EXPERIMENT 2

Implementation of Greedy Agent

Code

import numpy as np

import matplotlib.pyplot as plt

n\_arms = 3

true\_rewards = [0.35, 0.70, 0.105]

n\_steps = 100

reward\_sums = np.zeros(n\_arms)

counts = np.zeros(n\_arms)

estimated\_rewards = np.zeros(n\_arms)

reward\_history = []

cumulative\_reward = []

for arm in range(n\_arms):

reward = int(np.random.rand() < true\_rewards[arm])

reward\_sums[arm] += reward

counts[arm] += 1

estimated\_rewards[arm] = reward\_sums[arm] / counts[arm]

reward\_history.append(reward)

cumulative\_reward.append(sum(reward\_history))

for step in range(n\_arms, n\_steps):

arm = np.argmax(estimated\_rewards)

reward = int(np.random.rand() < true\_rewards[arm])

reward\_sums[arm] += reward

counts[arm] += 1

estimated\_rewards[arm] = reward\_sums[arm] / counts[arm]

reward\_history.append(reward)

cumulative\_reward.append(cumulative\_reward[-1] + reward)

print("\n=== Final Results ===")

print("True Reward Probabilities: ", true\_rewards)

print("Estimated Reward Probabilities: ", np.round(estimated\_rewards, 2))

print("Number of times each arm was selected:", counts)

print("Total Reward Earned: ", int(sum(reward\_history)))

plt.figure(figsize=(10, 5))

plt.plot(cumulative\_reward, label="Cumulative Reward", color='blue')

plt.xlabel("Steps")

plt.ylabel("Total Reward")

plt.title("Greedy Agent - Reward Over Time")

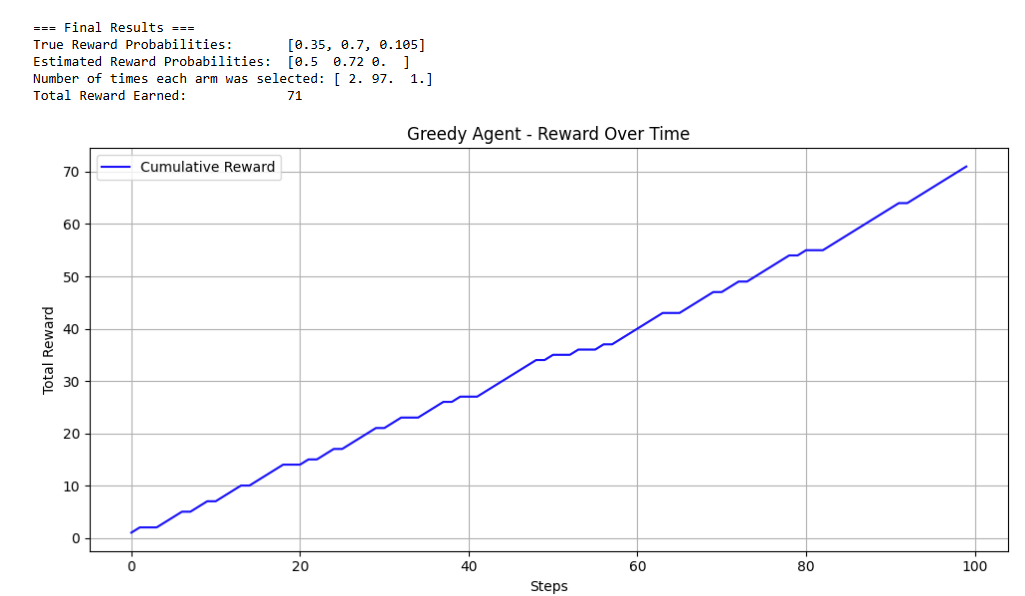
plt.grid(True)

plt.legend()

plt.tight\_layout()

plt.show()

Output



**MUGESH K**

**E0123038**