Họ và tên: Đinh Văn Quang
Lớp: 20DT1
Lớp HP: 20.38
MSSV: 106200032
STT: 11

HOMEWORK 3

================================

Problem 1: Consider one-dimensional Kalman Filter without process noise to solve the following problem.
Assume that we would like to estimate the height of a building using an imprecise altimeter. We know that building height doesn't change over time, at least during the short measurement process. Thus, the dynamic model is given by

$$\hat{x}_{n+1,n} = \hat{x}_{n,n},$$

$$p_{n+1,n} = p_{n,n}.$$

The true building height is 50 meters. The altimeter measurement error (standard deviation) is 5 meters. The ten measurements are: 49.03m, 48.44m, 55.21m, 49.98m, 50.6m, 52.61m, 45.87m, 42.64m, 48.26m, 55.84m.
Given the initial guess as follows:

$$\hat{x}_{0,0} = 60m,$$

$$p_{0,0} = 225.$$

(a) Write a MATLAB program to find the state estimates and estimate uncertainties.

```
true_height = 50;                    % True building height (meters)
measurement_std_dev = 5;                 % alimeter measurement error (standard deviation)
(meters)
measurements = [49.03, 48.44, 55.21, 49.98, 50.6, 52.61, 45.87, 42.64, 48.26, 55.84];

% Kalman Filter initialization with given initial guess
num_measurements = length(measurements);
x_est = 60;                          % Initial estimate of the building height
p_est = 225;                          % Initial uncertainty in the estimate

% Measurement noise (variance) from standard deviation
r = measurement_std_dev^2;

% Initialize arrays to store estimates
x_estimates = zeros(1, num_measurements);
uncertainties = zeros(1, num_measurements);

% Kalman Filter loop
for n = 1:num_measurements
   % Measurement updat
   Kn = p_est / (p_est + r);                 % Calculate Kalman Gain
```

```
    x_est = x_est + Kn * (measurements(n) - x_est); % Update estimate with measurement
    p_est = (1 - Kn) * p_est;                  % Update uncertainty

    % results
    x_estimates(n) = x_est;
    uncertainties(n) = p_est;
end

% Display the results
disp('State estimates and estimate uncertainties after each measurement:');
disp(table((1:num_measurements)', measurements', x_estimates', uncertainties', ...
    'VariableNames', {'Measurement_No', 'Measurement', 'Estimate', 'Uncertainty'}));
```

Results in MATLAB:

```
Command Window
>> q_hw3
State estimates and uncertainties after each measurement:
    Measurement_No    Measurement    Estimate    Uncertainty
    ______________    ___________    ________    ___________

           1             49.03        50.127         22.5
           2             48.44        49.328       11.842
           3             55.21        51.219       8.0357
           4             49.98        50.917       6.0811
           5              50.6        50.855       4.8913
           6             52.61        51.142       4.0909
           7             45.87        50.401       3.5156
           8             42.64        49.444       3.0822
           9             48.26        49.314       2.7439
          10             55.84         49.96       2.4725

fx >>
```

(b) Fill in the state estimates and estimate uncertainties in the following table.

n	1	2	3	4	5	6	7	8	9	10
z_n	49.03m	48.44m	55.21m	49.98m	50.6m	52.61m	45.87m	42.64m	48.26m	55.84m
$\hat{x}_{n,n}$	50.127	49.328	51.219	50.917	50.855	51.142	50.401	49.444	49.314	49.96
$p_{n,n}$	22.5	11.842	8.0357	6.0811	4.8913	4.0909	3.5156	3.0822	2.7439	2.4725

(c) Plot the true value, measured values, estimates, and 95% confidence intervals.
Code in MATLAB:

```
true_height = 50;                    % True building height (meters)
measurement_std_dev = 5;             % alimeter measurement error (standard deviation)
(meters)
measurements = [49.03, 48.44, 55.21, 49.98, 50.6, 52.61, 45.87, 42.64, 48.26, 55.84];

% Kalman Filter initialization with given initial guess
num_measurements = length(measurements);
x_est = 60;                          % Initial estimate of the building height
p_est = 225;                         % Initial uncertainty (variance) in the estimate

% Measurement noise (variance) from standard deviation
r = measurement_std_dev^2;

% Initialize arrays to store estimates
x_estimates = zeros(1, num_measurements);
uncertainties = zeros(1, num_measurements);
confidence_interval = zeros(1, num_measurements);

% Kalman Filter loop
for n = 1:num_measurements
   % Measurement update
   Kn = p_est / (p_est + r);              % Calculate Kalman Gain
   x_est = x_est + Kn * (measurements(n) - x_est); % Update estimate with measurement
   p_est = (1 - Kn) * p_est;              % Update uncertainty

   % results
   x_estimates(n) = x_est;
   uncertainties(n) = p_est;
   confidence_interval(n) = 1.96 * sqrt(p_est);       % 95% confidence interval
end

% Display the results
disp('State estimates and uncertainties after each measurement:');
disp(table((1:num_measurements)', measurements', x_estimates', uncertainties', ...
   'VariableNames', {'Measurement_No', 'Measurement', 'Estimate', 'Uncertainty'}));

% Plot the results
figure;
hold on;
plot(1:num_measurements, measurements, 'x', 'DisplayName', 'Measured Values'); %
Measured values
plot(1:num_measurements, x_estimates, '-o', 'DisplayName', 'Estimated Height'); % Estimates
```

```
plot(1:num_measurements, true_height * ones(1, num_measurements), '--r', 'DisplayName',
'True Height'); % True height

% Plot 95% confidence intervals
fill([1:num_measurements, num_measurements:-1:1], ...
    [x_estimates + confidence_interval, fliplr(x_estimates - confidence_interval)], ...
    'blue', 'FaceAlpha', 0.3, 'EdgeColor', 'none', 'DisplayName', '95% Confidence Interval');

xlabel('Measurement Number');
ylabel('Height (m)');
title('Kalman Filter Estimated Height with 95% Confidence Intervals');
legend;
hold off;
```

Plot results in MATLAB: