Name: **Enrolment No:**	UPES UNIVERSITY OF TOMORROW

UPES

End Semester Examination, May 2024

Course: Introduction to MATLAB Programming **Semester:** IV
Program: B.Sc. Physics **Time:** 03 hrs.
Course Code: MATH 2034K **Max. Marks:** 100

Instructions:

SECTION A
(5Qx4M=20Marks)

S. No.		Marks	CO
Q 1	**Define** different types of concatenation and **illustrate** its implementation in MATLAB. **Determine** the output of: `A = [10 20 30; 60 70 80]; A(3:4,4:5) = [2 3; 4 5]`	**2+1+1**	**CO1**
Q 2	**Discuss** any two ways to define colour while plotting in MATLAB. **Write** a MATLAB code to plot the complex roots of unity: $z^n = 1 \; \forall \; n$.	**2+2**	**CO2**
Q 3	**Define** mesh and surf plots in MATLAB. **Write** the MATLAB code to plot the function that characterizes the shape of the surface of water right at the moment a stone enters the water *(it is not just a simple sinusoidal function)* in three-dimensions.	**2+2**	**CO2**
Q 4	**Discuss** the output for the following two codes: i. $x = \text{linspace}(-5,5,100);\ y = \text{linspace}(-5,5,100);$ for $i = 1:4$ $Z = \sin(X+i) + \sin(Y+i);\ \text{surf}(X,Y,Z,'\text{FaceAlpha}', 0.2 * i);$ hold on; end ii. $x = \text{linspace}(-5,5,100);\ y = \exp(x);$ $\text{semilogx}(x,y);\ \text{semilogy}(x,y);$	**2+2**	**CO2**
Q 5	**Discuss** the residue command in MATLAB. **Write** a MATLAB code for determining the conditions when $\text{residue}(\text{residue}(b\_1, a\_1) + \text{residue}(b\_2, a\_2))$ is exactly equal to $\frac{b_1}{a_1} + \frac{b_2}{a_2}$, where $b(s)/b_i(s)$ and $a(s)/a_i(s)$ are polynomials of independent variable s. (Hint: Use the nested residue command for a single (b, a) and see what you get)	**2+2**	**CO3**

SECTION B
(4Qx10M= 40 Marks)

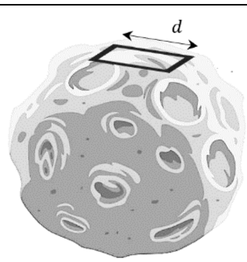
S. No.		Marks	CO
Q 6	**Explain** the difference between defining symbolic expressions using the syms function and the symfun function in MATLAB. Provide the MATLAB code for the function $f(x) = \tan^{-1}\left(\frac{1}{x+1}\right)$ using both.	**3+3+4**	**CO4**

	Describe how symbolic functions can be evaluated at specific points using the subs function in MATLAB. What will be the output of $\text{subs}(@(x)\ \tan(x), \text{pi}/2)$ and $\text{subs}(@(x,y)\exp(2*x+y), [3,4,5])$? **Illustrate** the usage of MATLAB's symbolic operations for differentiation and integration. Debug the following code: $\text{diff}\left(@(x)\ \exp\left(\exp\left(\exp(x^2)\right)\right)\right) + \text{diff}(@(x,y)\sin(13*y+pi*x), y) - \text{int}(@(x)\ 1/x\hat{}2, x\hat{}2,2,3)$ What would be the output of the corrected code?		
Q 7	**Define** two ways each, to undertake conditional and iteration statements in MATLAB. In a physics experiment, each particle exhibits motion with constant velocity, uniform acceleration, or simple harmonic motion (SHM). Given data representing particle positions at different time intervals, **write** a MATLAB code for classification of each particle's motion using a switch statement. If we had considered the dataset for a pendulum, while it primarily displays SHM, its acceleration due to gravity is approximately uniform near the equilibrium position. **Write** a MATLAB code using nested loops to determine such cases.	**4+4+2**	**CO3**
Q 8	**Write** a MATLAB code to simulate the motion of a simple pendulum consisting of a mass m attached to a string of length L in a uniform gravitational field, with the relevant parameters being gravitational acceleration g, initial angle θ_0, and initial angular velocity ω_0. In the program, **illustrate** with instances, numerical integration (using ode45) to solve the differential equation for the pendulum's motion for an assumed time interval, updating the angle and angular velocity at each time step. Plot in MATLAB θ vs. time and **discuss** how changes in the initial conditions can affect the pendulum's motion.	**5+5**	**CO2**
Q 9	**Discuss** the significance of the MATLAB command window and workspace window in a MATLAB session. **Describe** the process of defining variables using MATLAB, including the assignment operator and built-in elementary mathematical functions. **Illustrate** how can one manage workspace variables effectively using commands like who and whos. **Outline** the steps to quit a MATLAB session properly. **Explain** the process of creating arrays in MATLAB using built-in commands such as ones, eye, and zeros. **Discuss** arithmetic operations (addition, subtraction, multiplication) with arrays and the use of the transpose operation. **Illustrate** how to find the determinant, inverse, eigenvalues, eigenvectors, and rank of a matrix using built-in MATLAB commands. **Explore** indexing for	**5+5**	**CO1**

<table>
<tr><td></td><td>addressing array elements and the colon operator's role in array manipulation. Highlight built-in functions for handling array properties.

OR

Explain the structure and utility of script files in MATLAB, including the creation, saving, and execution of script files. Discuss the concept of MATLAB Path and the distinction between Global and Local variables. Explain how to create and utilize global variables, and demonstrate the usage of input, disp, and fprintf commands for output.

Describe the structure of a function file, including the function definition line, input and output arguments, H1 line, help text lines, and the function body. Discuss the handling of local and global variables within function files. Demonstrate the process of saving a function file and calling it from the command window. Provide examples of simple functions and compare function files to script files. Discuss the concept of inline functions, with a brief illustrative example.</td><td></td><td></td></tr>
<tr><td colspan="4">SECTION-C
(2Qx20M=40 Marks)</td></tr>
<tr><td></td><td>

Physics is often about formulating new laws based on observations that show divergence from behavior expected of established laws. In a hypothetical scenario, let us say, that in the study of an asteroid post-impact using interactions of samples with certain polycyclic aromatic hydrocarbons (PAHs), observations reveal a new property, Dramūlatva, stored in a file named observations.dat (that has entries indexed by an index number and sampling area dimension d with the other entries being average density, average granularity, average pH and average colour). Initial insights suggest that this property, as observed from rectangular cross-sections of dimensions $d \times 0.5d$ (in cm^2), can be modeled by a correlative multipeaked Gaussian function: $\Im(x) = 2.345 \times 10^{-6} \sum_i e^{-(x-x_i)^2}$ ð, where $n = d/10^7$ and x_i represents equispaced points along the longer dimension. Further analysis reveals a background contribution from bulk atoms, behaving as a polynomial divided by $z^{3/2}$, where z is the depth-variable from the surface. The net expression of the property is obtained by convolving the surface and bulk contributions.

1. Write a MATLAB code to read the observations file and efficiently organize the data into a three-dimensional array.
2. Write a MATLAB code-section to fit a polynomial of degree 10 to the data obtained from evaluating the expression from the initial insight at discrete number of points. Once all the polynomials are found for the</td><td>5+5+5+5</td><td>CO3</td></tr>
</table>

	various samples, take the average coefficient-vector, and store it as the net surface contribution for $\mathfrak{I}$. 3. You are given the average values of the background signals at $z = 0, 2, 4, \ldots, 2000$ nm for one depth-based analysis. It is seen that the numerical differences of the data-points vanish for the 7^{th} order difference. **Write** a MATLAB code-section to fit a polynomial for these values, and store it as the net bulk contribution for $\mathfrak{I}$. 4. **Write** a MATLAB code-section to convolve the two contributions and plot the expression as a function of distance.		

	a. **Define** the syms command. **Discuss** how we can work with symbolic variables, symbolic expressions, symbolic algebra, symbolic summation, symbolic Taylor series expansion and symbolic calculus. **Illustrate** with an example.
	b. **Discuss** the errors in the following code, correct it and give the output of the corrected code (don’t change the absence/presence of the semicolon at the end):
	`syms x, y;`
	`expr = sum(sin(x)^2, cos(x)^2);`
	`simplified_expr = simplify(@(x) expr)`
	`substituted_expr = sub(simplified_expr, x, pi/4)`
	`d_expr = diff(simplified_expr, x, h=0.001)`
	`integrated_expr = integ(simplified_expr, x)`
	`vars_in_expr = symvars(integrated_expr);`
	
	OR
	b. **Discuss**, with MATLAB based examples, the following concepts:
	i. Piecewise definition of a function
	ii. Symbolic matrices
	iii. Function plotting across scales
	iv. Solving non-linear equations
	v. Enhanced readability of mathematical expressions
	vi. Matrix left division
		8+12	**CO4**