Artificial Intelligence

1.What are intelligent systems? Explain with examples?

Intelligent system can be defined as the system that incorporates intelligence into applications being handled by machines.

In order to classify system as intelligent it is necessary to define intelligence.

IN earliest best method to guage the intelligence of a system was TURING TEST proposed by Alan Turing in 1950.

He said that system is intelligent if it has the ability to pass the Turing Test.

TURING TEST is informally defined as:

A system is said to have passed the Turing Test,if a human questionnaire is unable to determine from repeated questions of any kind,whether he / she is talking to a system or to a machine/system.

Example:-

ELIZA

The very first intelligent system named ELIZA passed the TURING TEST which was written by Weizenbaum during 1964 to 1966 at MIT.

•A program that conversed with user in English

•Able to converse about any subject, since it stored the subjects information in data banks.

•It was able to pick up speech patterns from user’s questions and provide responses

➢ The main characteristics of Eliza are mentioned as:

1)Simulation of intelligence,

2) Quality of response

3)Coherence,

4)Semantics

2.Write the history of AI in brief?

➢1931, Goedel layed the foundation of Theoretical Computer Science 1920-30s.

➢ 1936, Turing reformulated Goedel’s result and church’s extension thereof .

➢1943: The first work which is now recognized as AI was done by Warren McCulloch and Walter pits in 1943.

1949: Donald Hebb demonstrated an updating rule for modifying the connection strength between neurons

➢1950: The Alan Turing introduced the turing test.

➢1956: The word "Artificial Intelligence" first adopted by American Computer scientist John McCarthy at the Dartmouth Conference.

➢1957,aprogramnamedGPS(GeneralProblemSolver)was developed and tested on problems requiring common sense.

➢1958, John McCarthy announced his new developmenti.e.,LISP (LIStProcessinglanguage).

➢1966: Joseph Weizenbaum created the first chatbot in 1966, which was named as ELIZA.

➢ 1970:R.Kowalski developed PROLOG language around 1970.

➢ Year 1980: After AI winter duration, AI came back with "Expert System"

➢ Year 1997: In the year 1997, IBM Deep Blue beats world chess champion, Gary Kasparov, and became the first computer to beat a world chess champion.

➢ Year 2002: for the first time, AI entered the home in the form of Roomba, a vacuum cleaner.

➢ Year 2006: AI came in the Business world till the year 2006. Companies like Facebook, Twitter, and Netflix also started using AI.

➢Year 2011: In the year 2011, IBM's Watson won jeopardy, a quiz show, where it had to solve the complex questions as well as riddles.

➢ Year 2012: Google has launched an Android app feature "Google now", which was able to provide information to the user as a prediction.

➢ Year 2014: In the year 2014, Chatbot "Eugene Goostman" won a competition in the infamous "Turing test."

➢ By 2016, the market for AI-related products, hardware, and software reached more than 8 billion dollars .

➢ Year 2018: The "Project Debater" from IBM debated on complex topics with two master debaters and also performed extremely well.

3.Explain the components of AI?

Learning

There are a number of different forms of learning as applied to artificial intelligence. The simplest is learning by trial and error. For example, a simple computer program for solving mate-in-one chess problems might try moves at random until mate is found. The program might then store the solution with the position so that the next time the computer encountered the same position it would recall the solution. This simple memorizing of individual items and procedures—known as rote learning—is relatively easy to implement on a computer

Reasoning

To reason is to draw inferences appropriate to the situation. Inferences are classified as either deductive or inductive. An example of the former is, “Fred must be in either the museum or the café. He is not in the café; therefore he is in the museum,” and of the latter, “Previous accidents of this sort were caused by instrument failure; therefore this accident was caused by instrument failure.” The most significant difference between these forms of reasoning is that in the deductive case the truth of the premises guarantees the truth of the conclusion, whereas in the inductive case the truth of the premise lends support to the conclusion without giving absolute assurance. Inductive reasoning is common in science, where data are collected and tentative models are developed to describe and predict future behaviour—until the appearance of anomalous data forces the model to be revised. Deductive reasoning is common in mathematics and logic, where elaborate structures of irrefutable theorems are built up from a small set of basic axioms and rules.

Problem solving

Problem solving, particularly in artificial intelligence, may be characterized as a systematic search through a range of possible actions in order to reach some predefined goal or solution. Problem-solving methods divide into special purpose and general purpose. A special-purpose method is tailor-made for a particular problem and often exploits very specific features of the situation in which the problem is embedded. In contrast, a general-purpose method is applicable to a wide variety of problems. One general-purpose technique used in AI is means-end analysis—a step-by-step, or incremental, reduction of the difference between the current state and the final goal

Perception

In perception the environment is scanned by means of various sensory organs, real or artificial, and the scene is decomposed into separate objects in various spatial relationships. Analysis is complicated by the fact that an object may appear different depending on the angle from which it is viewed, the direction and intensity of illumination in the scene, and how much the object contrasts with the surrounding field.

Language

A language is a system of signs having meaning by convention. In this sense, language need not be confined to the spoken word. Traffic signs, for example, form a minilanguage, it being a matter of convention that ⚠ means "hazard ahead" in some countries. It is distinctive of languages that linguistic units possess meaning by convention, and linguistic meaning is very different from what is called natural meaning, exemplified in statements such as "Those clouds mean rain" and "The fall in pressure means the valve is malfunctioning."

4.Describe the Missionaries and Cannibals problem using production system?

■ Problem Statement: Three missionaries and three cannibals want to cross a river. There is a boat on their side of the river that can be used by either one or two persons.

❑How should they use this boat to cross the river in such a way that cannibals never outnumber missionaries on either side of the river? If the cannibals ever outnumber the missionaries (on either bank) then the missionaries will be eaten. How can they all cross over without anyone being eaten?

■ PS for this problem can be described as the set of ordered pairs of left and right bank of the river as (L, R) where each bank is represented as a list [nM, mC, B]

❑ n is the number of missionaries M, m is the number of cannibals C , and B represents boat.

■ Start state: ([3M, 3C, 1B], [0M, 0C, 0B]),

❑ 1B means that boat is present and 0B means it is not there on the bank of river.

■ Goal state: ([0M, 0C, 0B], [3M, 3C, 1B])

Set of Production Rules
Applied keeping constrains in mind

RN	Left side of rule	→	Right side of rule
	Rules for boat going from left bank to right bank of the river		
L1	([n_1M, m_1C, 1B], [n_2M, m_2C, 0B])	→	([(n_1-2)M, m_1C, 0B], [(n_2+2)M, m_2C, 1B])
L2	([n_1M, m_1C, 1B], [n_2M, m_2C, 0B])	→	([(n_1-1)M,(m_1-1)C,0B],[(n_2+1)M,(m_2+1)C, 1B])
L3	([n_1M, m_1C, 1B], [n_2M, m_2C, 0B])	→	([n_1M, (m_1-2)C, 0B], [n_2M, (m_2+2)C, 1B])
L4	([n_1M, m_1C, 1B], [n_2M, m_2C, 0B])	→	([(n_1-1)M, m_1C,0B],[(n_2+1)M, m_2C, 1B])
L5	([n_1M, m_1C, 1B], [n_2M, m_2C, 0B])	→	([n_1M, (m_1-1)C, 0B], [n_2M, (m_2+1)C, 1B])
	Rules for boat coming from right bank to left bank of the river		
R1	([n_1M, m_1C, 0B], [n_2M, m_2C, 1B])	→	([(n_1+2)M, m_1C, 1B], [(n_2-2)M, m_2C, 0B])
R2	([n_1M, m_1C, 0B], [n_2M, m_2C, 1B])	→	([(n_1+1)M,(m_1+1)C,1B],[(n_2-1)M,(m_2-1)C, 0B])
R3	([n_1M, m_1C, 0B], [n_2M, m_2C, 1B])	→	([n_1M, (m_1+2)C, 1B], [n_2M, (m_2-2)C, 0B])
R4	([n_1M, m_1C, 0B], [n_2M, m_2C, 1B])	→	([(n_1+1)M, m_1C,1B],[(n_2-1)M, m_2C, 0B])
R5	([n_1M, m_1C, 0B], [n_2M, m_2C, 1B])	→	([n_1M, (m_1+1)C, 1B], [n_2M, (m_2-1)C, 0B])

Solution Path for Missionaries and Cannibals Problem

Rule Number	([3M, 3C, 1B] , [0M, 0C, 0B]) ← Start State
L2:	([2M, 2C, 0B] , [1M, 1C, 1B])
R4:	([3M, 2C, 1B] , [0M, 1C, 0B])
L3:	([3M, 0C, 0B] , [0M, 3C, 1B])
R5:	([3M, 1C, 1B] , [0M, 2C, 0B])
L1:	([1M, 1C, 0B] , [2M, 2C, 1B])
R2:	([2M, 2C, 1B] , [1M, 1C, 0B])
L1:	([0M, 2C, 0B] , [3M, 1C, 1B])
R5:	([0M, 3C, 1B] , [3M, 0C, 0B])
L3:	([0M, 1C, 0B] , [3M, 2C, 1B])
R5:	([0M, 2C, 1B] , [3M, 1C, 0B])
L3:	**([0M,0C,0B],[3M, 3C, 1B]) → Goal State**

5.Write about applications of AI?

1. AI in Astronomy

•Artificial Intelligence can be very useful to solve complex universe problems. AI technology can be helpful for understanding the universe such as how it works, origin, etc.

2. AI in Healthcare

•Healthcare Industries are applying AI to make a better and faster diagnosis than humans. AI can help doctors with diagnoses and can inform when patients are worsening so that medical help can reach to the patient before hospitalization

3. AI in Gaming

•AI can be used for gaming purpose. The AI machines can play strategic games like chess, where the machine needs to think of a large number of possible places.

4. AI in Finance

•The finance industry is implementing automation, chatbot, adaptive intelligence, algorithm trading, and machine learning into financial processes.

5. AI in Data Security

•The security of data is crucial for every company and cyber-attacks are growing very rapidly in the digital world. AI can be used to make your data more safe and secure

. 6. AI in Social Media

•Social Media sites such as Facebook, Twitter, and Snapchat contain billions of user profiles, which need to be stored and managed in a very efficient way. AI can organize and manage massive amounts of data.

7. AI in Travel & Transport

•AI is becoming highly demanding for travel industries. AI is capable of doing various travel related works such as from making travel arrangement to suggesting the hotels, flights, and best routes to the customers.

8. AI in Automotive Industry

•Some Automotive industries are using AI to provide virtual assistant to their user for better performance. Such as Tesla has introduced TeslaBot, an intelligent virtual assistant.

6.What are the general problem solving techniques?

- Problem solving is a process of generating solutions from observed or given data. It is however not always possible to use direct methods (i.e. go directly from data to solution). Instead, problem solving often needs to use indirect or model based methods.

General Problem Solver

- ***(GPS)*** was a computer program created in 1957 by Simon and Newell to build a universal problem solver machine. *GPS* solved many simple problems, such as the Towers of Hanoi, but ***GPS could not solve any real-world problems***.

To build a system to solve a particular problem, we need to:

• Define the problem precisely – find input situations as well as final situations for an acceptable solution to the problem.

• Analyze the problem – find few important features that may have impact on the appropriateness of various possible techniques for solving the problem

. •Isolate and represent task knowledge necessary to solve the problem.

•Choose the best problem-solving technique(s) and apply to the particular problem.

7.Explain A* algorithm?

•A* Algorithm is one of the best and popular techniques used for path finding and graph traversals.
•A lot of games and web-based maps use this algorithm for finding the shortest path efficiently.

•It is essentially a best first search algorithm.

WorkingA*

Algorithm works as-

•It maintains a tree of paths originating at the start node

. •It extends those paths one edge at a time.

•It continues until its termination criterion is satisfied. A* Algorithm extends the path that minimizes the following function-

f(n) = g(n) + h(n)

The implementation of A* Algorithm involves maintaining two lists- OPEN and CLOSED.

OPEN contains those nodes that have been evaluated by the heuristic function but have not been expanded into successors yetCLOSED contains those nodes that have already been visited.

The algorithm is as follows

Step-01: Define a list OPEN. Initially, OPEN consists solely of a single node, the start node S.

Step-02: If the list is empty, return failure and exit.

Step-03: Remove node n with the smallest value of f(n) from OPEN and move it to list CLOSED. If node n is a goal state, return success and exit.

Step-04: Expand node n.

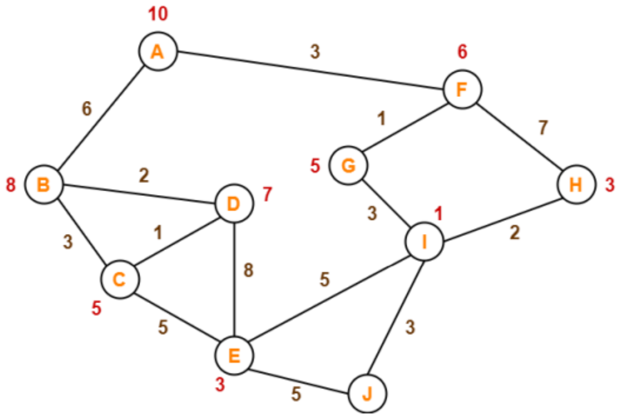
Step-05: If any successor to n is the goal node, return success and the solution by tracing the path from goal node to S. Otherwise, go to Step-06.

Step-06: For each successor node, Apply the evaluation function f to the node. If the node has not been in either list, add it to OPEN.

Step-07: Go back to Step-02.

A* Algorithm

- The numbers written on edges represent the distance between the nodes.
- The numbers written on nodes represent the heuristic value.
- Find the most cost-effective path to reach **from start state A to final state J** using A* Algorithm.



Step-01:

•We start with node A.
•Node B and Node F can be reached from node A.

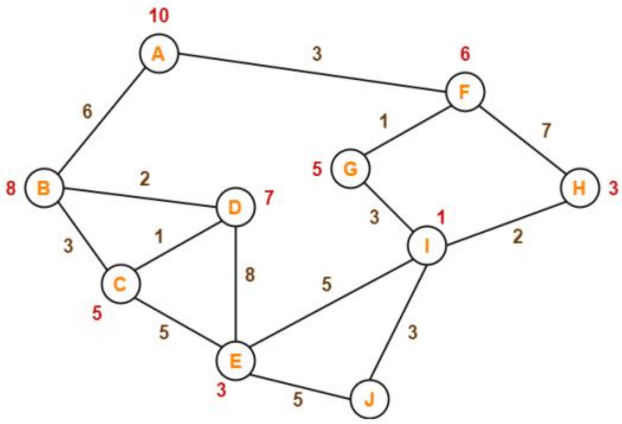
A* Algorithm calculates f(B) and f(F).
•f(B) = 6 + 8 = 14
•f(F) = 3 + 6 = 9

Since f(F) < f(B), so it decides to go to node F.

Path- A → F

ADITYA ENGINEERING COLLEGE(A)

A* Algorithm



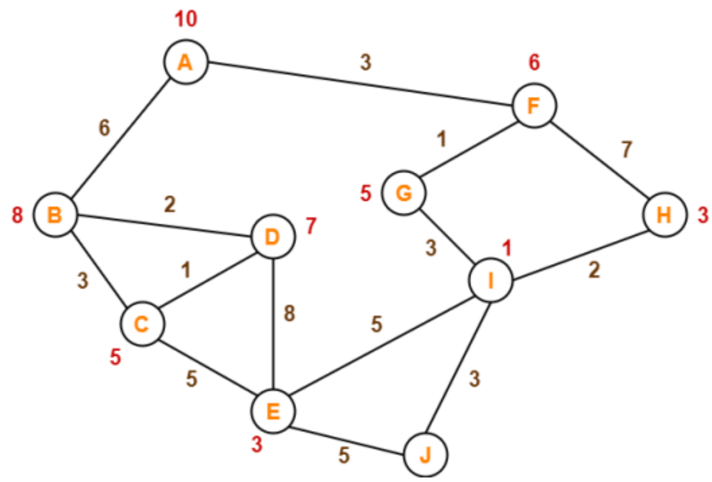
Step-02:

Node G and Node H can be reached from node F.

A* Algorithm calculates f(G) and f(H).
•f(G) = (3+1) + 5 = 9
•f(H) = (3+7) + 3 = 13

Since f(G) < f(H), so it decides to go to node G.

Path- A → F → G

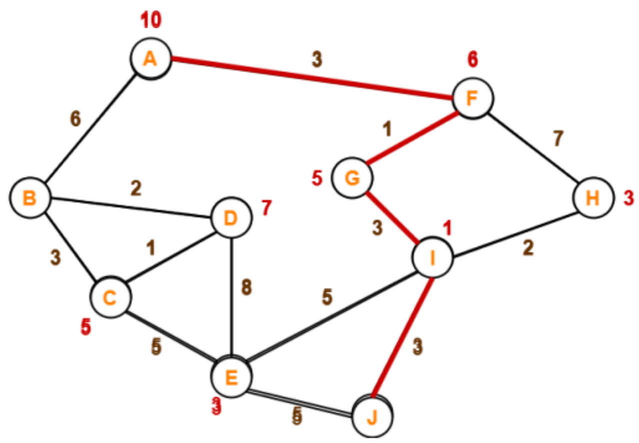


Step-03:

Node I can be reached from node G.

A* Algorithm calculates f(I).
f(I) = (3+1+3) + 1 = 8
It decides to go to node I.

Path- A → F → G → I



Step-04:

Node E, Node H and Node J can be reached from node I.

A* Algorithm calculates f(E), f(H) and f(J).
•f(E) = (3+1+3+5) + 3 = 15
•f(H) = (3+1+3+2) + 3 = 12
•f(J) = (3+1+3+3) + 0 = 10

Since f(J) is least, so it decides to go to node J.

Path- A → F → G → I → J
This is the required shortest path from node A to node J.

It is important to note that-
•A* Algorithm is one of the best path finding algorithms.
•But it does not produce the shortest path always.
•This is because it heavily depends on heuristics.

8.Explain the best first search algorithm.

The idea of Best First Search is to use an evaluation function to decide which adjacent is most promising and then explore.

Best First Search falls under the category of Heuristic Search or Informed Search.

- BFS uses Priority Queue to store costs of nodes.
- It is a Greedy Search Algorithm

Step 1: Place the starting node into the OPEN list.

Step 2: If the OPEN list is empty, Stop and return failure.

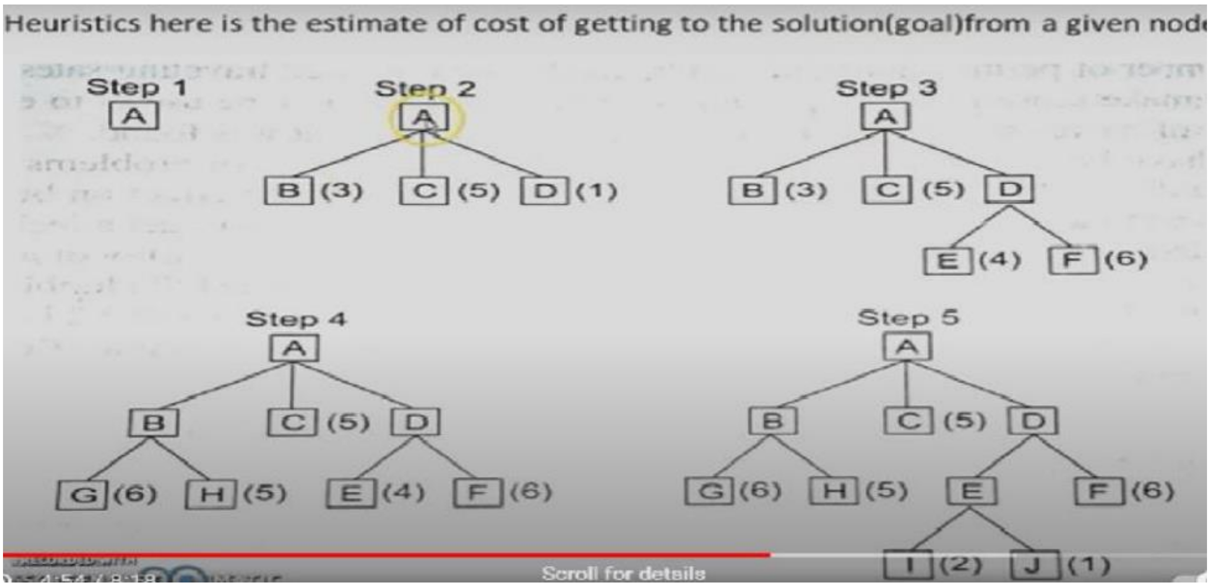
Step 3: Remove the node n, from the OPEN list which has the lowest value of h(n), and places it in the CLOSED list.

Step 4: Expand the node n, and generate the successors of node n.

Step 5: Check each successor of node n, and find whether any node is a goal node or not. If any successor node is goal node, then return success and terminate the search, else proceed to Step 6.

Step 6: For each successor node, algorithm checks for evaluation function f(n), and then check if the node has been in either OPEN or CLOSED list. If the node has not been in both list, then add it to the OPEN list.

Step 7: Return to Step 2.



Advantages:

• Best first search can switch between BFS and DFS by gaining the advantages of both the algorithms.

• This algorithm is more efficient than BFS and DFS algorithms.

• In worst case scenario the algorithm takes O(n*logn) time.

Disadvantages:

• It can behave as an unguided depth-first search in the worst case scenario.

• It can get stuck in a loop as DFS.

• This algorithm is not optimal

9.What is problem reduction? Give an example.

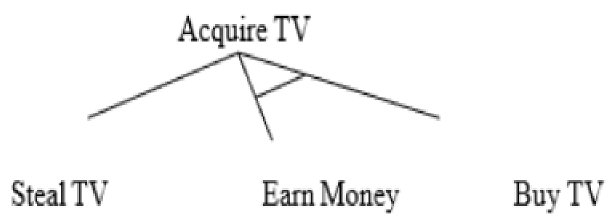
We already know about the divide and conquer strategy, a solution to a problem can be obtained by decomposing it into smaller sub-problems. Each of this sub-problem can then be solved to get its sub solution. These sub solutions can then be recombined to get a solution as a whole. That is called is Problem Reduction.

AND-OR graphs or AND - OR trees are used for representing the solution.

The decomposition of the problem or problem reduction generates AND arcs.

One AND are may point to any number of successor nodes. Figure shows an AND - OR graph.

Example of AND-OR Tree



In a problem reduction space, the **nodes** represent problems to be solved or goals to be achieved, and the **edges** represent the decomposition of the problem into subproblems.

Consider the Tower of Hanoi problem:

The Tower of Hanoi, is a mathematical problem which consists of three rods and multiple disks. Initially, all the disks are placed on one rod, one over the other in ascending order of size similar to a cone-shaped tower.

The objective of this problem is to move the stack of disks from the initial rod to another rod with the help of third rod, following these rules:

1. Only one disk can be moved at a time.

2. Each move consists of taking the uppermost disk from one of the rods and sliding it onto another rod, on top of the disks that may already be present on that rod.

3. No disk may be placed on top of a smaller disk.

10.What is alpha beta pruning? How it is different from minmax algorithm?

Alpha-beta pruning gets its name from two parameters.

• They describe bounds on the values that appear anywhere along the path under consideration:

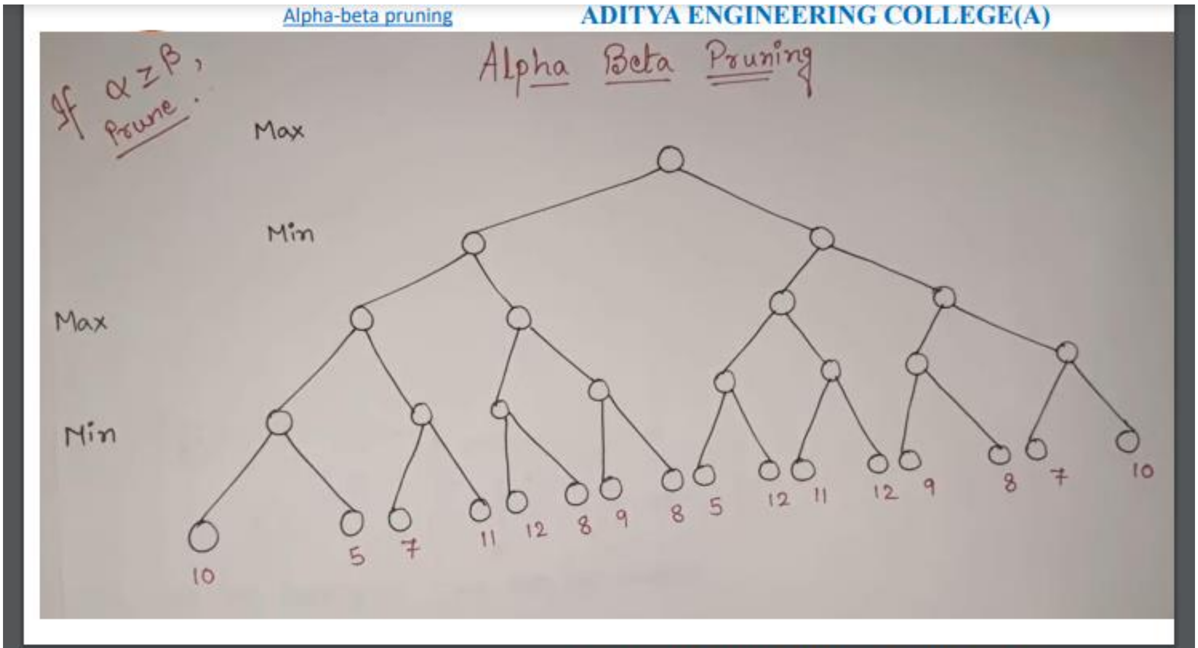
• α = the value of the best (i.e., highest value) choice found so far along the path for MAX

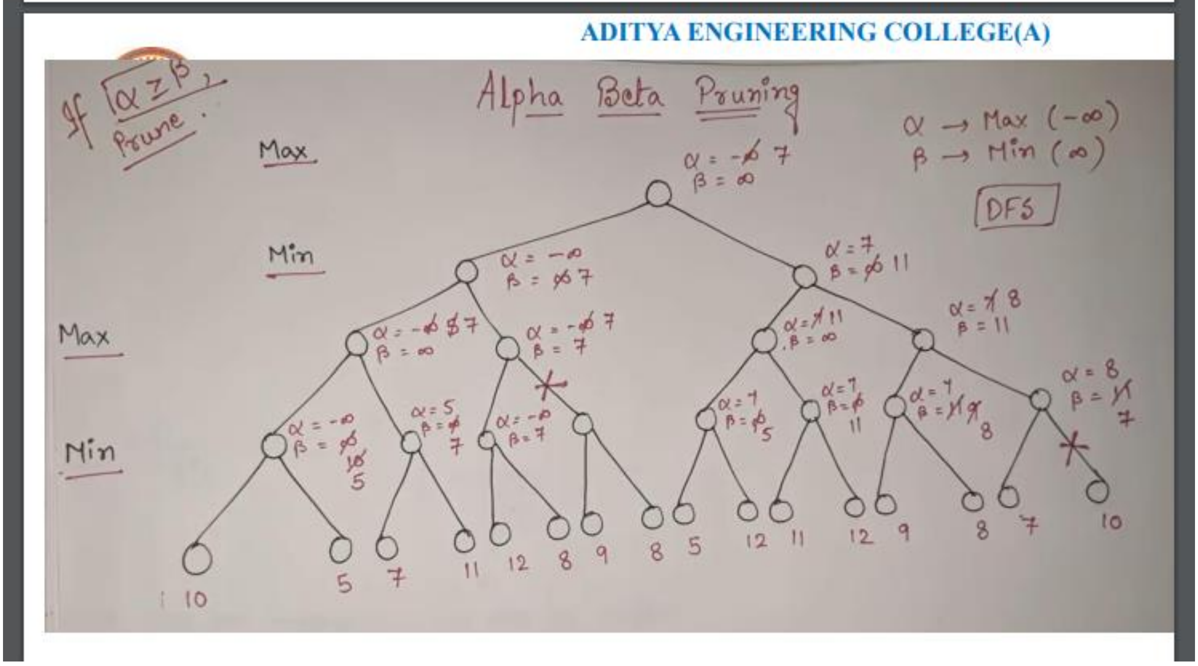
• β = the value of the best (i.e., lowest value) choice found so far along the path for MIN

• Alpha-beta search updates the values of α and β as it goes along

. •It prunes the remaining branches at a node (i.e., terminates the recursive call)

• as soon as the value of the current node is known to be worse than the current α or β value for MAX or MIN, respectively





MINMAX ALGORITHM:

The minimax algorithm computes the minimax decision from the current state.

• It uses a simple recursive computation of the minimax values of each successor state.

• Minimax is a kind of backtracking algorithm that is used in decision making and game theory to find the optimal move for a player, assuming that your opponent also plays optimally.

• The recursion proceeds all the way down to the leaves of the tree.

• Then the minimax values are backed up through the tree as the recursion unwinds.

PROCEEDURE

• Keep on generating the search tree until the limit, say depth d of the tree has been reached from the current position.

• Compute the static values of the leaf nodes in depth d from current position of the game tree using evaluation function.

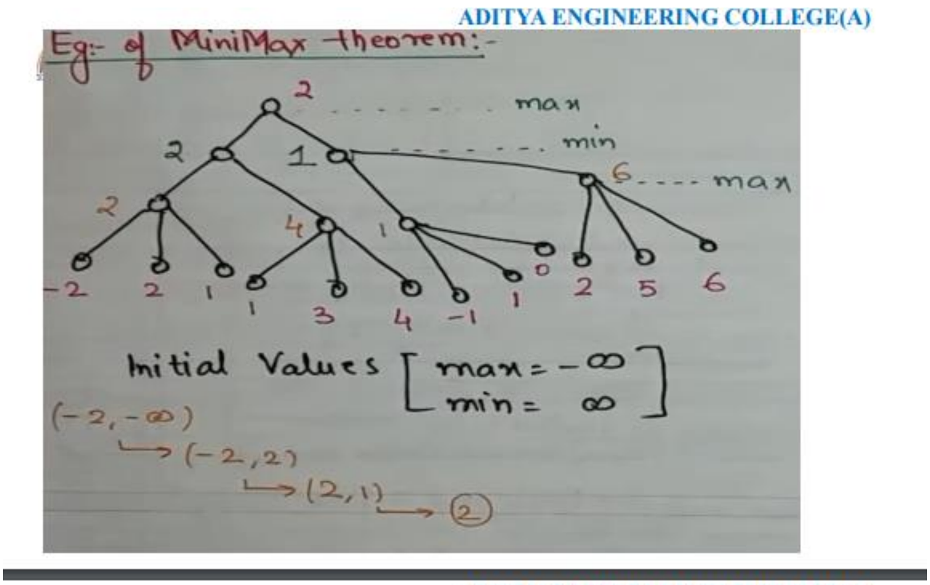
• Propagate the values till the current position on the basis of MINMAX strategy.

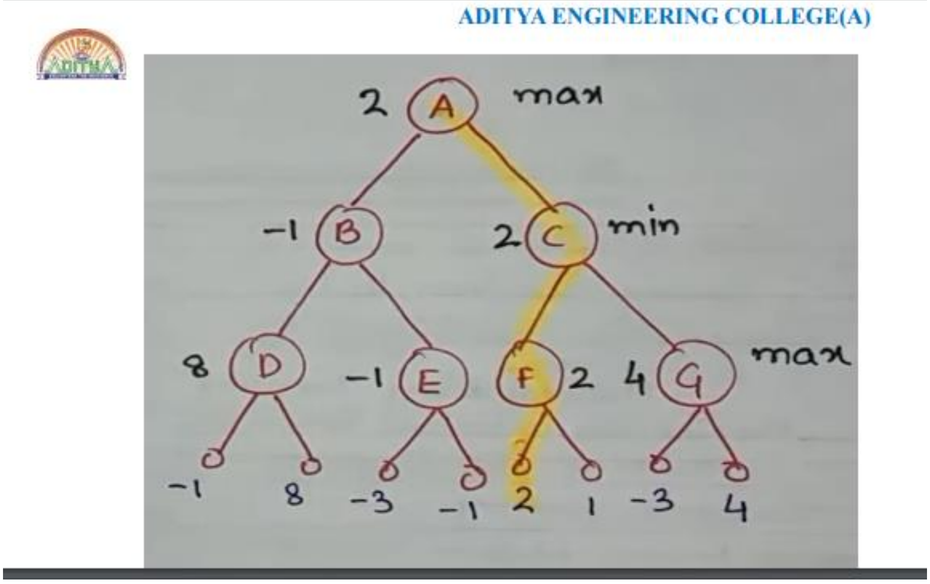
STRATEGY:

• Generate the successors of the current position.

• Apply MINMAX to each of the successors.

• Return the minimum (if the level is minimizing) and maximum (if the level is maximizing)





DIFFERENCE

• It is possible to compute the correct minimax decision without looking at every node in the game tree in alpha-beta pruning.

• Alpha-beta pruning allows to eliminate large parts of the tree from consideration, without influencing the final decision

11.What is propositional logic? Give any two examples?

- Propositional logic (PL) is the simplest form of logic where all the statements are made by propositions.
- A proposition is a declarative statement which is either true or false. It is a technique of knowledge representation in logical and mathematical form.
- Propositional logic is also called Boolean logic as it works on 0 and 1.

Syntax of propositional logic:

- The syntax of propositional logic defines the allowable sentences for the knowledge representation. There are two types of Propositions:
- **Atomic Propositions:**

  Atomic Proposition: Atomic propositions are the simple propositions. It consists of a single proposition symbol. These are the sentences which must be either true or false.
- a) 2+2 is 4, it is an atomic proposition as it is a true fact.
- b) "The Sun is cold" is also a proposition as it is a false fact

  Compound propositions
- Compound propositions are constructed by combining simpler or atomic propositions, using parenthesis and logical connectives.
- Example:
- a) "It is raining today, and street is wet."
- b) “Raj is a doctor, and his clinic is in Mumbai."

Examples:

Raj did not read book

~ read(Raj,book)

Raj watches Amazon or Netflix

Watches(Raj,Amazon) v Watches(Raj,Netflix)

If Raj buys mobile then color is black

Buys(Raj,Mobile) -> Color(Mobile,black)

Raj becomes happy if and only if Raj eats dairy milk.

becomes(Raj,happy) <-> eats(Raj,Dairy milk)

12.What is logic? Define well-formed formula?

- Logic is a study of principles used to
  - distinguish correct from incorrect reasoning.
  - Logical system should possesses properties such as consistency, soundness, and completeness.
  - Consistency implies that none of the theorems of the system should contradict each other.
  - Soundness means that the inference rules shall never allow a false inference from true premises.

1.Well-Formed Formula

- Well-formed formula is defined as:
  - An atom is a well-formed formula.
  - If a is a well-formed formula, then ~a is a well-formed formula.
  - If a and b are well formed formulae, then (a L b), (a V b), (a ® b), (a « b) are also well-formed formulae.

A propositional expression is a well-formed formulaif and only if it can be obtained by using above conditions

Well-Formed Formula Examples

R1-An atom is a well-formed formula.

R2-If a is a WFF, then ~a is a well-formed formula.

R3-If a and b are well formed formulae, then (a L b), (a V b), (a ®b),(a « b) are also well-formed formulae.

WFF	explanation
A	by rule 1
~A	by rule 2, since A is a WFF
~~A	by rule 2 again, since ~A is a WFF
(~A • B)	by rule 3, joining ~A and B
((~A • B) ⊃ ~~C)	by rule 3, joining (~A • B) and ~~C
~((~A • B) ⊃ ~~C)	by rule 2, since ((~A • B) ⊃ ~~C) is a WFF