

## Artificial & Computational Intelligence

### Assignment 1 - Question 1

#### Problem statement

There are two agents named R1 and G1. Both are searching for a "heart" as shown in the below configuration as "H" that gives everlasting power. Both agents are trying to reach the heart. In this process many obstacles may be encountered to reach the heart. Help them in finding the best path to reach the heart from any arbitrary start positions. [Dynamically fetch the start position while executing the code]

For the agent R1 the obstacle is the green room. If R1 enters the green room it incurs a penalty of +10 cost and if it uses the red room it incurs a penalty of -10 points. For the agent G1 the obstacle is the red room. If G1 enters the red room it incurs a penalty of +10 cost and if it uses the green room it incurs a penalty of -10 points. In addition to the given cost, for every transition an agent visits incurs a path cost of 1.

For any arbitrary node "n" the heuristic to reach the Heart  $h(n)$  is given by the below:

$$\text{Manhattan distance} + \text{Color Penalty}$$

where, Color Penalty = +5 if the node "n" and goal node is in different colored room  
and Color Penalty = -5 if the node "n" and goal node is in same colored room

Use the A\* algorithm for both the below configurations and interpret which agent works well in which environment. Justify your interpretation with relevant performance metrics.

**Note:** The agents are not competing with each other. You need to run the simulation for both agents in each of the below scenarios separately & submit the results of 4 runs.

Scenario I							Scenario II						
	0	1	2	3	4	5		0	1	2	3	4	5
0							0						
1							1						
2				H			2			H			
3							3						
4							4						
5							5						

**Evaluations will be based on the following:**

1. Explain the PEAS (Performance measure, Environment, Actuator, Sensor.) for your agent. (20% marks)
2. Use the above mentioned algorithm for all the scenarios and implement in PYTHON. (20% + 20% = 40% marks)
3. Print the simulation results. (20% marks)
4. Include code in your implementation to calculate the space complexity and time complexity for the informed search and print the same. For local search interpret the significance of the hyperparameters if any applicable . (20% marks)

Note 2:

- You are provided with the python notebook template which stipulates the structure of code and documentation. Use well intended python code.
- Use a separate MS word document for explaining the theory part. Do not include the theory part in the Python notebook except Python comments.
- The implementation code must be completely original and executable.
- Please keep your work (code, documentation) confidential. If your code is found to be plagiarized, you will be penalized severely. **Parties involved in the copy will be considered equal partners and will be penalized severely.**