CS 352 - ASSIGNMENT 2

Arnav Jain - 220002018

MCQ Answers:

Question 1:

Answer: Option A) Creates a vector of 5 equally spaced points between 1 and 10.

Question 2:

Answer: Option B) A .* B

Question 3:

Answer: Option B) The solution to the linear system A×C=B.

Question 4:

Answer: Option B) unique(v)

Question 5:

Answer: Option B) Equal to 3

Coding Problems

Question 1

Code:

q1.m

/MATLAB Drive/CG LAB 2/q1.m

```
% Answering part a - Creating a vector from 1 to 20 with an increment of 2.
vector = 1:2:20;
disp('Vector from 1 to 20 with an increment of 2:');
disp(vector);

% Answering part b - Finding and printing the square of each element in the vector.
squared_vector = vector.^2;
disp('Square of each element in the vector:');
disp(squared_vector);

% Answering part c - Extracting and printing all even numbers from the vector.
even_numbers = vector(mod(vector, 2) == 0);
disp('Even numbers from the vector:');
disp(even_numbers);
```

Result:

Command Window

```
>> q1
Vector from 1 to 20 with an increment of 2:
     1     3     5     7     9    11    13    15    17    19

Square of each element in the vector:
     1     9    25    49    81   121   169   225   289   361

Even numbers from the vector:
>>
```

Question 2

Code:

q2.m

/MATLAB Drive/CG LAB 2/q2.m

```
% Answering part a - Defining a 3x3 matrix A with random integers.
A = randi([1, 10], 3, 3);
disp('Matrix A:');
disp(A);

% Answering part b - Calculating the determinant of A.
det_A = det(A);
disp('Determinant of A:');
disp(det_A);

% Answering part c - Finding and printing the transpose of A.
transpose_A = A.';
disp('Transpose of A:');
disp(transpose_A);

```

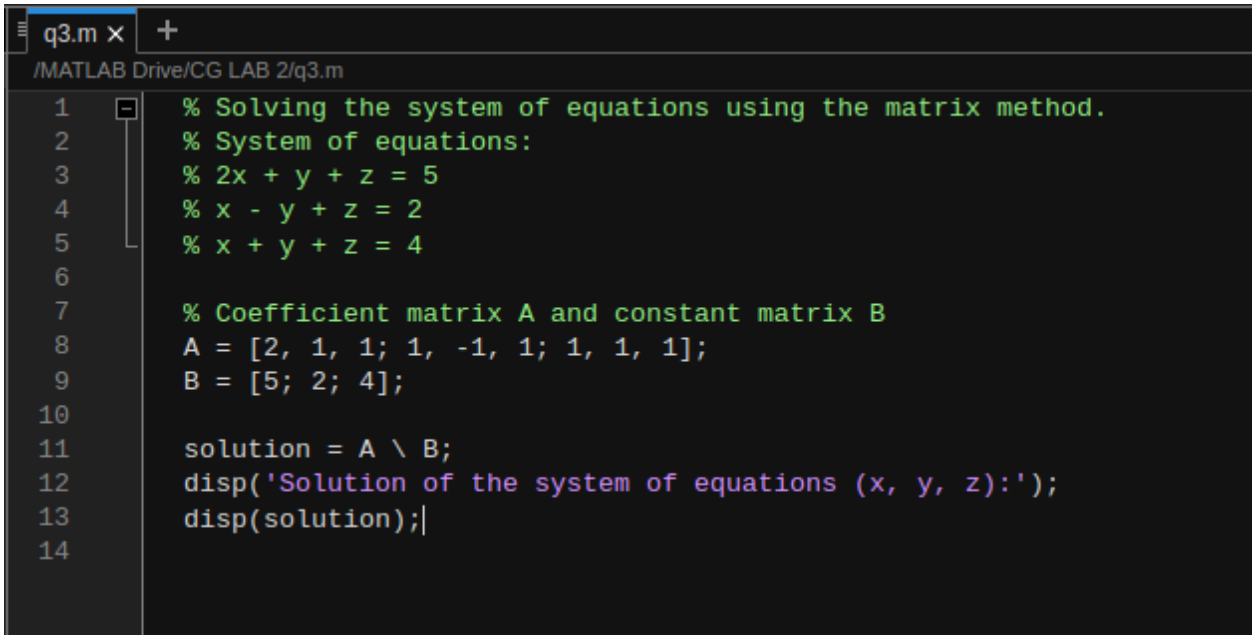
Result:

Command Window

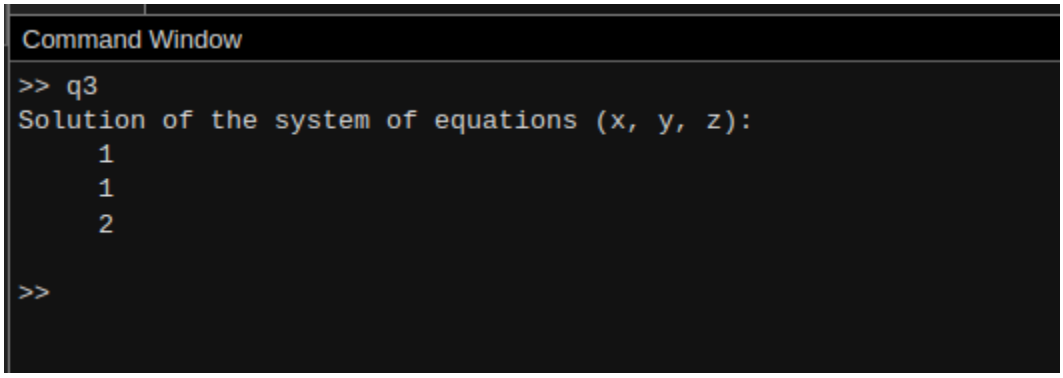
```
>> q2
Matrix A:
     9    10     3
    10     7     6
     2     1    10

Determinant of A:
  -316

Transpose of A:
     9    10     2
    10     7     1
     3     6    10

>>
```

Question 3

Code:



Result:



Question 4

Code:

q4.m

/MATLAB Drive/CG LAB 2/q4.m

```
% Writing a script to take a number as input from the user
% and printing whether the number is positive, negative, or zero.

number = input('Enter a number: ');

if number > 0
    disp('The number is positive.');
elseif number < 0
    disp('The number is negative.');
else
    disp('The number is zero.');
end

```

Result:

Command Window

```
>> q4
Enter a number:
23
The number is positive.
>> q4
Enter a number:
-12
The number is negative.
>> q4
Enter a number:
0
The number is zero.
>>
```

Question 5

Code:

q5.m

/MATLAB Drive/CG LAB 2/q5.m

```
% Answering part a - Representing the polynomial 3x^2 + 2x + 1 as a vector.
poly = [3, 2, 1];  % Coefficients of 3x^2 + 2x + 1
disp('Polynomial:');
disp('3x^2 + 2x + 1');

% Answering part b - Finding the derivative of the polynomial and printing it.
derivative_poly = poly(1:end-1) .* (length(poly)-1:-1:1);
disp('Derivative of the polynomial:');
disp(derivative_poly); % 6x + 2

% Answering part c - Calculating and displaying the value of the polynomial at x=5.
x = 5;
poly_value = polyval(poly, x);
disp(['Value of the polynomial at x = 5: ', num2str(poly_value)]);

```

Result:

Command Window

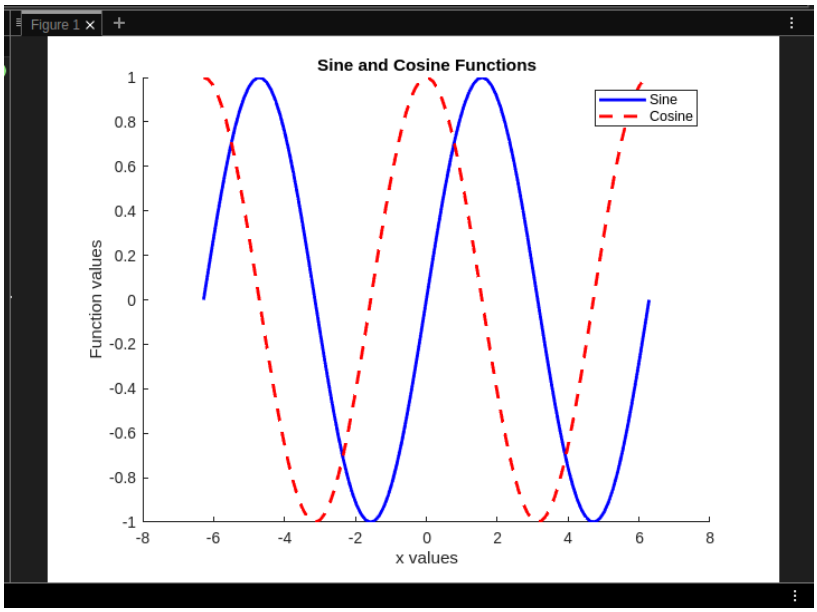
```
>> q5
Polynomial:
3x^2 + 2x + 1
Derivative of the polynomial:
     6     2

Value of the polynomial at x = 5: 86
>>
```

Question 6

Code:

```
% Answering part a, b, and c - Plotting sine and cosine functions
% for x values between -2π and 2π with different colours, styles, legend,

x = linspace(-2*pi, 2*pi, 100);

% Sine and cosine functions
y_sin = sin(x);
y_cos = cos(x);

figure;
hold on;

% Sine plot (blue line)
plot(x, y_sin, 'b-', 'LineWidth', 2);

% Cosine plot (red dashed line)
plot(x, y_cos, 'r--', 'LineWidth', 2);

% Adding title, labels, and legend
title('Sine and Cosine Functions');
xlabel('x values');
ylabel('Function values');
legend('Sine', 'Cosine');

hold off;
```

Result:



Question 7

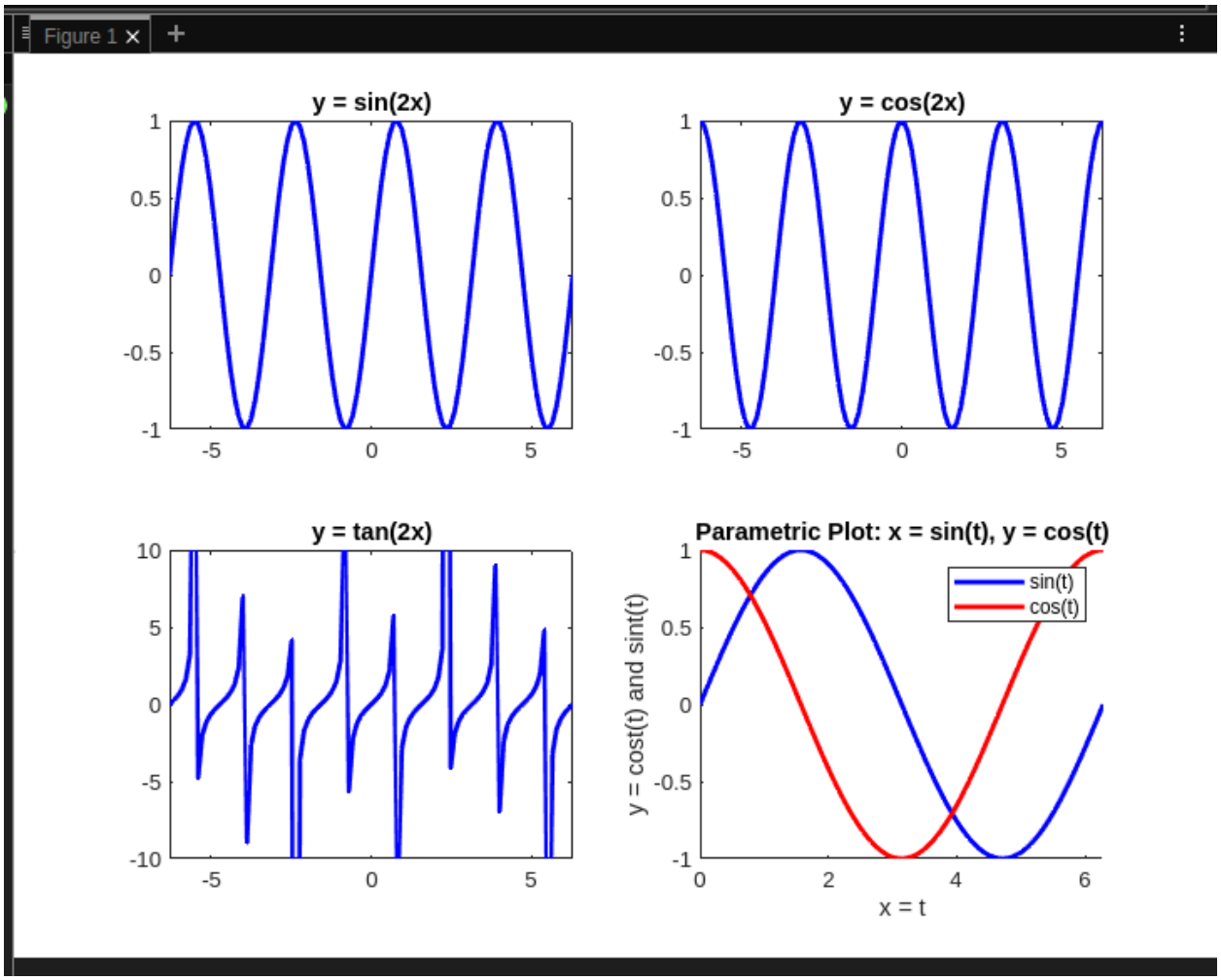
Code:

q7.m

/MATLAB Drive/CG LAB 2/q7.m

```
% Answering part a - Creating subplots for y = sin(2x), y = cos(2x), and y = tan(2x).
% Also, using the fourth subplot for a parametric plot of x = sin(t), y = cos(t) for t from 0 to 2π.

t = linspace(0, 2*pi, 100);
x = linspace(-2*pi, 2*pi, 100);
y_sin2x = sin(2*x);
y_cos2x = cos(2*x);
y_tan2x = tan(2*x);

figure;

% Subplot 1: y = sin(2x)
subplot(2, 2, 1);
plot(x, y_sin2x, 'b-', 'LineWidth', 2);
title('y = sin(2x)');

% Subplot 2: y = cos(2x)
subplot(2, 2, 2);
plot(x, y_cos2x, 'b-', 'LineWidth', 2);
title('y = cos(2x)');

% Subplot 3: y = tan(2x)
subplot(2, 2, 3);
plot(x, y_tan2x, 'b-', 'LineWidth', 2);
title('y = tan(2x)');
ylim([-10, 10]); % Limiting y-axis values to avoid extreme values of tan

% Subplot 4: Parametric plot of x = sin(t), y = cos(t)
subplot(2, 2, 4);
hold on;
plot(t, sin(t), 'b-', 'LineWidth', 2); % Plot sin(t) in blue
plot(t, cos(t), 'r-', 'LineWidth', 2); % Plot cos(t) in red
title('Parametric Plot: x = sin(t), y = cos(t)');
xlabel('x = t');
ylabel('y = cost(t) and sint(t)')
legend('sin(t)', 'cos(t)');
hold off;

```

Result:



For Code , refer GitHub:

https://github.com/arnavjain2710/Computer-Graphics-Lab/tree/main/LAB%202