Roll No.												

PRESIDENCY UNIVERSITY

BENGALURU

Mid - Term Examinations – March 2026

Date: 14 -03-2026 **Time:** 09.30am to 11.00am

School: SOCSE	**Program:** CBC, CBD, CDV, COM, CSD, CSE, CSG, CSI, CSN, IST	
Course Code: CSE2264	**Course Name:** Essentials of AI	
Semester: IV	**Max Marks**: 50	**Weightage**: 25%

CO - Levels	**CO1**	**CO2**	**CO3**	**CO4**	**CO5**
Marks	**20**	**30**			

Instructions:

(i) Read all questions carefully and answer accordingly.

(ii) Do not write anything on the question paper other than roll number.

Part A

Answer ALL the Questions. Each question carries 2marks. **5Q x 2M=10M**

1	State true or false. IDDFS has worse asymptotic time complexity than BFS because IDDFS repeatedly searches at every level till the solution, unlike BFS which searches only once.	**2 Marks**	**L1**	**CO1**
2	State yes or no. Given the jugs with volumes 2 litre and 4 litre we can fill the volume of 3 litres.	**2 Marks**	**L1**	**CO1**
3	State the term which means that, for all nodes, the heuristic value is less than or equal to the minimum cost to reach the goal node.	**2 Marks**	**L1**	**CO1**
4	Given the function $f(x) = x^2$, state the value of the gradient ($f'(x)$) at $x = 1$.	**2 Marks**	**L1**	**CO1**
5	State the term used for changing the value of a part of an intermediate solution in genetic algorithms.	**2 Marks**	**L1**	**CO1**

Part B

Answer the Questions. **Total Marks 40M**

<table>
<tr><td>6.</td><td>Consider a situation where we are performing uninformed search. The solution state is at a depth d = 5 from the start state (where the start state is at a depth of d = 0). Each state generates 10 more states (i.e. the branching factor b = 10). Compute the number of operations needed for the BFS and the IDDFS.</td><td>10 Marks</td><td>L2</td><td>CO1</td></tr>
<tr><td colspan="5">Or</td></tr>
<tr><td>7.</td><td>Consider a situation of the water jug problem (similar to one of the questions in CA1). Here, we have 3 jugs of volumes 3 litres, 5 litres and 8 litres, and we have to fill the 5 litre jug and the 8 litre jug with 4 litres each. Initially, the 8 litre jug is full, and the 3 litres and 5 litres jugs are empty. Unlike the standard 3 litre and 5 litre water jug problem, we cannot empty or fill any jug from an external source.

Recall the rules from the original problem:

• Fill(x) – Fills the x litre jug
• Empty(x) – Empties the x litre jug
• Pour(x, y) – Pours water from the x litre jug to the y litre jug.

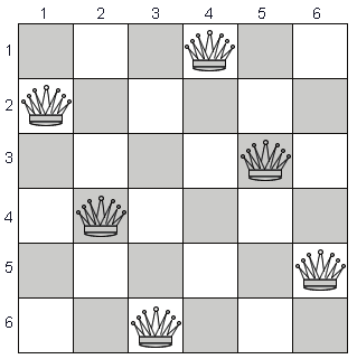
In the new problem, we can only pour from 1 jug to another.

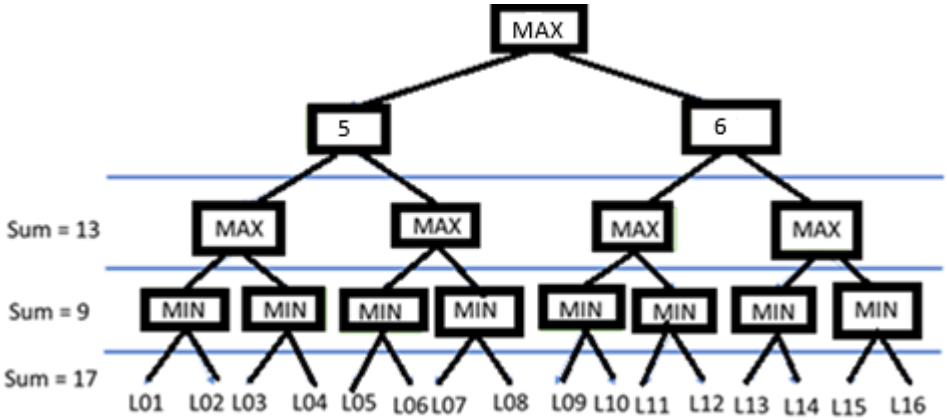
With that in mind, complete the below table by computing to come up with equivalent rules for the new water jug problem (Pour(5,3) is given as an example):

<table>
<tr><td>Old Rule</td><td>Pour from X</td><td>Pour to Y</td></tr>
<tr><td>Fill(3)</td><td>_____</td><td>_____</td></tr>
<tr><td>Fill(5)</td><td>_____</td><td>_____</td></tr>
<tr><td>Empty(3)</td><td>_____</td><td>_____</td></tr>
<tr><td>Empty(5)</td><td>_____</td><td>_____</td></tr>
<tr><td>Pour(3,5)</td><td>_____</td><td>_____</td></tr>
<tr><td>Pour(5,3)</td><td>5 litres</td><td>3 litres</td></tr>
</table>
NOTE: For your answer, you should only write the values 3 litres, 5 litres, and 8 litres in the blanks.</td><td>10 Marks</td><td>L2</td><td>CO1</td></tr>
</table>

8.	The nation of PanEm has a central Capital City (called Capital), and is surrounded by a number of **N** tributaries called Sectors. Each sector borders 2 adjacently numbered sectors (Sector i borders Sector (i-1) and Sector (i+1), with Sector N bordering Sector (N-1) and Sector 1) and Capital. If Capital is coloured BLUE in the map, calculate the number of colours needed to colour the map if (a) N is even and (b) N is odd. Based on this, complete the below table:	**15 Marks**	**L3**	**CO2**

		N	No. of colours for the map			
		3				
		4				
		99				
		100				
		1000000				
	NOTE: For calculating the number of colours if N is even or odd, you need to write a brief explanation, not just the final number of colours.					
	Or					
9.	Solve the cryptarithmetic problem: TWO + TWO = FOUR where all the digits are in **BASE 7** (i.e. all the digits should be from 0 to 6).			**15 Marks**	**L3**	**CO2**

10.	Consider a ***6 Queen's Problem*** where we have to arrange 6 queens on a 6x6 chessboard. We plan to solve this using local search by having a fitness function F(c), where c is the configuration of the chessboard, represented as a 6-dimension vector where each number is the row-number of the queen on the chessboard.  For example, in the above figure the configuration, C = [2,4,6,1,3,5] and our fitness function F(C) = 15 – P, where P is the number of *distinct attacking pairs of queens* (NOTE: Q1 attacking Q2 is considered the same as Q2 attacking Q1). **Calculate the fitness function of [1,3,5,4,2,6]**. NOTE: You must not solve the 6-Queen's Problem (the above is a solution). You only have to calculate the value of F([1,3,5,4,2,6]).	**15 Marks**	**L3**	**CO2**
	Or			
11.	Consider the following game. Every move has 2 options for the players either a left move (L) or a right move (R), and where at a given level (after the first time the MIN player plays), the sum of utilities of the	**15 Marks**	**L3**	**CO2**

	children are the same. For example, in the below tree, the sum of utilities of each of the leaf nodes are 17, so each pair of leaves could be either 1&16, 2&15, etc.  Calculate all the leaf nodes (L01 to L16) of the above tree **AFTER IDEAL ORDERING.** NOTE: The given tree **may not be ideally ordered**. Then, perform alpha-beta pruning to find out all the leaves which are pruned.			