

Total points: 100

HW 1: **Agents and Uninformed Search**

Due date: Oct 16, 2023

Instructions: This homework assignment consists of a written portion and a programming portion. Collaboration is not allowed on any part of this assignment. Solutions must be typed (hand written and scanned submissions will not be accepted) and saved as a .pdf file. You will submit a single .zip file that contains the the code base and solutions as a .pdf file.

1. **(30 points)** Complete the following tasks for a smart home assistant such as Amazon Alexa or Google Home.
 - (a) (10 points) Develop a PEAS description (Performance, Environment, Actuators, Sensors)
 - (b) (10 points) Describe its environment according to the following properties:
 - Fully vs Partially Observable
 - Deterministic vs Stochastic
 - Episodic vs Sequential
 - Static vs Dynamic
 - Discrete vs Continuous
 - Single agent vs Multi-agent

Note that in some cases, both answers might be correct. Justify each answer to the task environment properties with a one sentence explanation.

- (c) What type of agent design (goal-directed or utility-directed) is best suited for this problem? Briefly justify your design choice.
2. **(10 points)** For each of the following, identify if the system is an automated system or an intelligent system. Briefly justify your answer.
 - A water sprinkler system for your home garden
 - A home assistant like Amazon Alexa
 - HVAC system that adjusts temperature based on occupancy levels
 - A route guidance system such as Google maps

3. **(30 points)** For the graph in Figure 1, implement **Breadth First Search** and complete the table below with the order in which the nodes will be expanded. Include a screenshot of your code output in this document, in addition to filling in the table below. You are provided with a skeleton code in Python. You can also code it up your own way or in another language of your choice.

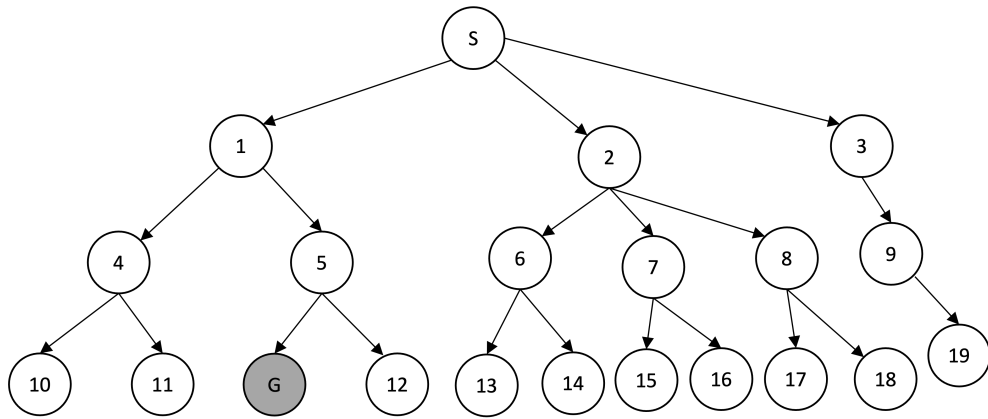


Figure 1:

Expanded Nodes List	Frontier List

4. **(30 points)** For the graph in Figure 1, implement **Depth First Search** and complete the table below with the order in which the nodes will be expanded. Include a screenshot of your code output in this document, in addition to filling in the table below. You are provided with a skeleton code in Python. You can also code it up your own way or in another language of your choice.

Expanded Nodes List	Frontier List