Name: Ya Zou
Email: zouy2@oregonstate.edu

# HW 1: Agents and Uninformed Search

1 (a). Develop a PEAS description (Performance, Environment, Actuator, Sensor)

| Performance | Environment | Actuator | Sensor |
|---|---|---|---|
| **1. Accuracy**(Correctly identify the physical activity (sitting, standing, walking, running, or other) with high precision.)<br>**Energy** efficiency(Minimize battery consumption and computational resources.)<br><br>**Responsiveness**(Provide real-time activity recognition)<br>**User Satisfaction**(Ensure the user finds the activity recognition reliable and useful) | **Physical location**( home, office, outdoors, gym)<br><br>**Device**( The agent operates on a smartphone which may be in different positions such as pocket, hand, and armband)<br><br>**User Activitie**s(sitting, standing, walking and others) | **Display screen**<br>**Speaker**<br>**Internet**<br>**Notification system**<br>**Data storage**<br>**User interface** | **Accelerometer**: To detect movement and orientation changes.<br><br>**Gyroscope**<br><br>**GPS** |

1(b).Describe its environment according to the following properties:

- **Partially observable:** Because the system only has access to accelerometer data, there might be noisy and inaccurate or incomplete sensor data because data from the accelerometer is on a cell phone.
- **Stochastic:** The environment is stochastic because the same sensor readings can result from different activities due to variations in user behavior, device placement, and external factors.
- **Sequential:** The environment is sequential because the decisions (activity classifications) made by the agent are dependent on the previous actions or data points over time
- **Dynamic:** The user's activity changes while an agent is thinking such as user's activities change over time,
- **Continuous:** The environment is continuous because accelerometer data is generated continuously over time, providing a constant stream of information that needs to be processed.
- **Single-agent:** The phone is the only agent working on physical activity recognition.

1(c). What type of agent design(goal-directed or utility-directed) is best suited for this problem?

Utility-based agents will be more reasonable for this system because the agent runs on a cell phone and performs physical activity recognition using data, it doesn't have a specific goal but rather optimizes the ongoing task (recognizing physical activity) based on various factors like accuracy, efficiency, and resource constraints.

2. For each of the following, identify if the system is an automated system or an intelligent system. Briefly justify your answer.

- At a grocery store, customers use a self-checkout kiosk to scan and pay for their items. The system provides instructions, processes payment, and dispenses a receipt. Is the self-checkout kiosk an automated system or an intelligent system? Briefly justify your answer.
  **Automated system:** because grocery stores have all of the products in the database and all products are labeled with a barcode, so we only need an automated system to map what products to respective prices then we can sum up the prices for customers.It doent adapt or learn from interactions but performs specific tasks based on programmed rules.

- A water sprinkler system for your home garden that runs for 30 minutes everyday at 4:00 pm. Is this water sprinkler an automated system or an intelligent system? Briefly justify your answer.

**Automated system:** This system operates on a fixed schedule (running for 30 minutes every day at 4:00 pm). There is no adaptation to environmental factors (like weather or soil moisture) or decision-making involved, and the system performs a repetitive task based on a timer.

- HVAC system that adjusts the room temperature based on occupancy levels in the room. Is this HVAC system an automated system or an intelligent system? Briefly justify your answer.
  **Intelligent system:** because this system adjusts its behavior based on real-time information, such as occupancy levels. It requires sensors to detect whether people are in the room and uses this data to modify the temperature accordingly.

- A large e-commerce company uses machines to move items from shelves to packing stations. The machines pick up items based on a barcode scanner that identifies where the item should go, and will place them in respective shelves. Is this machine automated or intelligent? Briefly justify your answer.
  **Automated system:** the machines follow pre-defined rules by reading barcodes and moving items to designated locations based on that information. While the machines are capable of executing tasks efficiently, they do not learn or make decisions beyond the scope of their programming.
- A vacuum cleaner that can identify high-trafic areas where dust a accumulates most frequently and adjusts its cleaning patterns accordingly. It this vacuum cleaner automated or intelligent?

  **Intelligent System:** because the system needs to identify high-traffic areas where dust acclimates more frequently and adjusts its cleaning pattern, all these must be learned during the decision-making process.

3. 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.

| Expanded Nodes List | Frontier List |
|---|---|
| | {s} |
| {s} | {1,2,3} |
| {s,1} | {2,3,4,5} |
| {s,1,2} | {3,4,5,6,7,8} |
| {s,1,2,3} | {4,5,6,7,8,9} |
| {s,1,2,3,4} | {5,6,7,8,9,10,11} |

| | |
|---|---|
| {s,1,2,3,4,5} | {6,7,8,9,10,11,G,12} |
| {s,1,2,3,4,5,6} | {7,8,9,10,11,G,12,13} |
| {s,1,2,3,4,5,6,7} | {8,9,10,11,G,12,13,14,15} |
| {s,1,2,3,4,5,6,7,8} | {9,10,11,G,12,13,14,15,16,17} |
| {s,1,2,3,4,5,6,7,8,9} | {10,11,G,12,13,14,15,16,17,19} |
| {s,1,2,3,4,5,6,7,8,9,10} | {11,G,12,13,14,15,16,17,19} |
| {s,1,2,3,4,5,6,7,8,9,10,11} | {G,12,13,14,15,16,17,19} |
| {s,1,2,3,4,5,6,7,8,9,10,11,G} | {} 12. |

bfs.csv > data

```
1   ,bfs
2   0,S
3   1,1
4   2,2
5   3,3
6   4,4
7   5,5
8   6,6
9   7,7
10  8,8
11  9,9
12  10,10
13  11,11
14  12,G
```

```python
def bfs(visited, graph, node, solution):
    """Breadth First Search Function to implement
    visited (set): set of visited nodes
    graph (dictionary): input graph to search on
    node (string): node to expand
    solution (list): the solution
    """
    # TODO: Write your code here
    visited = set()
    queue = [node]
    visited.add(node)
    while queue:
        current_node = queue.pop(0)
        solution.append(current_node)
        if current_node == 'G':
            break
        for adjacent in graph.get(current_node, []):
            if adjacent not in visited:
                visited.add(adjacent)
                queue.append(adjacent)
```

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 |
|---|---|
| | {s} |
| {s} | {1,2,3} |
| {s,1} | {4,5,2,3} |
| {s,1,4} | {10,11,5,2,3} |
| {s,1,4,10} | {11,5,2,3} |
| {s,1,4,10,11} | {5,2,3} |
| {s,1,4,10,11,5} | {G} |
| {s,1,4,10,11,5,G} | {} |

```python
def dfs(visited, graph, node, solution):
  """Depth First Search Function to implement
  visited (set): set of visited nodes
  graph (dictionary): input graph to search on
  node (string): node to expand
  solution (list): the solution
  """
  # TODO: Write your code here
  if node not in visited:
    visited.add(node)
    solution.append(node)
    if node =='G':
      return True
    for adjacent in graph.get(node,[]):
      if dfs(visited, graph, adjacent, solution):
        return True
  return False
```

dfs.csv U

dfs.csv > data

```
1   ,solution
2   0,S
3   1,1
4   2,4
5   3,10
6   4,11
7   5,5
8   6,G
9
```

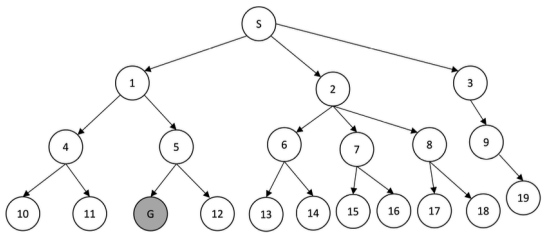


Figure 1: