In [5]:
```
# Name : Thokala Kavyasree
# Roll No: 25201318

# ---------------- Parameters ----------------
mu = 10        # Mean (μ) given
phi1 = 0.5     # Coefficient for ε(t-1)
phi2 = 0.3     # Coefficient for ε(t-2)
error = [-2, 1, 0, 2, 1]   # Error terms (εt values)
n = len(error) # Number of time steps
# ---------------- MA(1) ----------------
print("----- MA(1) -----")
print("t\t f̂t (MA1)\t εt\t ft (MA1)")
j_pred = []  # list to store predicted values (f̂t)
j_act = []   # list to store actual values (ft)
# Loop over time steps
for i in range(n):
    if i == 0:
        # For t=1, prediction uses only μ
        f_hat = mu
    else:
        # For t>1, prediction: f̂t = μ + φ1 * ε(t-1)
        f_hat = mu + phi1 * error[i-1]
    # Actual value: ft = f̂t + εt
    f_t = f_hat + error[i]
    # Save values in lists
    j_pred.append(f_hat)
    j_act.append(f_t)
    # Print row of the table
    print(f"{i+1}\t {f_hat:.2f}\t\t {error[i]}\t {f_t:.2f}")
# ---------------- MA(2) ----------------
print("\n----- MA(2) -----")
print("t\t f̂t (MA2)\t εt\t ft (MA2)")
j_pred_2 = []  # list to store predicted values (f̂t)
j_act_2 = []   # list to store actual values (ft)
# Loop over time steps
for i in range(n):
    if i == 0:
        # For t=1, prediction uses only μ
        f_hat = mu
    elif i == 1:
        # For t=2, prediction uses μ + φ1 * ε(t-1)
        f_hat = mu + phi1 * error[i-1]
    else:
```

```
            # For t>=3, prediction: f̂t = μ + φ1 * ε(t-1) + φ2 * ε(t-2)
            f_hat = mu + phi1 * error[i-1] + phi2 * error[i-2]
        # Actual value: ft = f̂t + φ1ε(t-1) + φ2ε(t-2) + εt
        f_t = f_hat \
              + (phi1 * error[i-1] if i-1 >= 0 else 0) \
              + (phi2 * error[i-2] if i-2 >= 0 else 0) \
              + error[i]
        # Save values in lists
        j_pred_2.append(f_hat)
        j_act_2.append(f_t)
        # Print row of the table
        print(f"{i+1}\t {f_hat:.2f}\t\t {error[i]}\t {f_t:.2f}")
# ---------------- Moving Averages ----------------
# Moving Average of predicted values from MA(1)
n1 = int(input("\nEnter time upto 5 till which you need moving average (MA1): "))
mov_avg = sum(j_pred[:n1]) / n1   # average of first n1 predicted values
print(f"Moving Average_1 ({n1}) is: {mov_avg:.2f}")
# Moving Average of predicted values from MA(2)
n2 = int(input("Enter time upto 5 till which you need moving average (MA2): "))
mov_avg_2 = sum(j_pred_2[:n2]) / n2   # average of first n2 predicted values
print(f"Moving Average_2 ({n2}) is: {mov_avg_2:.4f}")
```

```
----- MA(1) -----
t        f̂t (MA1)        εt      ft (MA1)
1        10.00           -2      8.00
2        9.00            1       10.00
3        10.50           0       10.50
4        10.00           2       12.00
5        11.00           1       12.00

----- MA(2) -----
t        f̂t (MA2)        εt      ft (MA2)
1        10.00           -2      8.00
2        9.00            1       9.00
3        9.90            0       9.80
4        10.30           2       12.60
5        11.00           1       13.00

Enter time upto 5 till which you need moving average (MA1): 4
Moving Average_1 (4) is: 9.88
Enter time upto 5 till which you need moving average (MA2): 4
Moving Average_2 (4) is: 9.8000
```

In [ ]:

In [ ]:

```

```