USN | | | | | | | | | |

Internal Assessment Test 1 – Oct 2023

Sub:	Artificial Intelligence & Machine Learning					Sub Code:	18CS61	Branch:	CSE	
Date:	30.10.2023	Duration:	90 mins	Max Marks:	50	Sem/Sec:	6 A,B,C		OBE	

Answer any FIVE FULL Questions — M | CO | RBT

1 (a) List requirements of good control strategies. (M: **2**, CO: **CO1**, RBT: **L1**)

- Should cause motion
- Should be systematic

Production systems provide structures for AI problems

BFS and DFS provide means for searching through a state space.

Best First Search allows for greedy search.

Heuristic search improves efficiency of search to arrive at non-optimal solutions.

(b) Consider the following 4-gallon water jug problem. (M: **8**, CO: **CO1**, RBT: L2)

You are given 2 jugs, 4-gallon, 3-gallon. There are no measuring markers.

How do you get 2 gallons in the 4-gallon jug?

Write the production rules and solve the problem.

1	(x,y) if x<4	→ (4,y)	Fill 4-gallon jug
2	(x,y) if y<3	→ (x,3)	Fill 3-gallon jug
3	(x,y) if x>0	→(x-d,y)	Pour some water out of the 4-gallon jug
4	(x,y) if y>0	→(x,y-d)	Pour some water out of the 3-gallon jug
5	(x,y) if x>0	→(0,y)	Empty 4-gallon jug
6	(x,y) if y>0	→(x,0)	Empty 3-gallon jug
7	(x,y) if(x+y)≥4 and y>0	→(4,y-(4-x))	Pour water from 3-gallon jug to 4-gallon jug until 4-gallon jug is full

8	(x,y) if(x+y)≥3 and x>0	→(x-(3-y),3)	Pour water from 4 gallon jug to 3-gallon jug until 3-gallon jug is full
9	(x,y) if(x+y)≤4	→(x+y,0)	Pour all water from 3-gallon jug into 4 gallon jug
10	(x,y) if(x+y>≤3	→(0,x+y)	Pour all water from 4-gallon jug into 3 gallon jug
11	(0,2)	→(2,0)	Pour 2 gallons from 3-gallon jug into 4-gallon jug
12	(2,y)	→(0,y)	Empty 2-gallons into 4-gallon jug

Solution

(0,0)

(0,3) #Apply Rule 2

(3,0) #Apply Rule 9

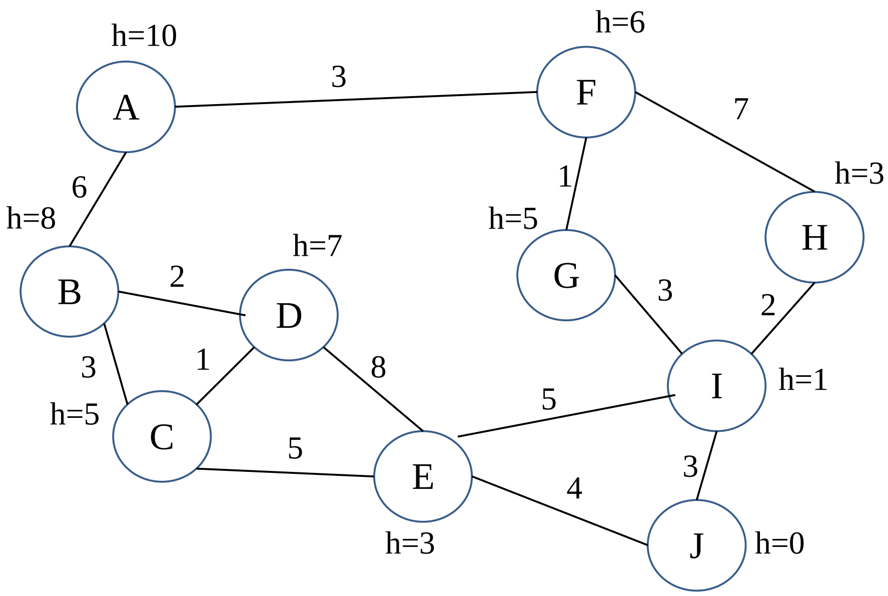
(3,3) #Apply Rule 2

(4,2) #Apply Rule 7

(0,2) #Apply rule 5 or 12

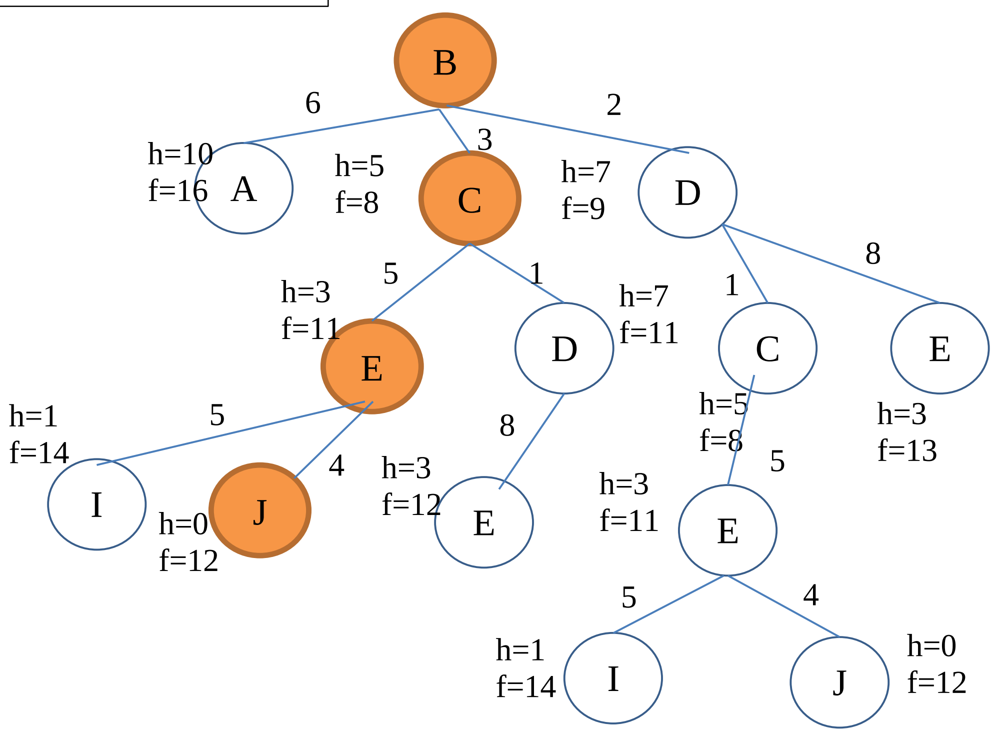
(2,0) #Apply rule 9 or 11

2 (a)	Differentiate exact cost and heuristic cost. Exact cost (g): measure of the **actual cost** Heuristic cost (h): **estimate** of cost from current node to goal node. In a path finding scenario, exact cost is the actual cost from node A to node B. The heuristic path may be a Euclidean distance if any direction of movement is allowed, Manhattan if 4 moves are allowed and diagonal (Chebyshev) if 8 moves are allowed.)	**2**	**CO1**	L2
(b)	i) Use BFS to find path from **B to J** ii) Use DFS to find path from **B to J** iii) Use A* algorithm to find path from **B to J. Give the path and cost. Show steps.** In BFS and DFS, use alphabetical order.	**8**	**CO2**	L3

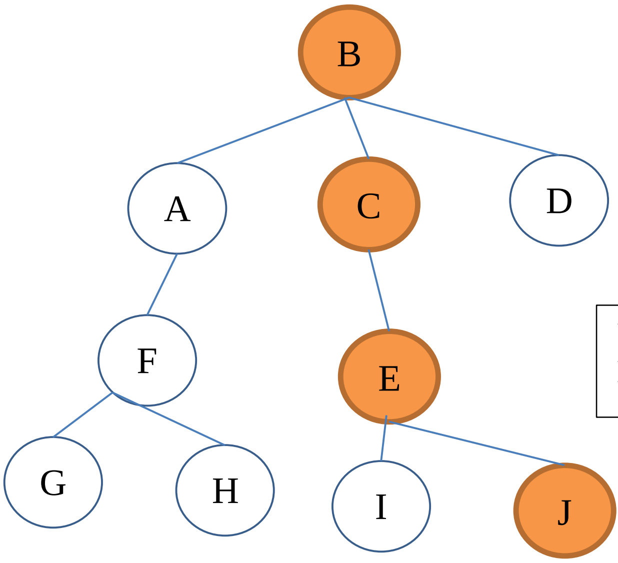


A*

Path found by A*
B->C->E->J – Cost - 12

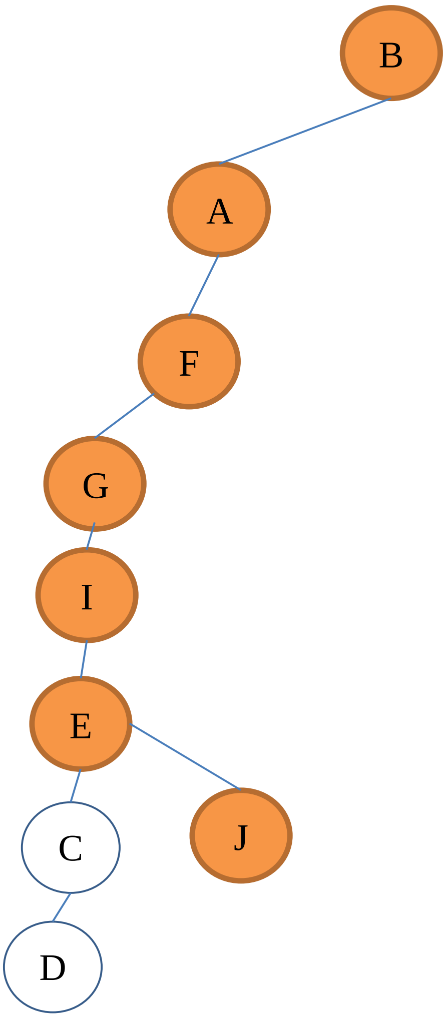


BFS



Path found by BFS
B->C->E->J

DFS



Path found by DFS
B->A->F->G->I->E->J

<table>
<tr><td>3 (a)</td><td>Explain concept of problem reduction using a neat diagram of AND-OR graph
Or graphs find a single path to the goal
AND-OR graphs – represent solutions to a problem that can be solved by decomposing them into smaller portions.


▪ AND arc points to any number of successor nodes, all of which must be solved for arc to point to a solution.
    ▪ Each arm of an AND arc may lead to it’s own solution node.
▪ In order to find solutions in AND-OR graphs, algorithm should be like a best-first search but with ability to handle AND arcs.
▪ It may be necessary to go to more than one solution state because each arm of an AND arc must lead to it’s own solution node.</td><td>4</td><td>CO1</td><td>L2</td></tr>
<tr><td>(b)</td><td>Solve the following crypt arithmetic problem by using constraint satisfaction:
CROSS+ROADS = DANGER
Adding 2 5-digit number will not cause overflow>1
So D=1
D+S = E
For DA > 10, C+R+carry_4 >9
R+0 = N, S+S=R
S>1
R is even
Guess 1: S=2</td><td>6</td><td>CO2</td><td>L3</td></tr>
</table>

Guess 1 : S = 2

```
   4 7 2 2
+ 4 1 5 1 2
-----------
1 - 4 1 3 4
      A≠N
```

• O+A = G
R+O = N

• O > 4

Guess. O = 5
Guess O = 6
Guess O = 7

Guess 2 : S=3

```
  9 6 2 3 3
+ 6 2 5 1 3
-----------
1 5 8 7 4 6
```

{2,5,7,8,9}

• C > 3, 4 E=4, so C > 4
Guess C=5 C≠A so C > 5
Guess C=7 → A=3 S=3
Guess C=8 — A=4 but E=4
Guess C=9

• O + 5 = G
• R+O = N

numbers left are 2, 7, 8

Guess O=2

96233
+ 62513
=158746

4 (a) Use Candidate Elimination to get the hypothesis and version space respectively for the following training examples: **8** **CO2** L3

Ex	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Cloudy	Warm	Normal	Weak	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Cool	Change	No

4	Sunny	Warm	High	Strong	Warm	Change	Yes

Concept learning Task:

Given

Instances X : Possible days described by attributes

- Sky (with values Sunny, Cloudy, Rainy)
- AirTemp (with values Warm, Cold)
- Humidity (with values Normal, High)
- Wind (with values Strong, Weak)
- Water (with values Warm, Cool),
- Forecast (with values Same, Change)

Hypothesis H: with 6 attributes described by conjunction of constraints.

Target concept c : EnjoySport:X→{0,1}

Training Examples, D : 4, 3 positive, 1 negative.

```
S[0]: {('0', '0', '0', '0', '0', '0')}
 G[0]: {('?', '?', '?', '?', '?', '?')}
```

1	Sunny	Warm	Normal	Strong	Warm	Same	Yes

```
S[1]: {('Sunny', 'Warm', 'Normal', 'Strong', 'Warm', 'Same')}
G[1]: {('?', '?', '?', '?', '?', '?')}
```

2	Cloudy	Warm	Normal	Weak	Warm	Same	Yes

```
S[2]: {('?', 'Warm', 'Normal', '?', 'Warm', 'Same')}
 G[2]: {('?', '?', '?', '?', '?', '?')}
```

3	Rainy	Cold	High	Strong	Cool	Change	No

```
('Cloudy', '?', '?', '?', '?', '?'), ('Sunny', '?', '?', '?', '?', '?
') - will not be included because S boundary is more general
('?', '?', '?', 'Weak', '?', '?') - will not be included because S bo
undary is more general

S[3]: {('?', 'Warm', 'Normal', '?', 'Warm', 'Same')}
 G[3]: {('?', 'Warm', '?', '?', '?', '?'), ('?', '?', '?', '?', 'Warm
', '?'), ('?', '?', '?', '?', '?', 'Same'), ('?', '?', 'Normal', '?',
 '?', '?')}
```

4	Sunny	Warm	High	Strong	Warm	Change	Yes

```
Remove from G boundary ('?', '?', '?', '?', '?', 'Same'), ('?', '?',
'Normal', '?', '?', '?')} because S boundary is more general.
```

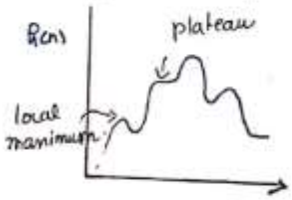
	Version Space
	
	`S[4]: {('?', 'Warm', '?', '?', 'Warm', '?')}`
	`G[4]: {('?', 'Warm', '?', '?', '?', '?'), ('?', '?', '?', '?', 'Warm', '?')}`			

(b)	What are the drawbacks of Find-S as compared to Candidate Elimination? • Find-S finds a hypothesis consistent with the training data but there is no way to determine whether there are other consistent hypothesis also. • Find-S may find a most specific hypothesis. We may want a more general hypothesis also. • Training examples are error-prone / noisy. Inconsistent sets severely mislead Find-S as it ignores negative samples. • There should be a mechanism to back track from maximally specific hypothesis reported by Find-S to accommodate possibility of a target concept on a different branch.	**3**	**CO1**	L2
5 (a)	What are the problem characteristics that must be analyzed for deciding heuristic to use. 1. Is the problem decomposable 2. Can solution steps be undone or ignored? 3. Is the universe predictable? 4. Is a good solution absolute or relative? 5. Is the solution a state or a path? 6. The role of knowledge 7. Does a task require interaction with a person?	**3**	**CO2**	L2
(b)	Explain simulated annealing with an example. The intuition behind simulated annealing comes from the concept of annealing where metals are cooled until solid state is reached. It usually moves from a higher energy state to a lower one but there is some probability of it moving to a higher energy state also. $$P = 1 - e^{\frac{-\Delta E}{k*t}}$$ Where ΔE is the change in the energy level, T is the temperature K is the Boltzmann's constant that gives the correspondence between units of temperature and units of energy. Algorithm 1. Evaluate initial state. If it is goal state, return, quit. Otherwise, continue with initial state as current state	**4**	**CO2**	L2

	2. Initialize best_so_far to current state
	3. Initialize T as per annealing schedule
	4. Loop until solution OR no new operator is left in current state:
	    a. Select an operator yet to be applied and apply it to produce new state.
	    b. Evaluate new state. Compute $\Delta E = value\ of\ current - value\ of\ new\ state$
	    If new state is goal state, return, quite
	    If goal state is better than current state, make it current.
	    Set best_so_far to new state
	    If it is not better than current state, make it current state with probability $p' = e^{-\Delta E/T}$
	    c. Revise T as necessary, according to annealing schedule.
	5. Return best_so_far as answer			

(c)	What is "Inductive Learning Hypothesis" Any hypothesis found to approximate target function well over a sufficiently large set of training examples will also approximate the target function well over other unobserved examples.	**2**	**CO1**	L1

6 (a)	Explain Steepest Hill climbing algorithm. Explain it's limitations with a neat diagram or example.
	Algorithm:
	1. Evaluate initial state. If it is goal state, return
	2. Loop
	    a. Let SUCC be successor of current state
	    b. For each operator that applies to current state, Do
	        i. Apply operator to generate new state
	        ii. Evaluate new state. If it is goal state, return it, quit.
	        If not, compare with SUCC
	        If it is better, set SUCC to this state
	        If it is not better, leave SUCC as it is.
	    c. If SUCC is better than current state, set current state to SUCC
		5	**CO1**	L2



local maximum: better than all its neighbors but not better than states further away.

plateau: flat area in search space, where all neighboring states have same value.

ridge: A special kind of local maximum. Higher than surrounding areas.

* backtrack
* Make big jump.

(b) Apply Find-S and find the maximally specific hypothesis — 5 — CO1 — L2

EXAMPLE	COLOR	TOUGHNESS	FUNGUS	APPEARANCE	POISONOUS
1.	GREEN	HARD	NO	WRINKELD	YES
2.	GREEN	HARD	YES	SMOOTH	NO
3.	BROWN	SOFT	NO	WRINKLED	NO
4.	ORANGE	HARD	NO	WRINKLED	YES
5.	GREEN	SOFT	YES	SMOOTH	YES
6.	GREEN	HARD	YES	WRINKLED	YES
7.	ORANGE	HARD	NO	WRINKLED	YES

ATTRIBUTES ON WHICH THE CONCEPT DEPENDS ON — CONCEPT

Concept learning Task:

Given

Instances X : Possible varieties described by attributes

- Color (with values Hard, Soft)
- Toughness(with values Warm, Cold)
- Fungus (with values No, Yes)
- Appearance (with values Wrinkled, Smooth)

Hypothesis H: with 4 attributes described by conjunction of constraints.

Target concept c : Poisonous:X→{0,1}

Training Examples, D : 7, 5 positive, 2 negative.

$$S0 = \{\emptyset, \emptyset, \emptyset, \emptyset\}$$

1 – Positve (Green, Hard, No, Wrinkled)

$$S1 = \{Green, Hard, No, Wrinkled\}$$

2,3 – Negative examples so ignore

4 – Positive example (Orange,Hard, No, Wrinkled)

	$S4 = \{?, Hard, No, Wrinkled\}$ 5-Positive example (Green, Soft, Yes,Smooth) $S4 = \{?, ?, ?, ?\}$			

CI **CCI** **HOD**

Course Outcomes		Blooms Level	Modules covered	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4	
CO1	Appraise the theory of Artificial intelligence and Machine Learning.	**L2**	**1,2**	3	3	2	2	0	2	2	0	0	0	0	0	0	2	0	3	
CO2	Illustrate the working of AI and ML Algorithms.	**L3**	**2,3,4**	3	3	3	3	3	3	0	0	0	0	0	0	0	2	0	3	
CO3	Demonstrate the applications of AI and ML.	**L2**	**4,5**	3	3	3	3	3	3	0	0	0	0	0	0	0	2	0	3	

CO PO Mapping

COGNITIVE LEVEL	REVISED BLOOMS TAXONOMY KEYWORDS
L1	List, define, tell, describe, identify, show, label, collect, examine, tabulate, quote, name, who, when, where, etc.
L2	summarize, describe, interpret, contrast, predict, associate, distinguish, estimate, differentiate, discuss, extend
L3	Apply, demonstrate, calculate, complete, illustrate, show, solve, examine, modify, relate, change, classify, experiment, discover.
L4	Analyze, separate, order, explain, connect, classify, arrange, divide, compare, select, explain, infer.
L5	Assess, decide, rank, grade, test, measure, recommend, convince, select, judge, explain, discriminate, support, conclude, compare, summarize.

PROGRAM OUTCOMES (PO), PROGRAM SPECIFIC OUTCOMES (PSO)				CORRELATION LEVELS	
PO1	Engineering knowledge	PO7	Environment and sustainability	0	No Correlation
PO2	Problem analysis	PO8	Ethics	1	Slight/Low
PO3	Design/development of solutions	PO9	Individual and team work	2	Moderate/ Medium
PO4	Conduct investigations of complex problems	PO10	Communication	3	Substantial/ High
PO5	Modern tool usage	PO11	Project management and finance		
PO6	The Engineer and society	PO12	Life-long learning		
PSO1	Develop applications using different stacks of web and programming technologies				
PSO2	Design and develop secure, parallel, distributed, networked, and digital systems				
PSO3	Apply software engineering methods to design, develop, test and manage software systems.				
PSO4	Develop intelligent applications for business and industry				