* What is the fundamental limitation of Bayesian approach?

→ When you need to consider many evidence in favor of a hypothesis, then the size of joint probability that we need to compute grows exponentially as 2^n,

2019

QA.1 a). Give a scientific definition of intelligence. Show how it differs from dictionary definition of intelligence.

The scientific definition of intelligence;

Intelligence is a measure of the success / performance of an entity achieving its objectives/ goals by interaction with its environment. The level of intelligence should be judged by consideri[ng] the difficulty of goals, and the success in achievi[ng] them.

The dictionary definition of intelligence is,

Intelligence is the faculty of acquiring and applying knowledge.

Clearly, the dictionary definition is quite abstract and inefficient to give clear idea of what intelligence means, where as the scientific definition is precise and detailed. With the scientific definition, we can identify if there is presence of intelligence in any entity by observing:

- If the entity can achieve its goal or objectives
- If it can interact with the environment to achieve it goals.

But the dictionary definition does not provide such directions.

(b) Do you think humanoid Sophia and Jarvis possess the features and signs of intelligence? Elaborate you answer by accounting for the behavior of Sophia and Jarvis.

Sophia is a social humanoid robot. The features and signs of intelligence that Sophia posses are depicted below:

1. Sophia can interact with its environment.

Sophia can imitate human gestures and facial expressions, she can answer certain questions and make simple conversations on predefined topics.

2. Sophia can learn from information and experience.

Sophia's AI program can analyze conversations & extracts data that allows it to improve responses in the future.

3. Sophia can make rational decisions and respond quickly and successfully to new situations

When questioned about her potential for abuse, she had a quick rebuttal. "You've been reading to much Elon Musk and watching too many Hollywood movies. Don't worry, if you're nice to me, I'll be nice to you"

Jarvis in an AI system which an can control home system and perform basic tasks such as turning the lights off or on, controlling room temperature etc.
It uses natural language processing and speech recognition to understand the user's voice and the context of the command, in order to perform the task it is asked to do. So it has features and signs of intelligence which are:

- interact with its environment
- make rational decisions.
- understand and infer in ordinary, rational ways.
- understand from experience.
- learn or understand from experience

If it gets something wrong, Zuckerberg corrects it and tells it how it needs to be done, so that the home AI learns and does better next time.

Answer the following about the Mark Zuckerberg.

i) Identify its PAGE (percepts, actions, goals and environment) description.

Percepts	Actions	Goals	Environment.
Spirit's sensor include: i) Panoramic and microscopic cameras. ii) A radio receiver iii) spectrometers for studying rock samples including an alpha ray spectometer, Mössbauer spectrometer and miniature thermal emission spectrometer.	i) Motor-driven wheels for locomotion ii) Robotic arm to bring sensors close to interesting rocks iii) Rock abrasion tool (RAT) capable of efficiently drilling 45 mm holes in hard volcanic rock. iv) a radio transmitter for communication	1) Maximizing the distance or variety of terrain it traverses. 2. Collecting as many samples as possible 3. Finding life. 4. Maximizing lifetime 5. Minimizing power consumption.	The Martian surface

ii) Characterize it's environment as being either accessible or inaccessible, deterministic or nondeterministic, episodic or non episodic, static or dynamic, and discrete or continuous, explain what each of your selected terms mean.

1.. Partiall observable or accessible.

It means the agent's sensor's give access to only some portion of the state of environment needed to choose an action.

2. Non deterministic.

The next state of the ~~martian~~ Martian environment ~~is~~ ~~cannot~~ is not determined by the current state of environment and action of the agent rover.

3. Sequential or non episodic

Subsequent episodes depends on what actions occurred in previous episodes.

4. Dynamic

- The environment ~~also~~ may change while the rover is thinking.

- Time to compute a good strategy matters.

5. Continuous

The number of percepts are not limited.

6. May be a single agent / without rational adversaries

- As there are no adversaries, it don't have to worry about ~~them~~ any other agent interfering its tasks.

iii) What agent architecture is best for the automated taxi driving agent? Justify your answer.

~~3+~~ Combination of 3 types of architecture ~~we~~ might be best for automated taxi driving agent which are as follows:

1. Model-based reflex agent
   - for low level navigation

2. Goal-based agent
   - for safe driving
   - reaching to correct destination

3. Utility based agent

- route planning.
- finding shortest path to destination
- minimizing power consumption
- optimizing the use of resources.

Q.2 How can you formulate a search problem? Give example of data structures that can be used to represent a state space both in explicit and implicit ways?

Search is the process of imagining sequences of operators applied to the initial state, and checking which sequence reaches a goal state.

A search problem can be formulated as below:

S: the set of full states

S_0: Initial state

A: the set of operators

G: the set of final states.

Search problem: Find a sequence of actions which transforms the agent from the initial state to a goal state $a \in G$

State space can be represented either by using graph or tree data structures.
If we use graph, then it is considered as the explicit or direct way of representing state space.
On the other hand, if you use tree, then it is considered implicit.

b) Give the initial state, goal ~~or~~ test, successor function, and cost function for the following:
A 3-foot-tall monkey is in a room where some bananas are suspended from the 8-foot ceiling. He would like to get the bananas. The room contains two stackable, movable, climbable 3-foot high crates.

Initial state: The monkey on the ground ~~and~~ the bananas suspended from the ceiling and the two crates on the ground.

Action/Transition model/Successors:

- Hop on crate
- Hop off crate

- Push crate from one spot to another
- Stack on crate on another
- Walk from one spot to another.
- grab bananas (if standing on crate)

Goal test: Monkey has bananas.

Cost function: Number of actions

From slide

Using only four colors, you have to color a planar map so that no two adjacent regions have the same color.

Initial state: No regions colored

Actions/Successors/Operators:

- Assign a color to an uncolored region.

Transition model: The previously uncolored region has the assigned color

Goal test: All regions colored, and no two adjacent regions have the same color.

Cost function: Number of assignments.

2018 2(b)ii You have three jugs measuring 12 gallons, 8 gallons and 3 gallons and a water faucet. You can fill the jugs up or empty them out from one to another or onto the ground. You need to measure out exactly one gallon.

Initial state: jugs have values [0,0,0].

Actions/ Transition model /Successors:

- Given values [x, y, z], generate [12, y, z], [x, 8, z], [x, y, 3]
- (by filling): [0, y, z], [x, 0, z], [x, y, 0]
- (by emptying): or for any two jugs with current values x any y, pour y into x's this changes the jug with x to the minimum of (x+y) and the capacity of the jug, and decrements the jug with y by the amount gained by the first jug.

Cost function: Number of actions.

An. 3 a) Write down steepest-ascent hill climbing algorithm. What problems the hill climbing search process may encounter? How can the problems be dealt with?

The steepest ascent algorithm is a variation of simple hill climbing algorithm. This algorithm examines all the neighboring nodes of the current state and selects one neighbor node which is closest to the goal state. This algorithm consumes more time as it searches for multiple neighbors.

Algorithm:

Step 1: Evaluate the initial state, if it is goal state then return success and stop, else make current state as initial state.

Step 2: Loop until a solution is found or the current state does not change.

a. let SUCC be state such that any successor of the current state will be better than it.

b. For each operator that applies to the current state:

(i) Apply the new operator and generate a new state.

(ii) Evaluate the new state

(iii) If it is goal state, then return it and quit else compare it to the SUCC

a) If it is better than SUCC, then set new state as SUCC

(iv) If the SUCC is better than the current state then set current state to SUCC

Step 3: Exit.

Drawbacks of Hill-Climbing

1. Local Maximum is a state that is better than all its neighbors but is not better than some other states farther away.

2. Plateau is a flat area of the search space in which a whole set of neighbouring states have the same value.

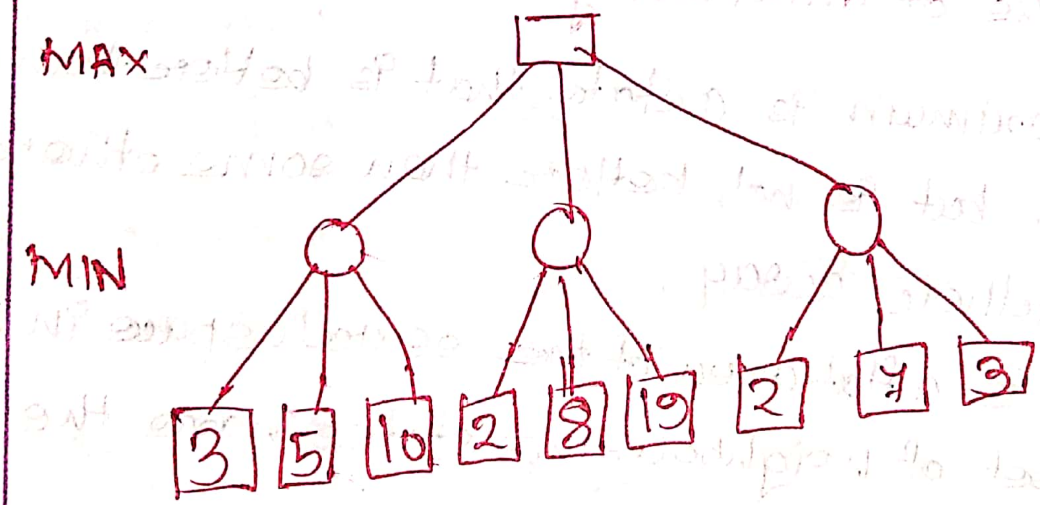
3. Ridges is a special kind of local maximum. It is an area of the search space that is higher than surrounding areas and that itself has a slope which one would like to climb.

In each case the algorithm reaches a point at which no progress is being made.

Remedy

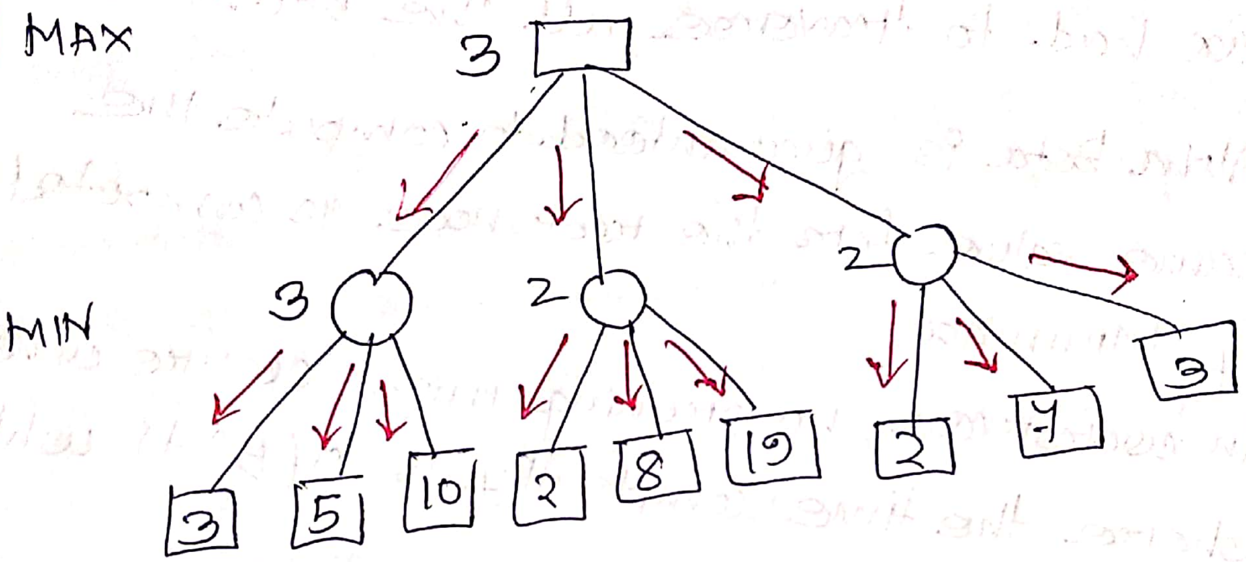
- Backtrack to some earlier node and try going in a different direction.
- Start again from a different start point.

c) Explain how alpha-beta pruning algorithm addresses the limitations of minimax algorithm of searching game trees. Elaborate your answer by taking account of following tree.

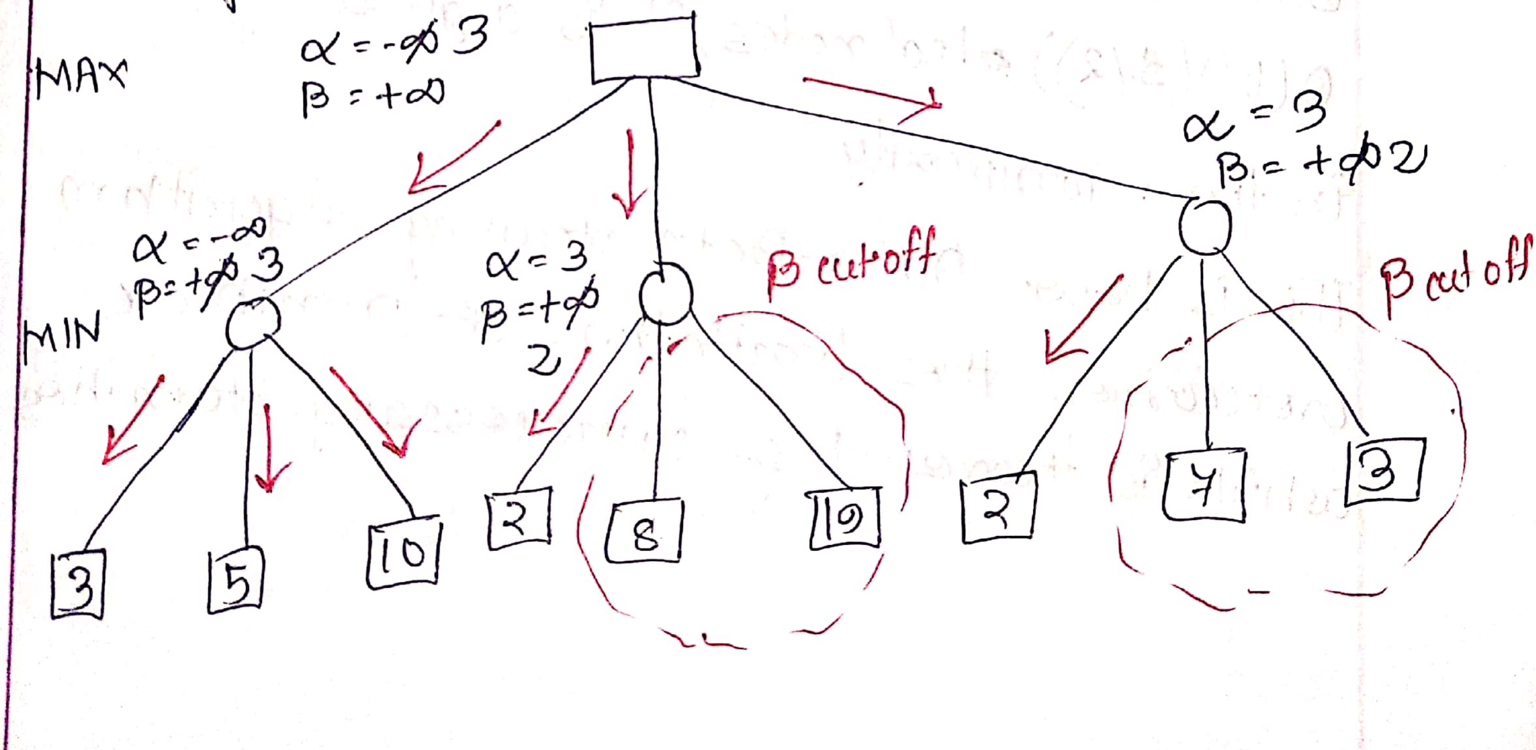


The alpha beta algorithm does the same calculation as minmax but is more efficient because it prunes (allows us to ignore portions of the search tree that make no difference to the final choice) away branches of the search tree that it can prove are irrelevant to the final outcome.

Applying minimax algorithm,



Applying alpha beta pruning algorithm,



After applying both the algorithm to the given example, we can see that we could prune away 4 branches in alpha beta pruning algorithm where as in MiniMax, we had to traverse all the branches.

Alpha-Beta is guaranteed to compute the same value for the root node as computed by Minimax.
In worst case, no pruning might occure where the time complexity is O(b^d) which where the time complexity is same as MiniMax.
But in best case, it will need examine only O(b^(d/2)) leaf nodes, it is a great improvement in time complexity.
This is how Alpha-Beta pruning algorithm overcome the limitation of minimax which is traversing unnecessary branches.

(d) What is CSP? Do you consider the following problem to be an example of CSP?
Coloring a planar map using only four colors in such a way that no two adjacent regions have the same color?
If so identify the variable, domains and constraints associated with this problem.

Constraint Satisfaction Problems are defined by a set of variables X_i, each with a domain D_i of possible values and a set of constraints C.
The aim is to find an assignment of the variables X_i from the domains D_i in such a way that none of the constraints C are violated.
The given problem is an example of CSP. In the given problem:-

Variables: The set of regions in the map.
Domains D_i = { Color1, Color2, Color3, Color4 }
Constraints: ~~two~~ no two adjacent regions can have the same color that is adjacent regions must have different colors.

(b) Differentiate between DFID and IDA * algorithms. Demonstrate how DFID incorporates the advantages of breadth-first as well as the depth first search algorithm with an example.

DFID is a combination of both BFS and DFS. The algorithm for DFID is as follows:

Initialize c=0
until solution found do
DFS with depth bound c
c=c+1

IDA* is similar to IDDF (DFID) except now at each iteration the DF search is not bound by the current depth_limit but by the current f_limit.

- At each iteration, all nodes with f(n)<=f_limit will be expanded (in DF Fashion).
- If no solution is found at the end of an iteration, increase f_limit and start the next iteration. The algorithm is such:
* Initialization: f_limit:=h(s)
Increment at the end of each (unsuccessful) iteration,

f_limit := max {f(n) | n is a cut-off node}

DFID is complete (iteratively generate all nodes up to depth d). IDA* is also complete.

DFID is optimal/admissible if all operators have the same cost. IDA* is admissible if h is admissible.

DFID has linear space complexity that is O(bd)

IDA* has space usage proportional to depth of solution which means it also has linear space complexity.

Iterative Deepening Depth-first search:

→ combination of both DFS and BFS
→ Best depth limit is found out by gradually increasing limit.

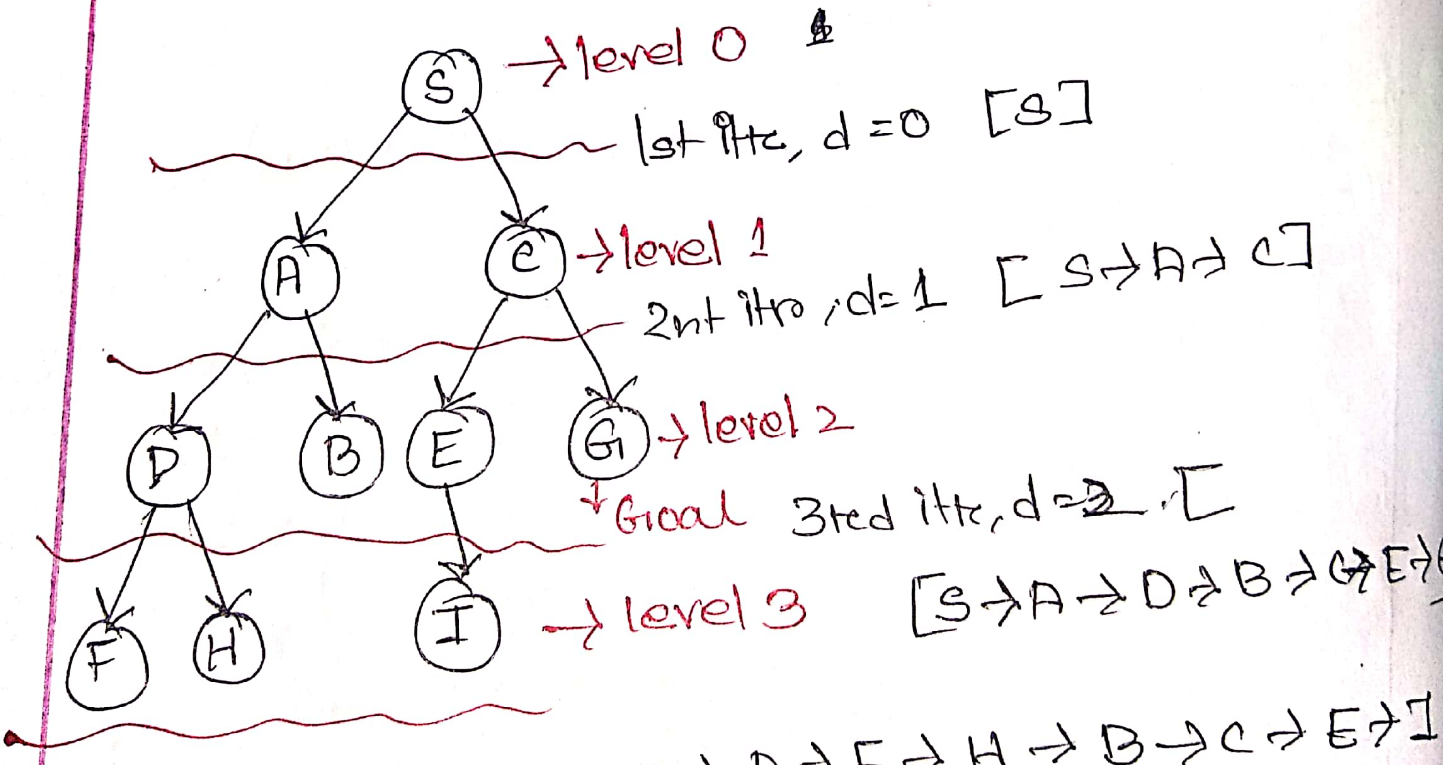
Initially d = 0, it every iteration it increases by 1.

Advantage:

- Incorporates benefits of both BFS and DFS
- Faster search, less memory.

Disadvantage:

- Repeats the work/process.



4th itr, d = 3, [S → A → D → F → H → B → C → E → I → G]

A.4 a) Briefly explain why FOL is considered generalization of PL.

In general, propositional logic can deal with only a finite number of propositions.
For example, if there are only three dogs Tommy, Jimmy and Laika, then

- T: Tommy is faithful
- J: Jimmy is faithful
- L: Laika is faithful
- All dogs are faithful $T \wedge J \wedge L \ldots$

If there are infinite number of dogs, we cannot represent it in proposition logic.

FOL is the generalization of PL that allows us to express and infer argument in infinite models like

- All men are mortal
- Some birds cannot fly

It expresses such infinite models using quantifiers
Two basic quantifiers ~~at~~ in FOL are

- $\forall$ "for all" ← Universal quantifier
- $\exists$ "there exists" ← Existential quantifiers

We can represent the previous examples using quantifiers as follows.

$\forall x(dog(x) \rightarrow faithful(x))$

$\forall x(man(x) \rightarrow mortal(x))$

$\exists x(bird(x) \rightarrow \neg Fly(x))$

[A.U] b) a) First translate the sentences above into FOL sentences.

i) If a perfect square is divisible by prime P, then it is also divisible by square of P

$\forall x,y\,(perfectSquare(x) \wedge prime(y) \wedge divisible(x,y) \rightarrow divisible(x, square(y)))$ --- ①

ii) Every perfect square is divisible by some prime

$\forall x \exists y\,(perfectSquare(x) \wedge prime(y) \wedge divisible(x,y))$ --- Ⓐ

iii. 36 is a perfect square

$perfectSquare(36)$ --- Ⓑ

iv. Does there exist a prime q such that the square of q divide 36?

$\neg\exists y\,(\text{prime}(y) \wedge \text{divisible}(36, \cancel{y}\ \text{square}(y)))$ (negating the conclusion)

b. Convert to clause Form

$\boxed{\text{Step 1}}$: Eliminating →

1. $\forall x,y\,(\neg(\text{perfectSquare}(x) \wedge \text{prime}(y) \wedge \text{divisible}(x,y)) \vee \text{divisible}(x, \text{square}(y)))$

2. $\forall x \exists y\,(\text{perfectSquare}(x) \wedge \text{prime}(y) \wedge \text{divisible}(x,y))$
3. $\text{perfectSquare}(36)$
4. $\neg\exists y\,(\text{prime}(y) \wedge \text{divisible}(36, \cancel{y}\ \text{square}(y)))$

$\boxed{\text{Step 2}}$: Reducing the scope of negation:

1. $\forall x,y\,((\neg\text{perfectSquare}(x) \vee \neg\text{prime}(y) \vee \neg\text{divisible}(x,y)) \vee \text{divisible}(x, \text{square}(y)))$

2. $\forall x \exists y\,(\text{perfectSquare}(x) \wedge \text{prime}(y) \wedge \text{divisible}(x,y))$
3. $\text{perfectSquare}(36)$
4. $\cancel{\neg}\ \forall y\,(\neg\text{prime}(y) \vee \neg\text{divisible}(36, \cancel{y}\ \text{square}(y)))$

Step 3: Standardize variables apart

1. $\forall x \forall y$ ($\neg$ perfectSquare(x) $\vee$ $\neg$ prime(y) $\vee$
 $\neg$ divisible (x, y) $\vee$ divisible (x, square(y)))

2. $\forall w \exists z$ (perfectSquare (w) $\wedge$ prime(z)
 $\wedge$ divisible (w, z))

3. perfectSquare (36)
4. $\forall u$ ($\neg$ prime (u) $\vee$ $\neg$ divisible (36, ~~u))~~ square(u))

Step 4: Move all quantifiers to the left.

No change

Step 5: Eliminate $\exists$ (skolemization)

1. same as before
2. $\forall w$ (perfectSquare(w) $\wedge$ prime (name1)
 $\wedge$ divisible (w, name1))

[name 1 is a skolemization constant]

3. same as before
4. "

Step 6: Drop all $\forall$

1. $\neg$perfectSquare(x) $\vee$ $\neg$prime(y) $\vee$ $\neg$divisible (x,y) $\vee$ divisible (x, square(y))

2. perfectSquare(w) $\wedge$ prime(name1) $\wedge$ divisible (w, name1)

3. perfectSquare(36)
   divisible (36, ~~u~~ square(u))

4. $\neg$ prime (u) $\vee$ $\neg$divisible

Step 7: convert to conjunct of disjunct form
No change

Step 8: make each conjunct a separate clause

1. $\neg$perfectSquare(x) $\vee$ $\neg$ prime(y) $\vee$ $\neg$ divisible(x,y) $\vee$ divisible (x, square(y))

2. perfectSquare(w)

3. prime(name1)

4. divisible (w, name 1)

5. perfectSquare(36)
   divisible (36, ~~u~~ square(u))

6. $\neg$prime (u) $\vee$ $\neg$divisible

Step 9:

1, 2, 3, 5, 6 $\rightarrow$ same as before

4. divisible (z, v)

So the resolution with unification,

6. ¬prime(v) ∨ ¬divisible(36, ~~u~~ square(v))   3. prime(name1)

$\boxed{\text{name1/4}}$

¬divisible(36, ~~u~~ square(u))   1. ¬perfectSquare(x) ∨ ¬prime(y) ∨ ¬divisible(x, y) ∨ divisible(x, square(y))

$\boxed{\text{36/x, u/y}}$

¬perfectSquare(x) ∨ ¬prime(y) ∨ ¬divisible(x, y)   2. perfectSquare(ω)

$\boxed{\text{x/ω}}$

¬prime(y) ∨ ¬divisible(ω, y)   3. prime(name1)

$\boxed{\text{y/name1}}$

~~¬prime~~ ¬divisible(ω, name1)   4. divisible(ω, name1)

null clause

(c) Determine whether the sentence $(P\wedge Q)\vee(P\vee\neg Q)\vee P$ is satisfiable, contradictory and valid

The truth table for the sentence:

P	Q	$\neg Q$	$P\wedge Q$	$P\vee\neg Q$	$(P\wedge Q)\vee(P\vee\neg Q)\vee P$
T	T	F	T	T	T
T	F	T	F	T	T
F	T	F	F	F	F
F	F	T	F	T	T

From the truth table, we see that the sentence is satisfiable.

2012

B.1 a) What is planning? Write down the factors considered in formulating a planning problem?

Planning is the task of coming up with a sequence of actions that will achieve a goal. In the planning problem the task of the agent is to synthesis goal directed behavior.

The factors ~~need~~ considered in formulating a planning problem are

- A way to describe the world
- An initial state of the world
- A goal description
- A set of possible actions to change the world.

For example, a planning problem formulation can look like,

Goal: have a birthday party

Current situation: agent at home, has flour, does not have butter, doesn't have sugar

To do: (sequence of actions):

- invite friends
- buy butter
- buy sugar
- buy balloons
- decorate house
- bake cake.

6 What is knowledge base? Discuss the architecture of a knowledge base agent. Distinguish between knowledge-base agent and problem-solving agent in one point.

Knowledge base is set of rules and facts.

where

Rules: Assertion given in implication form

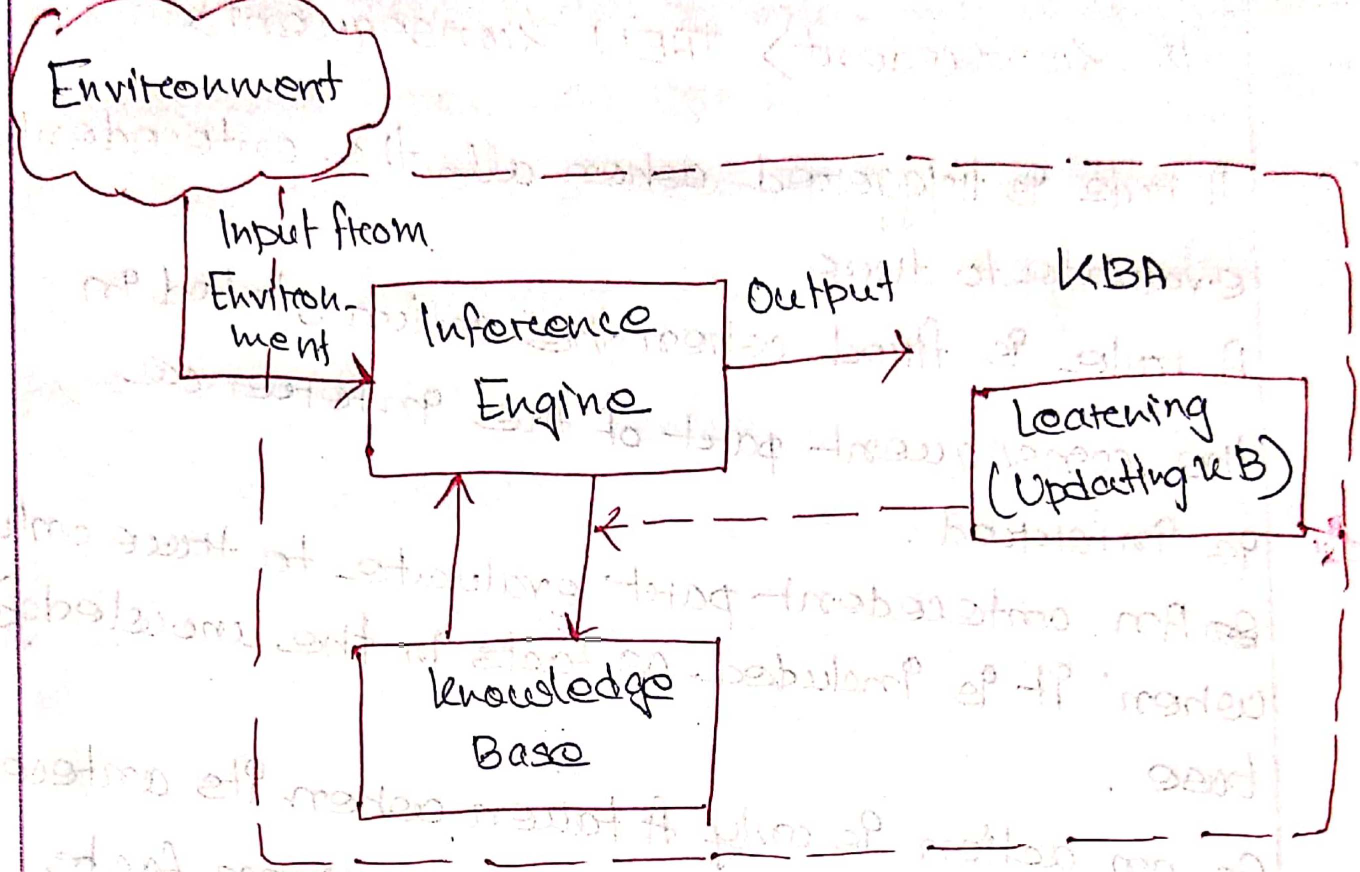
Facts: Assertions that represent domain specific knowledge.

Knowledge-based agents are those agents who have the capability of maintaining an internal state of knowledge, reason over that knowledge update their knowledge after observations and take actions.

Problem-solving agents are able to plan ahead - to consider the consequence of actions - before acting.

Knowledge base agents can select actions ~~on~~ based on explicit, logical representation of the current state and effects of actions. This allows the agent to succeed in complex, inaccessible environment that are too difficult for a problem solving agent.

The architecture of knowledge-based agent.



B.2 (a) What is rule-based system (RBS)? Briefly describe the way in which if then rules can be used as a basis for knowledge representation and reasoning in RBS.

A system whose KB is represented as a set of rules and facts is called a Rule Based system (RBS).

A rule-based system consists of a collection of IF-THEN rules, a collection of facts and some interpreter controlling the application of the rules, given the facts.

Rules are represented in the following form.

IF <antecedent> THEN <consequent>

A rule is triggered when all the antecedents evaluate to true.

A rule is fired when the action stated in the consequent part of the inference is inferred.

An antecedent part evaluate to true only when it is included as facts in the knowledge base.

So an action is only taken when its antecedent part evaluates to true by the given facts.

It means every action is logically entailed from the given facts.

So we can say IF-THEN rules serves as the basis for knowledge representation and reasoning in RBS.

6) What is backtracking? Show how good conflict resolution strategy can be used to reduce backtracking.

If the Inference Machine (IM) reaches a dead end that is no new rule is enabled and the goal is not met, then we need to backtrack to the earliest node. This process is called backtracking.

Good conflict resolution strategy reduces backtracking.

(c) Given rules,

R_1: If A and C then E

R_2: If C and D then F

R_3: If B and E then F

R_4: If B then C

R_5: If F then G.

Initial Facts

A
B

Step 1: Triggering R_4 as B is true in facts.

Facts

A
B
C

Step 2: As A and C are true in facts, triggering R_1,

Facts

A
B
C
E

Step 3:

As B and E are true in facts, triggering R_3

Facts

A
B
C
E
F

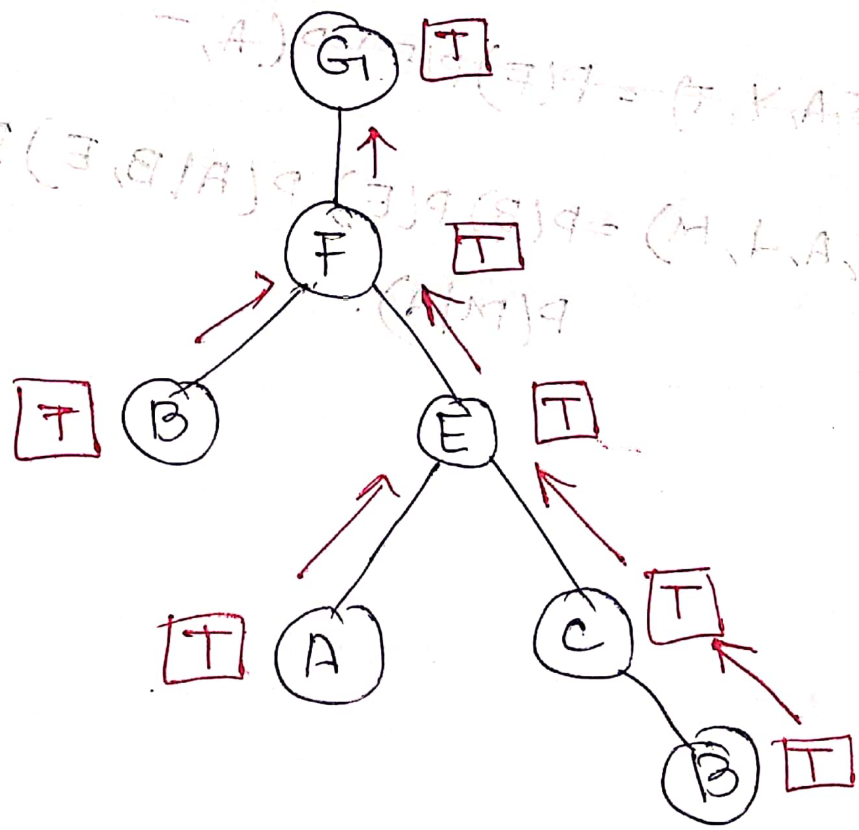
Step 4: As F is true in facts, triggering R_5

Now Goal G is in facts hence goal (G) is proven by forward chaining.

Facts

A
B
C
E
F
G

Using backward chaining,



By propag~~ing~~ating the truth by backtracking, the goal (G) is also proven by backward chaining.

B.3 (b) i) Express the joint distribution $P(B, E, A, J, M$ in terms of conditional probabilities (and independencies) expressed in the Bayesian network above.

$P(B, E, A, J,$

Expressing the joint distribution ~~P(F, E, A, Y,~~ in terms of conditional probability (indepe expressed in Bayesian network:

$\cancel{P(F, E, A, Y, T) = P(F) P(E) P(A,} -$

$$P(B, E, A, J, M) = P(B)\, P(E)\, P(A \mid B, E)\, P(J \mid A)\, P(M \mid A).$$

ii) Determine the probability of an event that the alarm has sounded but neither a burglary nor an earthquake has occurred and both Mary and John call.

$= P(A \wedge \neg E \wedge \neg B \wedge J \wedge M)$

$= P(A \mid \neg B, \neg E) \cdot P(J \mid A) \, P(M \mid A) \, P(\neg E) \, P(\neg B)$

$= .001 \times .90 \times .70 \times (1 - .002) \times (1 - .001)$

$= 0.000628$

iii) Determine the probability of the event that the alarm has sounded and burglary has occurred, an earthquake has not occurred and both Mary and John call.

$= P(A \wedge B \wedge \neg E \wedge J \wedge M)$

$= P(A \mid B, \neg E) \, P(J \mid A) \, P(M \mid A) \, P(B) \, P(\neg E)$

$= .94 \times .90 \times .70 \times .001 \times (1 - .002)$

$= 0.000591$

Q.4 a) What is learning? Describe the inductive learning method with necessary example.

Learning is essential for unknown environments i.e when the agent lacks omniscience. Learning is useful as a system construction method i.e expose the agent to reality rather than trying to write it down. Learning modifies the agent's decision mechanisms to improve performance. Learning in ANN is to reduce the difference between the produced output and the desired output.

Inductive learning

It is also called as deterministic supervised learning. It is a process by which A.I systems attempt to use a generalized rule to carry out observations.

In this first input x is given to a function f and the output is f(x). Then we give different set of inputs (raw inputs) to the same

function f and verify the output f(x).
By using the outputs, we generate or learn the rules.
Inductive learning, also known as discovery learning is a process where the learner discovers rules by observing examples. Here system tries to make a general rule from a set of observed instances.

Example:
Mango → f(Mango) → sweet (e1)
Banana → f(Banana) → sweet (e1)
- - - - - - - - - - - -
Fruits → f(Fruits) → sweet (general rule)

b) What is artificial neural network? Compare artificial and biological networks. What aspects of biological network are not mimicked by artificial ones? What aspects are similar?

ANN is a model that emulates a biological neural network. ANN are hardware or software network, simulation of the parallel processes that involve processing elements (artificial neurons) interconnect in a netwoork architecture. ANN plays dual role

- to improve the computer capabilities
- to test various hypothesis about information processing in the brain.

Comparision between ANN and BNN:

Artificial Neural Network	Biological Neural Network
1. Processing speed is fast compared to BNN	1. They are slow in processing information.
2. Processes operate in sequential mode.	2. The process can operate in massive parallel mode.
3. If any information gets corrupted in the memory it cannot be retrieved	3. Information is distributed into the network throughout into subnodes, even if it gets corrupted, it can be retrieved.
4. The activities are continuously monitored by a control unit.	4. There is no control unit to monitor the information being processed into the network.

ANN imitates how BNN works. The impulse received by dendrites in BNN as the input layer in ANN. The impulse passes through the axons in BNN as the ~~input layer in ANN.~~ process from input layers to the output layer passes through hidden layer in the ANN. Finally the transmission from the synaptic terminals to the other dendrites in BNN as the output layer in ANN.

But this model does no mimic neither the creation nor the destruction of connections (dendrites or axons) between neurons and ignores signal timing. Also ANN has much less number of "neurons" compared to BNN.

c) What are supervised learning, unsupervised learning and reinforcement learning?

Supervised learning: Any situation in which both the inputs and outputs of a component can be perceived is called supervised learning.

Supervised learning is the task of inferring a function from labeled training data. A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples.

Examples: delta rule, Backpropagation.

Unsupervised learning

Learning when there is no hint at all about the correct outputs is called unsupervised learning.

Unsupervised learning is the task of inferring a function to describe hidden structure from unlabelled data. Since the examples given to the learner are unlabeled, there is no error or reward signal to evaluate a potential solution - this distinguishes unsupervised learning from supervised learning and reinforcement learning

Examples: Principal component analysis

competitive learning

Reinforcement learning:

In learning the condition action component, the agent receives some evaluation of its action but is not told the correct action. This is called reinforcement learning.

Hence the agent learns from a series of reinforcements, rewards or punishments. It is up to the agent to decide which of the actions prior to the reinforcement were most responsible for it.

d) Suppose you are given two jug: a 4 gallon one and a 3 gallon one. Neither have any measuring markers on it. There is a pump that can be used to fill the jugs with water. How can you exactly get 2 gallons of water into the 4 gallon jug.

- State (x, y)

  $x = 0, 1, 2, 3,$ or 4 $y = 0, 1, 2, 3$

- Start state: $(0, 0)$
- Goal state: $(2, n)$ for any n
- Attempting to end up in a goal state:

1. $(x,y) \to (4,y)$

   if $x < 4$

2. $(x,y) \to (x,3)$

   if $y < 3$

3. $(x,y) \to (x-d,y)$

   if $x > 0$

4. $(x,y) \to (x,y-d)$

   if $y > 0$

5. $(x,y) \to (0,y)$

   if $x > 0$

6. $(x,y) \to (x,0)$

   if $y > 0$

7. $(x,y) \to (4, y-(4-x))$

   if $x+y \geq 4, y > 0$

8. $(x,y) \to (x-(3-y), 3)$

   if $x+y \geq 3, x > 0$

9. $(x,y) \to (x+y, 0)$

   if $x+y \leq 4, y > 0$

10. $(x,y) \to (0, x+y)$

    if $x+y \leq 3, x > 0$

11. $(0,2) \to (2,0)$

12. $(2,y) \to (0,y)$

1. Current state (0,0)
2. Loop until reaching the goal state (2,0)
   - Apply a rule whose left side ~~m~~ matches the current state
   - Set the new ~~state~~ current state to be the resulting state.

(0,0)
2 → (0,3)
~~9~~ → (3,0)
2 → (3,3)
7 → (4,2)
5 → (0,2)
11 → (2,0)

e) Write down A* search algorithm.

A* search algorithm is best known form of best-first search. It avoids expanding paths that are already expensive.

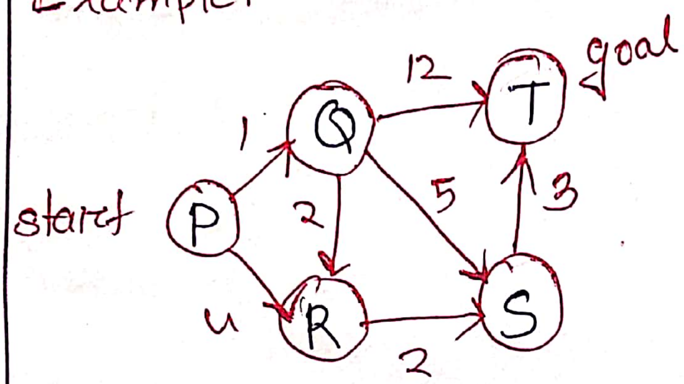
Evaluation function: $f(n) = g(n) + h(n)$

where $g(n)$ = the cost to reach the node n from the start node

$h(n)$ = estimated cost to get from the node n to the goal

$f(n)$ = estimated total cost of path through n to goal.

Example:



P	7
Q	6
R	2
S	1
T	0

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

$P \to 0 + 7 = 7$

$P \to Q = 1 + 6 = 7$

$P \to R = 4 + 2 = 6$

$P \to R \to S = 4 + 2 + 1 = 7$

$P \to Q \to R = 1 + 2 + 2 = 5$

$P \to Q \to S = 1 + 5 + 1 = \cancel{6}\,7$

$P \to Q \to T = 1 + 12 + 0 = 13$

~~P→Q=7~~

~~P→R=6~~

~~P→R→S=7~~

~~P→Q→R=5~~

~~P→Q→S=7~~

$\boxed{P \to Q \to T = 13}$

~~P→Q→R→S=6~~

$\boxed{P \to Q \to R \to S \to T = 8}$

~~P→R→S→T=9~~

$P \to Q \to R \to S = 1+2+2+1 = 6$

$P \to Q \to R \to S \to T = 1+2+2+3+0 = 8$

$P \to R \to S \to T = 4+2+3+0 = 9$

$P \to Q \to S \to T = 1+5+3+0 = 9$

$\boxed{P \to Q \to R \to S \to T}$ ← Optimal path.

Algoroithm of A * search.

Step1: Place the starting node in the OPEN list.

Step2: Check if the OPEN list is empty or not, if the list is empty then return failure and stop.

Step3: Select the node from the OPEN list which has the smallest value of evaluation function (g+h), if node n is goal node then return success and stop, otherwise,

- Expand node n and generate all of its successor and put n into the closed list.
- For each successor n', check whether n' is already in the OPEN or CLOSED list, if not then compute evaluation function for n' and place into open list.

- else if n' is already in OPEN and CLOSED list, then it should be attached to the back pointer which reflects the lowest g(n') value.

Step 4: Return to step 2.

f) What is uncertainty? What are different sources of uncertainity?

In a proposition logic A → B, if A is true then B is true but in some situation we are not sure whether A is true or not, then we cannot express this situation, such a situation is called uncertainty.

~~example~~ For example, if we consider the proposition below:

If doorbell rings, the Karim wakes up.

Here, this proposition might not always be true. Karim may wake up even if the doorbell doesn't ring. On the other hand, even if the doorbell rings, Karim might not wake up.

Sources of uncertainty.

1. Conflicting Information
- Experts often provide conflicting information.

2. Propagation of uncertainities
- In absence of interdependencies, propagation of uncertain knowledge increase the uncertainty of the conclusion

3. Data
- missing data, unreliable, ambiguous, Imprecise representation, inconsistent, subjective, derived from defaults.

4. Knowledge representation
- Restricted model of the real system
- Limited expressiveness of the representation mechanism.

5. Inference process
- Deductive
  - The derived result is formally correct but wrong in the real system
- Inductive
  - New conclusion are not well-founded
  - unsound reasoning methods