**Date:29.10.25**

**TASK:12**

**Implementation of multi-agent decision system**

**CO4, CO5 S3**

**PROBLEM STATEMENT**

In modern decision-making systems, multiple autonomous agents collaborate to reach optimal or consensus-based outcomes. The challenge lies in designing a system where agents can communicate, share knowledge, and make independent yet coordinated decisions to achieve a common goal.

**AIM**

To design and implement a Multi-Agent Decision System that allows multiple intelligent agents to cooperate and make optimal decisions based on predefined rules and dynamic data.

**OBJECTIVE**

1. To study the concept of agent-based systems.

2. To develop a simple simulation of multiple agents making a joint decision.

3. To demonstrate communication and coordination among agents.

4. To analyze collective decision accuracy and performance

**DESCRIPTION**

A Multi-Agent Decision System (MADS) consists of multiple agents that can perceive their environment, communicate with one another, and act autonomously to achieve shared or individual objectives. These systems are widely used in distributed AI, robotics, and decision-support applications. Each agent can evaluate possible actions and contribute to the final group decision using voting or negotiation strategies

**ALGORITHM**

Step 1: Define agents with unique preferences or objectives.

Step 2: Initialize a set of possible decisions.

Step 3: Allow each agent to evaluate decisions based on internal logic or data.

Step 4: Agents communicate their preferences or votes.Step 5: Aggregate decisions using a consensus or voting mechanism.

Step 6: Display the final decision result.

Step 7: Evaluate system performance and decision accuracy.

**PROGRAM**

# Python program for a simple Multi-Agent Decision System

import random

class Agent:

def \_\_init\_\_(self, name):

self.name = name

def decide(self, options):

return random.choice(options)

# Create agents

agents = [Agent("Agent1"), Agent("Agent2"), Agent("Agent3"), Agent("Agent4")]

# Possible decisions

decisions = ["Option A" , "Option B" , "Option C"]

votes = {}

# Each agent votes

for agent in agents:

choice = agent.decide(decisions)

votes[choice] = votes.get(choice, 0) + 1

print(f"{agent.name} voted for {choice}")

# Determine the final decision

final\_decision = max(votes, key=votes.get)

print("\nFinal Decision based on majority vote:" , final\_decision)

**OUTPUT**

Agent1 voted for Option B

Agent2 voted for Option A

Agent3 voted for Option B

Agent4 voted for Option C

Final Decision based on majority vote: Option B

**CONCLUSION**

The Multi-Agent Decision System successfully demonstrated collaborative decision-making through multiple autonomous agents. The agents were able to independently select options and reach a majority-based decision. This experiment shows how distributed systems can enhance decision efficiency and accuracy in real-world applications such as autonomous vehicles, smart grids, and intelligent simulations.