reinforcement learning	Supervised learning
Reinforcement learning is all about making decisions sequentially. In simple words, we can say that the output depends on the state of the current input and the next input depends on the output of the previous input	In Supervised learning, the decision is made on the initial input or the input given at the start
In Reinforcement learning decision is dependent, So we give labels to sequences of dependent decisions	In supervised learning the decisions are independent of each other so labels are given to each decision.
Example: Chess game	Example: Object recognition

20. State the algorithm for an exploratory Q-learning agent.



```
function Q-LEARNING-AGENT(percept) returns an action
  inputs: percept, a percept indicating the current state $s'$ and reward signal $r'$
  static: $Q$, a table of action values index by state and action
          $N_{sa}$, a table of frequencies for state-action pairs
          $s, a, r$, the previous state, action, and reward, initially null

  if $s$ is not null then do
     increment $N_{sa}[s, a]$
     $Q[a,s] \leftarrow Q[a,s] + \alpha(N_{sa}[s,a])(r + \gamma \max_{a'} Q[a',s'] - Q[a,s])$
  if TERMINAL?[$s'$] then $s, a, r \leftarrow$ null
  else $s, a, r \leftarrow s', \mathrm{argmax}_{a'} f(Q[a',s'], N_{sa}[s',a']), r'$
  return $a$
```