Week 3:

Question 1: Artificial Intelligence

Q1. Implement an abstract two-player game (e.g., Tic-Tac-Toe or Connect-4) where one player is human and the other is controlled by an AI algorithm.

Task:

Solve the game using the following algorithms:

1. **Iterative Deepening Search (IDS)**
2. **Depth First Search (DFS)**

- **Requirements**:
  1. Design the game environment with a clear state representation and rules.
  2. Implement AI decision-making for the computer player using IDS and DFS.
  3. Compare the performance of the algorithms based on:
     - Solution quality (optimality of the moves).
     - Execution time (time to compute each move).
     - Explored states (number of nodes visited during the search).

Deliverables:

1. Python program implementing the game and the two algorithms.
2. A report comparing IDS and DFS based on:
   - Performance metrics (solution quality, execution time, explored states).
   - Insights on when one algorithm performs better than the other.
3. (Optional) Extend the program to support two AI players using IDS and DFS for decision-making.

Q2. Deep Learning Lab Exercise: GPU Performance for Fashion MNIST Dataset

Task:

Benchmark the performance of CPU versus GPU for training a deep learning model on the Fashion MNIST dataset. Specifically, compare the training times and performance metrics for models with:

1. One hidden layer.
2. Five hidden layers.

Objective:

- Understand the computational advantages of using GPUs over CPUs for deep learning tasks.
- Analyze the impact of model complexity (hidden layers) on training time and performance.
- Fill in the provided table with the benchmarked performance metrics.

Deliverables: Submit the code showcasing below criteria

1. Fill out the table with benchmarked training times.
2. Provide a brief report analyzing:
3. The difference in training times for CPU versus GPU.
4. The effect of increasing hidden layers on training time for both CPU and GPU.
5. Recommendations on when to use a GPU for training deep learning models.