KESHAV KRISHNA
2401831530115

PAGE No
DATE : / 202

Assignment - 1

Q1

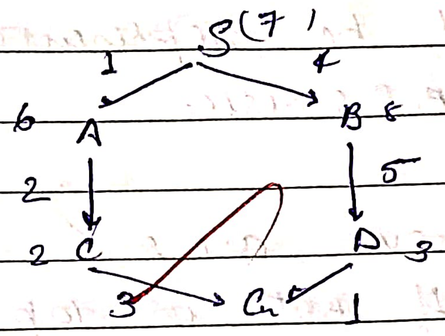
$S \to A(1)$, $S \to B(4)$, $h(S) = 7$

$B \to D(5)$ $h(A) = 6$

$A \to C(2)$ $h(B) = 5$

$C \to G(3)$ $h(C) = 2$

$D \to G(1)$ $h(D) = 3$, $h(G) = 0$



$S \to A$

$f(A) = g(A) + h(A)$
$= 1 + 6 = 7$

$S \to B$

$f(B) = 4 + 5$
$= 9$

$S \to A \to C \Rightarrow f(C) = 5$

$S \to A \to C \to G$

$f(G) = 3 + 2 + 1 = 6$

Path = $S \to A \to C \to G$

Q2

Initial state

1 2 3
4 5 6
7 8 -

Goal State

1 2 3
4 5 6
7 - 8

i) Each states are represented in BFS of 3x3 grid as :-
A 1D array of length 9 where

each index corresponds to a tile position, the empty space is represented by a special symbol ex :- 0 as '_'

initial state can be represented as [1, 2, 3, 4, 5, 6, 7, 8, 0] each state will be represented as node in the BFS Tree.

ii) Valid moves are :-
→ UP (if not in top row)
→ Down (if not in bottom row)
→ Left (if not in left most col.)
→ Right (if not in right most col.)

Each states move produces a new state

iii) a) start at initial state in queue.
(b) At each step :-
i) remove front ~~~~ state.
ii) Generate all valid child states
iii) Add unseen states to queue.
(c) Maintain a visited ~~~~ set to avoid revisiting
(d) BFS explores level by level & be ensures fewest moves.

(iv)

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & \_ \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & \_ & 8 \end{bmatrix} \qquad \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & \_ \\ 7 & 8 & 6 \end{bmatrix}$$

↓

Goal

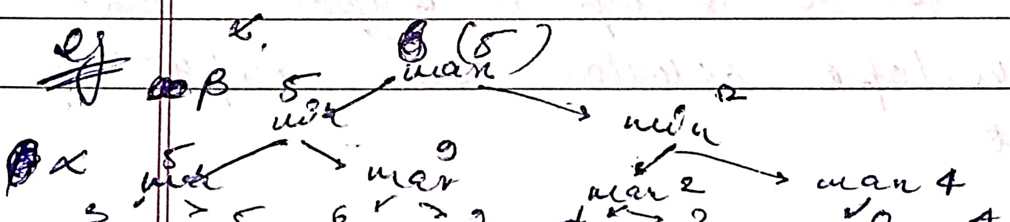
- BFS searches by level by level order.
- Here only one moves is needed. we can move either UP or left i.e. by going left we achieve goal.

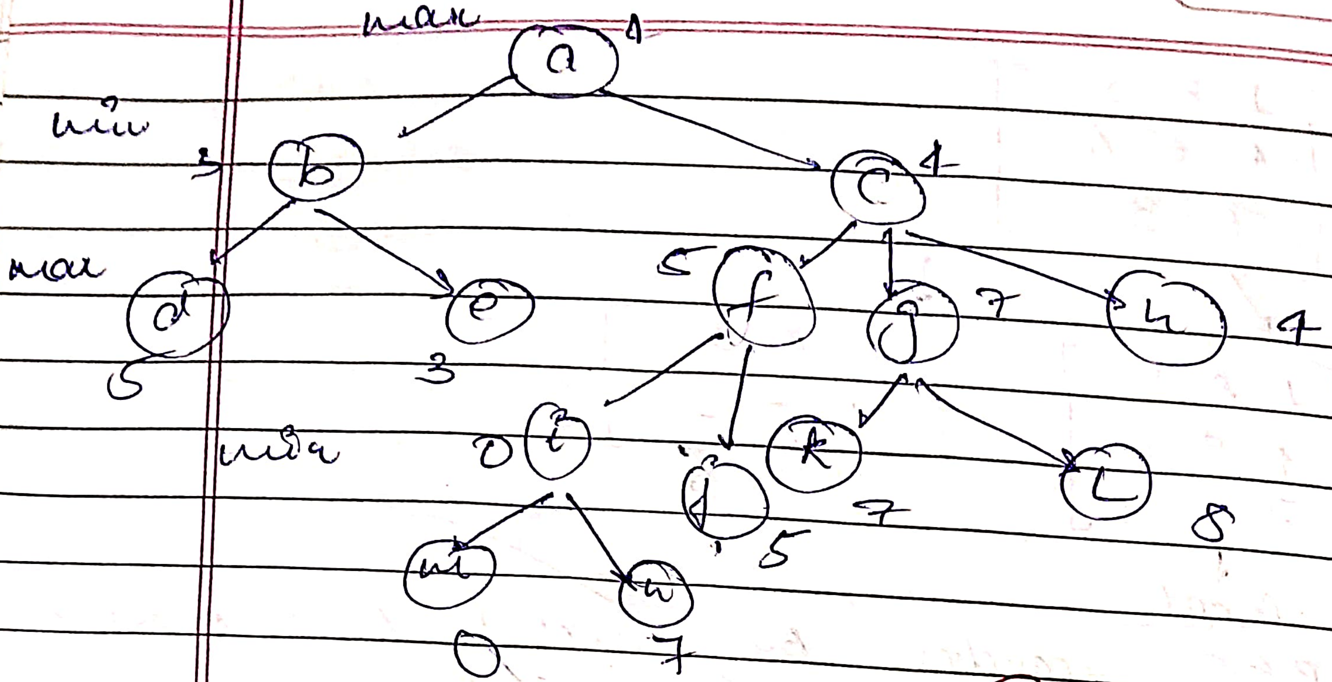
Q3

Min Max :- used in 2 player games where Max tries to maximize the score, Min tries to minimize the score, without pruning it evaluates all leaf nodes.

- Alpha - beta pruning
  Improves MINMAX by pruning unnecessary branches. Alpha (α) → Best value for Max.
  Beta (β) :- Best value for min.
  If $\alpha \geq \beta$ → prune (stop exploring)

eg





During Traversal :-

i) If a min node a value ≤ current alp remaining children pruned.

ii) If a max node finds a value ≥ current Beta, remaining children are pruned.

iii) Reduces numbers of nodes evaluated.

iv) Improves speed up of decision making without changing the final output.

Q4

Supervised Learning :-

Involves Training a model using labeled dataset.

The Goal is to develop a mapping that can develope relation b/w input & output

input	output

→ Learning Algo ← Model

New data → Model → Output

[Naive, Bayes,
Linear Regression
KNN, SVM]

Examples :- email spam detection.

* **Unsupervised Learning**

Learning based only on input database without labelled outputs. Finds patterns or groups in unlabelled data like cluster or dimensionality reduction. No supervisor is present.

Input (Raw data) → Unknown output [No training Data] → Interpretation & Algo → processing → output

Example :- Customer Segmentation in market

* **Reinforcement Learning**

Involves training an agent to interact with an environment to learn from the feedback it receives
eg:- self driving cars, chess, etc.

Q5

→ Suppose we have a dataset of students

Students	Maths	Physics	chemistry	english
S_1	80	75	72	65
S_2	90	88	85	70
S_3	78	70	68	60
S_4	85	82	80	72
S_5	92	90	88	74

feature Scaling
→ Since PCA is sensitive to the scale of data, we standardise each feature

$$z = \frac{x - \mu}{\sigma}$$

Now, every features has mean = 0, variance

* Covariance Matrix
we compute the covariance matrix of the standardized dataset.
i) it tell us how features vary with each other.
ii) The matrix will be 4×4 since we have 4 features.

* Eigen value & Eigen vectors.
Eigen values → amount of variance explained by PC
eigen vectors → direction of new axis (PC

PAGE No
DATE : / 202

Suppose we get :-

PC1 = 70 % variance
PC2 = 20 % variance
PC3 = 4 % variance
PC4 = 3 % variance

★ Dimension Reduction
→ We select the top 2 principal components (PC1 + PC2)
→ Together they retain about the variance
→ So we reduce dataset fro 4D→ 2D without loosing much information.

★ Interpretation
→ If (PC1 + PC2) = 90 % it means our 2D reduced dataset preserves. 90 % of original information.
→ Clusters, trends & pattern in data can now be clearly observed.