Birla Institute of Technology & Science, Pilani
Work Integrated Learning Programmes Division
First / 2024-2025

Mid-Semester Test
(EC-2 Regular)

Course No.	:	AIMLCZG557
Course Title	:	Artificial and Computational Intelligence
Nature of Exam	:	
Weightage	:	30%
Duration	:	2 Hours
Date of Exam	:	29-06-2025 (AN)

No. of Pages = 3
No. of Questions = 4

Note to Students:
1. Please follow all the *Instructions to Candidates* given on the cover page of the answer book.
2. All parts of a question should be answered consecutively. Each answer should start from a fresh page.
3. Assumptions made if any, should be stated clearly at the beginning of your answer.

Q1 Answer to the below question for the following scenarios. Vague theory will not be awarded marks. **[2+2+2= 6 Marks]** — 6 Marks

Imagine you are tasked with designing an automated quality control system for a manufacturing facility producing electronic components. The system is equipped with sensors installed at critical points along the production line to monitor and gather data on the quality of components being produced. It scans for defects and irregularities, comparing them with a database of known manufacturing standards to ensure that only high-quality components are shipped to customers.

a. Provide the complete problem formulation.
b. Provide the PEAS description.
c. Identify the various dimensions of the task environment with appropriate justification for each in no more than 30 words

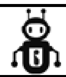
Sample Solution and marking scheme:

a. Problem Formulation 2 Marks

- **Initial State**: The production line is active; components are being manufactured; sensors are initialized; no component is inspected yet.
- **Goal State**: All manufactured components are scanned, evaluated, and classified as either *defective* or *non-defective*. Only *non-defective* components are allowed for shipment.
- **Actions (Operators)**:
  - Scan component using sensors
  - Compare scanned data with standards in database
  - Classify component as defective or non-defective
  - Reject or approve component

	• **Transition Model**: After each scan and comparison, the system updates the classification status of a component and forwards it to the appropriate destination (rework/reject or shipment). • **Path Cost**: Minimize error rate (false positives/negatives) to optimize throughput and accuracy. **b. PEAS Description [0.5 * 4 = 2 marks]** **Performance**: Accuracy of defect detection, false positive/negative rate, inspection speed, throughput, and cost-efficiency. **Environment**: Factory floor, production line, varying lighting, temperatures, and component types. **Actuators**: Robotic arms (to remove defective items), conveyor controls, alarm systems, visual display units. **Sensors**: Cameras, infrared scanners, X-ray sensors, vibration sensors, optical sensors, barcode/RFID readers. **c. Dimensions of the Task Environment (with justification) – At least 4 with justification –[2 marks]** 1. **Fully Observable** – All component attributes can be observed by sensors. 2. **Deterministic** – Given sensor input, component classification follows a predictable rule. 3. **Episodic** – Each inspection is independent; no past data influences current decision. 4. **Static** – The environment doesn't change during component inspection. 5. **Discrete** – Each component has a finite set of states (defective, non-defective). 6. **Single-Agent** – Only the QC system acts to achieve its goal.	
Q2	The Adventure agent stands at the entrance of a mysterious cave, marked as LOC A1. This cave contains lot of precious gems, including diamonds and rubies. However, the cave is also replete with obstacles, traps, and challenges, that only the bravest adventurers can conquer. The aim of Treasure agent is to efficiently reach the "exit" in the shortest way possible and tries to grab as much as diamonds and rubies while exploring. The agent is equipped with four distinct actions: MoveUp, MoveDown, MoveLeft, and MoveRight, but it cannot move into black cells. Furthermore, ***when the agent reaches a cell containing gems, it automatically collects them***. **[3+3+3 = 9 Marks]**	9 Marks

Initial State:

	1	2	3	4
A		R	D	SW
B	D	SW	R	SW
C	R		R	EXIT
D	D	SW	SW	D

D : Diamond

R : Ruby

SW : Spider web

a. Expand & depict the search tree for up to exactly three levels (level 1, level 2 & level 3). (Given initial state can be assumed to be on level-0.)

b. Calculate the path cost and heuristic values for all generated nodes in the search tree.

Path cost calculation:

- If the agent enters a cell with a spider web, its energy diminishes and requires a reboot, adding the cost of **+15.** If the agent grabs a diamond, the energy automatically increases, adding the cost by **-10**. If the agent grabs a ruby, the cost is added by **-5** for that resultant state.

Heuristic design for calculation:

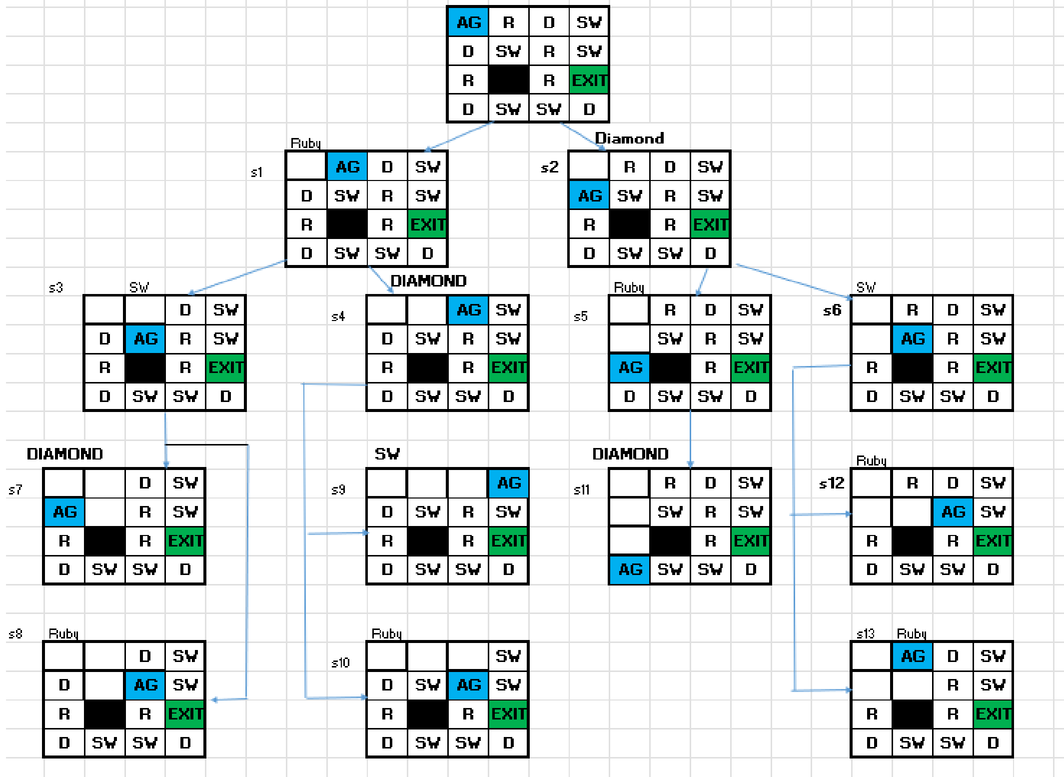
- H(n) = Min [Manhattan distance (Agent current position , Exit))] + No of remaining gems in resultant state + Number of spider web adjacent to Agent's position in the resultant state.

c. Apply IDA* search algorithm for the search tree obtained from part a and using heuristic design calculated under part b, only till first 4 closed list updates or till no more nodes are left. Show the status of OPEN and CLOSE list at each level and the step by step procedure as discussed in the class.

Sample Solution and marking scheme:

a) Constructing the complete game tree – 3 Marks

- Search Tree for level 1 expansion – 0.5 mark
- Search Tree for level 2 expansion – 0.5 mark
- Search Tree for level 3 expansion – 2 mark

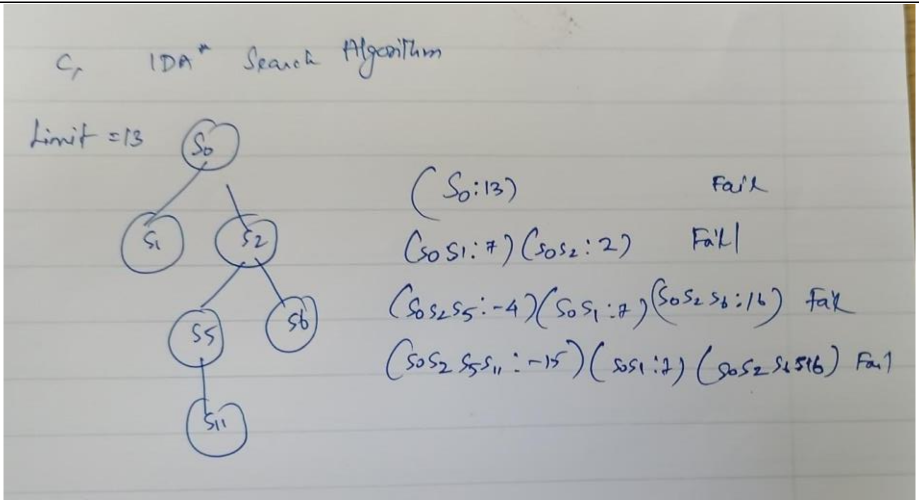


b. Calculating static evaluation functions- each level 1 mark (1* 3 = 3 Marks)

		G(N)	Heuristic design					
		Path Cost	CUM PATH COST	DIST	No rem gems	No adj SW	H(n)	F(n)
S1	RUBY	-5	-5	4	7	1	12	7
S2	DIAMOND	-10	-10	4	7	1	12	2
S3	SW	15	10	3	7	0	10	20
S4	DIAMOND	-10	-15	3	6	1	10	-5
S5	RUBY	-5	-15	5	6	0	11	-4
S6	SW	15	5	3	7	0	10	15
S7	DIAMOND	-10	0	4	6	0	10	10
S8	RUBY	-5	5	2	6	1	9	14
S9	SW	15	0	2	6	1	9	9
S10	RUBY	-5	-20	2	5	2	9	-11
S11	DIAMOND	-10	-25	4	5	1	10	-15
S12	RUBY	-5	0	2	6	1	9	9
S13	RUBY	-5	0	4	6	0	10	10

c. Using IDA* algorithm game tree construction – 2 Mark

Best path chosen – 1 Mark



Q3 Arjun, a fire department route planner, needs to optimize the response route for a fire truck that starts from the Main Fire Station (MFS) and must inspect two high-risk fire zones—Zone A (ZA) and Zone B (ZB)—before returning to the station. Each inspection includes checking hydrants, equipment readiness, and evacuation plans. Due to urban traffic and road layouts, travel distances between the points vary and are known. **The goal is to determine the** shortest route **the fire truck should take using the** Ant Colony Optimization (ACO) **algorithm**.

8 Marks

Distance Matrix :

	MFS	ZA	ZB
MFS	0	6	8
ZA	6	0	5
ZB	8	5	0

Initial Pheromone Matrix :

	MFS	ZA	ZB
MFS	0	0.20	0.5
ZA	0.20	0	0.3
ZB	0.5	0.3	0

MFS = Main Fire Station (start and return point)

ZA = Fire Zone A

ZB = Fire Zone B

Rate of evaporation = 0.1; Q = 100; The relative importance of pheromone is 0.5 and the relative importance of distance is 0.3;

Sample Solution and marking scheme:

1) Next Transition Probability:

- **Formula : 1 Marks**
- **Calculation : 2 Marks**

Transition Probability Formula :

$$P_{ij} = \frac{(\tau_{ij})^{\alpha} \cdot (\eta_{ij})^{\beta}}{\sum_{h \notin V} (\tau_{ih})^{\alpha} \cdot (\eta_{ij})^{\beta}}$$

-------------------> 1 Mark

(1 Marks)	η
MFS - ZA	0.167
ZA - ZB	0.200
ZB - MFS	0.125

	Probability **(1 Marks)**
MFS - ZA	0.408
MFS - ZB	**0.592 (max)**

2) Pheromone Updation :

- **Formula : 0.5 Marks**
- **Calculation : 4 Marks**
- **Final path : 0.5 Marks**

$$\tau_{ij}^{new} = (1-\rho)\tau_{ij}^{old} + \Delta\tau_{ij}^{k}$$

$$\Delta\tau_{ij}^{k} = \begin{cases} \frac{Q}{f_k} & ; \text{ if } k^{th} \text{ ant passes } i-j \\ 0 & ; \text{ otherwise} \end{cases}$$

---→ 0.5 Marks

1.5 Marks

VISIT [MFS - ZB]

Between	τ	1-ρ	Δτ	(1-ρ)τ+ Δτ
MFS - ZA	0.20	0.9	0	0.1800
ZA - ZB	0.30	0.9	0	0.2700
ZB - MFS	0.50	0.9	12.5	12.9500

1.5 Marks

VISIT [MFS - ZB - ZA]

Between	τ	1-ρ	Δτ	(1-ρ)τ+ Δτ
MFS - ZA	0.18	0.9	0	0.1620
ZA - ZB	0.27	0.9	20	20.2430
ZB - MFS	12.95	0.9	0	11.6550

1 Marks

VISIT [MFS - ZB - ZA - MFS]

Between	τ	$1-\rho$	$\Delta\tau$	$(1-\rho)\tau+\Delta\tau$
MFS - ZA	0.16	0.9	16.67	16.8125
ZA - ZB	20.24	0.9	0	18.2187
ZB - MFS	11.66	0.9	0	10.4895

Final tour is

MFS $\longrightarrow$ ZB $\longrightarrow$ ZA $\longrightarrow$ MFS.

Total distance is 8+5+6 = 19.

----------------------- > **0.5 Marks**

Q4 You have been asked to solve below linear equation problem with multiple variables using genetic algorithm: **[2+2+3 = 7 Marks]** — 7 Marks

$$2a - 4b + 2c - 5d + 5 = 50$$

where a,b,c and d are integers in the range [-10,10] (equation can have more than one solution)

a. Design the Problem Solving Agent formulation & fitness function. The fitness score of a particular state can be determined by calculating the difference between the left-hand side (LHS) and right-hand side (RHS) values of the equation. A higher difference will result in a lower fitness score. Explain with numerical example.

b. Describe the Chromosome/String representation of a parent state for **four randomly** selected states with their fitness score.

c. Detail the approach toward the selection, crossover and mutation steps for this problem. Show these with only one iteration of numerical example.

Sample Solution and marking scheme:

a) For Complete problem formulation (Initial state, Goal test, transition, path cost, possible action) – 1.5 Mark

Numerical example for Fitness score calculation – 0.5 Mark

Component	Description
Initial State	Random population of [a, b, c, d] with values ∈ [-10, 10]
Goal Test	Check if 2a - 4b + 2c - 5d + 5 = 50
Transition	Crossover and mutation operations to evolve new states
Path Cost	Implied by fitness function: lower error = better solution
Possible Actions	Selection, Crossover, Mutation

b) Chromosome representation of 4 randomly selected state -1 Marks

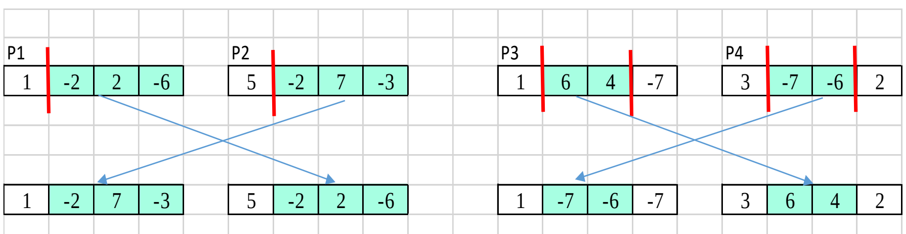
Fitness score Calculation – 1 Marks

Chromosome	a	b	c	d	L.H.S	R.H.S	**Fitness Score**
Parent 1	0	1	2	0	52	50	**2**
Parent 2	1	2	0	0	49	50	**1**
Parent 3	2	1	1	0	17	50	**33**
Parent 4	1	0	1	2	26	50	**24**

c) Selection with justification – 1 Mark, without justification 0.5

P1				P2				P3				P4			
1	-2	2	-6	5	-2	7	-3	1	6	4	-7	3	-7	-6	2

Crossover steps as discussed in class- 1 Mark without explanation 0.5



Mutation with detailed explanation - 1 Mark without explanation 0.5

Before Mutation —Random Gene Selection

1	-2	7	-3	5	-2	2	-6	1	-7	-6	-7	3	6	4	2

After Mutation — Gene modification

1	-2	8	-3	5	6	2	-6	1	-7	-6	-7	3	6	4	2

*****