University Larbi Ben M'hidi - Oum El Bouaghi-

Faculty of Exact Sciences, Natural and Life Sciences

Department of Mathematics and Computer Science

Exam : Programming Tools 2 **Duration: 1 hour 30 minutes**

Exercise 1 (5 points):

Provide the MATLAB commands that allow you to:

1. Define the polynomial $P(x) = -x^3 + x^2 - 2$ and $Q(x) = x^2 - x$.
2. Decompose $\frac{P}{Q}$ into partial fractions.
3. Define the vector V whose components range from -2 to 2 in steps of 0.05.
4. Evaluate the polynomials P and Q at the points in V.
5. Plot the curves of the polynomials P and Q over the interval $[-2,2]$ side by side. Use two different colors to distinguish the two curves, add a grid, a legend, and a title for each curve.

Exercise 2 (3 points) : Write a program that calculates an electricity bill based on units consumed:

Input: number of units consumed in a month
Price per unit:
First 100 units: 2 AD per unit
Next 200 units (101–300): 3 AD per unit
Above 300 units: 5 AD per unit
Display: units consumed and final bill.

Exercise 3 (5 points): Use symbolic computation to:

1. Compute the integral: $\int_{-\infty}^{+\infty} \ln(x^2 + 1)dx$
2. Compute the Taylor series expansion of cos(x) around 0 up to the 8th order.
3. Solve the system of equations: $3x + 4y = 10$ and $2x - y = 1$
4. Solve the differential equation: $y'' + 2y' + 5y = 0$,
   with initial conditions y(0) = 2 $y'(0) = 1$

Exercise 4(2 points):

Plot the 3D surface given by: Z=$\sqrt{x^2 + y^2}$ for $x \in [-5, 5]$, and $y \in [-3, 3]$.

Exercise 5 (5 points):

Given D=[1 2; 3 4], provide the MATLAB commands that allow you to :

1. Use the end operator to extract the last column.
2. Replace all elements less than 3 with 0.
3. Convert linear index 3 to classical indexing (row/column).
4. Use repmat to create a 6×4 matrix from D .
5. Use kron to create a 4×6 matrix.

Solutions for Programming Tools 2 Exam

Exercise 1 (5 points)

```
%% 1.1 Define the polynomials
P = [-1 1 0 -2    0.25
Q = [1 -1 0];       0.25

%% 1.2 Decompose P/Q into partial fractions
 [r, p, k] = residue( P ,  Q);    0. 5
%% 1.3 Define vector V
V = -2:0.05:2;    0.25
%% 1.4 Evaluate polynomials at points in V
P_vals = polyval(P, V);   0.25
Q_vals = polyval(Q, V);  0.25
%% 1.5 Plot the curves side by side
subplot(1,2,1);     0. 5
plot(V, P_vals, 'b-');  0. 5
grid on;  0.25
title('Curve of P(x)); 0.25
legend('P(x)');  0.25
subplot(1,2,2);   0. 5
plot(V, Q_vals, 'r-');  0. 5
grid on;
 title('Curve of Q(x) ');  0.25
legend('Q(x)');  0.25
```

Exercise 2 (3 points)

```
%% Program to calculate electricity bill
units = input('Enter the number of units consumed this month: '); 0.25
    % Calculate bill based on conditions
    if units <= 100  0.25
        bill = units * 2; 0.25
    elseif units <= 300  0.25
        bill = 100 * 2 + (units - 100) * 3;  0. 5
    else  0.25
        bill = 100 * 2 + 200 * 3 + (units - 300) * 5; 0.5
    end  0.25
    % Display results
    fprintf('Units consumed: %d units\n', units);  0.25
    fprintf('Total bill: %d Algerian Dinars\n', bill);  0.25
```

Exercise 3 (5 points)

```
%% 3.1 Compute the integral
syms x 0.25
```

```
integral1 = int(log(x^2 + 1),x, -inf, inf);  1pts
syms x 0.25
%% 3.2 Taylor series expansion of cos(x)
taylor_cos = taylor(cos(x), x, 8);  1pts
%% 3.3 Solve the system of equations
syms x y  0.25
[a,b] = solve(['3*x + 4*y = 10', '2*x - y = 1', x, y); 1pts


%% 3.4 Solve the differential equation
syms y ,x  0.25

sol = dsolve('D2y + 2*Dy + 5*y=0',' y(0) = 2, Dy(0) = 1','x'); 1pts
```

Exercise 4 (2 points)

```
%% Plot the 3D surface
x = linspace(-5, 5, 100);  0.25
y = linspace(-3, 3, 100);  0.25

[X, Y] = meshgrid(x, y);  0.5

Z = sqrt(X.^2 + Y.^2);  0.5
surf(X, Y, Z);  0.5
```

Exercise 5 (5 points)

```
%% 5.1 Use end operator to extract last column
last_column = D(:, end);  0.5pts

%% 5.2 Replace all elements less than 3 with 0
D_modified(D < 3) = 0; 0.5pts

%% 5.3 Convert linear index 3 to row/column indexing
[ind_row, ind_col] = ind2sub(size(D), 3);  1pts

%% 5.4 Use repmat to create 6×4 matrix
D_repmat = repmat(D, 3, 2);  1.5pts


%% 5.5 Use kron to create 4×6 matrix
D_kron = kron(D, ones(2, 3)); 1.5pts
```