IT Workshop Lab

COURSE CODE : 18B17CI372

COURSE CREDITS : 2

CORE/ELECTIVE : CORE

SEMESTER : IIIrd Semester

SESSION : Jul-Dec, 2019

LAST UPDATED ON : 17-04-2019

FACULTY COORDINATORS:

L-T-P: 0-0-4

Pre-requisite: Basic Programming knowledge like (variable declaration, initialization, decision construct, loops, Array and structure definition, and procedures). Basic understanding of graphs types and formation

Course Objectives:

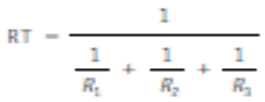
1. To introduce the students with the basic features of Matlab/Scilab for problem solving
2. To introduce the students about the Mathematical functions like matrix generation and Plotting with multiple data sets, line styles and colors.
3. To introduce the students about the Array operations and solving Linear equations in Matlab/Scilab .
4. To introduce the students about the control flow and operators using if-end structures and loops.
5. To introduce the students about the writing M-file scripts and Debugging M-files

Course Outcomes:

S.No.	Course Outcomes	Level of Attainment
CO1	Writing fundamental programs in Matlab/Scilab, creating variables and mathematical functions	Familiarity
CO2	Programming the fundamentals concepts of basic Plotting consisting of simple and multiple data sets in one plot	Usage
CO3	Understand how to program matrix operations, array operations and how to solve the system of linear equations	Assessment

CO4	Understand how to program M-file scripts, M-file functions, Input –output Arguments	Assessment
CO5	Program control flow operators, loops, flow structures and debugging M-files	Assessment

List of Experiments

S. No	Description	Hours
1	1. Create variable, pounds, to store a weight in pounds. Convert this to kilograms and assign the result to variable kilos. The conversion factor is 1 kilogram = 2.2 pounds. 2. The combined resistance RT of three resistors R1, R2, and R3 in parallel is given by  Create variables for the three resistors and store values in each, and then calculate the combined resistance.	**2**
2	Write a MATLAB/SCILAB program to calculate the following expression and round the answers to the nearest integer. a) z = 5x2 + y2 where x=2, y=4 b) z = 4cos(x)+ j6sin(x) where x=π/4 c) z = 3sin(x)+4cos(x)+3ey where x=π/3 , y=2 d) y=sin(x) / x where $0 \leq x \leq 2\pi$ 2. Solve the following system x + y - 2z = 3 2x + y = 7 x + y - z = 4	**2**
3	1. Write a program for three bits parity generator using even-parity bit. 2. Write a program to convert a three bits binary number into its equivalent gray code. 3. if q=[1 5 6 8 3 2 4 5 9 10 1],x=[3 5 7 8 3 1 2 4 11 5 9], then:	**2**

a) find elements of (q) that are greater than 4.

b) find elements of (q) that are equal to those in (x).

c) find elements of (x) that are less than or equal to 7.

4. If x=[10 3 ; 9 15], y=[10 0; 9 3], z=[-1 0; -3 2], what is the output of the following statements:

a) v = x > y

b) w = z >= y

c) u = ~z & y

d) t = x & y < z

4 1.Plot sin(x) on the interval [-pi,pi] using spacing 0.5, 0.1 and 0.01 between the points where you will sample the function. (This will change the resolution). Experiment with the hold on command. **2**

2. Attach labels to the axis of the previous plot and give a title to the graph.

3. Plot 5 cos(x2+1) on [-2pi,2pi]. Note that the squaring operation will require you to use the dot . in order for the squaring operation to act on each element individually. However, the addition operation (+) automatically acts on elements individually.

5 1. Type x=[1 2 3] **2**

y =[4 5 6]

a=2

x+y

x-y

a*x

and observe what happens.

If want to apply an operation such as squaring each element in a matrix we have to use a dot . before the operation we wish to apply. Type the following commands in MATLAB/SCILAB .

x=1:10

x.^2

A=[1 2 3 ; 4 5 6 ; 7 8 9]

A.^2

A^2

and observe the result. The dot allows us to do operations element wise.

All built-in functions such as sin, cos, exp and so on automatically act elementwise on a matrix. Type

y=[0 1/4 1/2 3/4 1]

y=pi*y

sin(y)

and observe the result.

2. Create a array x with 10 elements x=[1 2 3 4 5 6 7 8 9 10]

We can also create this vector by typing x=1:10. The vector (1 1.1 1.2 1.3 1.4 1.5) can be created by typing x=[1 1.1 1.2 1.3 1.4 1.5] or by typing x=1:0.1:1.5.

Matrices can be created according to the following example. The matrix A= $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$ is created by typing

A=[1 2 3 ; 4 5 6; 7 8 9],

i.e., rows are separated with semi-colons. If we want to use a specific element in a vector or a matrix, study the following example:

Example:

x=[10 20 30]

A=[1 2 3 ; 4 5 6 ; 7 8 9]

x(2)

A(3,1)

Here we extracted the second element of the vector by typing the variable and the position within parentheses. The same principle holds for matrices; the first number specifies the row of the matrix, and the second number specifies the column of the matrix. Note that in MATLAB/SCILAB the first index of a vector or matrix starts at 1, not 0 as is common with other programming languages.

6 1. A Pythagorean triple is a set of positive integers (a,b,c) such that a2 + b2 = c2. Write a function ispythag that will receive three positive integers (a, b, c in that order) and will return 1 for true if they form a Pythagorean triple, or 0 for false if not. **2**

2. Whether a storm is a tropical depression, tropical storm, or hurricane is determined by the average sustained wind speed. In miles per hour, a storm is a tropical depression if the winds are less than 38 mph. It is a tropical storm if the winds are between 39 and 73 mph, and it is a hurricane if the wind speeds are > = 74 mph. Write a script that will

prompt the user for the wind speed of the storm, and will print which type of storm it is.

7 **2**

1. Write a script that will prompt the user for N integers, and then write the positive numbers (>= 0) to an ASCII file called pos.dat and the negative numbers to an ASCII file called neg.dat. Error-check to make sure that the user enters N integers.

2. Write a script that will continue prompting the user for positive numbers, and storing them in a vector variable, until the user types a negative number.

3. Write a script that will use the menu function to present the user with choices for functions fix, floor, and ceil. Error-check by looping to display the menu until the user pushes one of the buttons (an error could occur if the user clicks the X on the menu box rather than pushing one of the buttons). Then, generate a random number and print the result of the user's function choice of that number (e.g., fix(5)).

8 **2**

1. Write a "currency exchange program" similar to the one in Example 1 which can handle two different exchange rates, exchange_rate1 = 0.5 and exchange_rate2 = 0.25. Design the program to first ask for the amount in dollars and then ask the user which rate (represented by the numbers 1 and 2 respectively) he/she wants. Let the program return the amount in the requested foreign currency.

2. Write a program that approximates PI by computing the sum $\pi \approx 4\sum_{k=0}^{m} \frac{(-1)^k}{2k+1}$.

The more terms you keep in the summation, the more accurate your answer will be. (In fact, the series converges to PI as m goes to infinity.) See how many terms you need to approximate PI with 5 decimals. (Note: This is by no means the most efficient way to approximate PI, but the formula is quite beautiful...)

3. Use the sum given in Exercise 2 to approximate PI using 10, 100, 1000, 10000 and 100000 terms. For each of these numbers, compute the error of the approximation. Plot the error as a function of the number of terms used in the sum.

9 **2**

1. In Europe daylight time starts on the last Sunday of March and ends on the last Sunday of October. Write a function that determines whether a given daynumber is in the summertime period or in the wintertime period of the Daylight Saving Time

2. Write a function that will receive the radius of a circle and will print both the radius and diameter of the circle in a sentence format. This function will not return any value; it simply prints

10 1. Write a program to calculate and print the area and circumference of a circle. There should be one script and three functions to accomplish this (one that prompts for the radius, one that calculates the area and circumference, and one that prints). **2**

2. The lump sum S to be paid when interest on a loan is compounded annually is given by S = P(1 + i)n, where P is the principal invested, i is the interest rate, and n is the number of years. Write a program that will plot the amount S as it increases through the years from 1 to n. The main script will call a function to prompt the user for the number of years (and error-check to make sure that the user enters a positive integer). The script will then call a function that will plot S for years 1 through n. It will use 0.05 for the interest rate and $10,000 for P.

11 1. Write a script that will prompt the user for a temperature in degrees Celsius, and then an F for Fahrenheit or K for Kelvin. The script will print the corresponding temperature in the scale specified by the user. For example, the output might look like this: Enter the temp in degrees C: 29.3 Do you want F or K? F The temp in degrees F is 84.7 The format of the output should be exactly as specified here. The conversions are: **2**

F = 9/5 *C+32

K = C+273.15

2. Write a function to calculate the volume and surface area of a hollow cylinder. It receives as input arguments the radius of the cylinder base and the height of the cylinder. The volume is given by Π r2 h, and the surface area is 2 Π r h.

3. Hurricanes are categorized based on the winds. The following table shows the category number for hurricanes with varying wind ranges and what the storm surge is (in feet above normal).

1	74–95	4–5
2	96–110	6–8
3	111–130	9–12
4	131–155	13–18
5	>155	>18

Write a function that will receive as an input argument the wind speed, and will return the category number and the minimum value of the typical storm surge.

12 1. Write a function called geomser that will receive values of r and n, and will calculate and return the sum of the geometric series: **2**

1 + r + r2 + r3 + r4 + ... + rn The following examples of calls to this function illustrate what the result should be:

>> geomser(1,5)

ans = 6

>> disp(geomser(2,4))

31

2. A sound engineer has recorded a sound signal from a microphone. The sound signal was sampled, meaning that values at discrete intervals were recorded (rather than a continuous sound signal). The units of each data sample are volts. The microphone was not on at all times, however, so that data samples below a certain threshold are considered to be data values that were samples when the microphone was not on, and therefore not valid data samples. The sound engineer would like to know the average voltage of the sound signal. Write a script that will ask the user for the threshold and the number of data samples, and then for the individual data samples. The program will then print the average and a count of the valid data samples, or an error message if there were no valid data samples. An example of what the input and output would look like in the Command Window is shown:

Please enter the threshold below which samples will be considered to be invalid:

3.0

Please enter the number of data samples to be entered: 7

Please enter a data sample: 0.4

Please enter a data sample: 5.5

Please enter a data sample: 5.0

Please enter a data sample: 2.1

Please enter a data sample: 6.2

Please enter a data sample: 0.3

Please enter a data sample: 5.4

The average of the 4 valid data samples is 5.53 volts

13 1. Create a vector of five random integers, each in the range from –10 to 10. Perform each of the following two ways: using built-in functions, and also using loops (with if statements if necessary): **2**

Subtract 3 from each element.

Count how many are positive.

Get the absolute value of each element.

Find the maximum.

2. Create a 3 × 5 matrix. Perform each of the following two ways: using built-in functions, and also using loops (with if statements if necessary):

Find the maximum value in each column.

Find the maximum value in each row.

Find the maximum value in the entire matrix.

3. Write a script that will print the following multiplication table:

1

2 4

3 6 9

4 8 12 16

5 10 15 20 25

14 1. Biomedical engineers are developing an insulin pump for diabetics. To do this, it is important to understand how insulin is cleared from the body after a meal. The concentration of insulin at any time t is described by the equation **2**

$C = C0\ e^{-30t/m}$ where C0 is the initial concentration of insulin, t is the time in minutes, and m is the mass of the person in kg. Write a script that will graphically show how the weight of the person influences the time for insulin to be cleared from the body. It will show in a 2 × 1 subplot the concentration of insulin for two subjects, one who weighs 120 pounds, and one who weighs 300 pounds. For both, the time should increment from 0 to 4 minutes in steps of 0.1 minute, and the initial concentration should be 85. The concentration over time will be shown in each subplot, and the weight of the person should be in the title. The conversion factor is 1 pound = 0.4536 kg. In order to better compare, use consistent axes for both plots.

2. Sales (in millions) from two different divisions of a company for the four quarters of 2006 are stored in vector variables, for example, div1 = [4.2 3.8 3.7 3.8];

div2 = [2.5 2.7 3.1 3.3];

Using subplot, show side-by-side the sales figures for the two divisions. What kind of graph shows this in the best way? Why? In one graph, compare the two divisions. What kind of graph shows this in the best way? Why?

15 1. For the following matrices A, B, and C: **2**

A	B	C
1 4	2 1 3	3 2 5
3 2	1 5 6	4 1 2
	3 6 0	

Which are symmetric?

For all square matrices, give their trace.

Give the result of 3*A.

Give the result of A*C.

Are there any other matrix multiplications that can be performed? If so, list them.

2. Given the following matrices:

A	B	C
3 2 1	2	1 0 0
0 5 2	1	0 1 0
1 0 3	3	0 0 1

Perform the following MATLAB/SCILAB operations, if they can be done. If not, explain why.

A * B

B * A

I +A

A .* I

trace(A)

16 1. Write a function issquare that will receive an array argument, and will return 1 for true if it is a square matrix, or 0 for false if it is not. **2**

2. Write a function mydiag that will receive an array argument, and will return a vector consisting of the main diagonal (without using the built-in diag function).

17 2. Write a function that will receive a square matrix as an input argument, and will return a row vector containing the diagonal of the matrix. If the function is called with a vector of two variables on the left-hand side of the assignment, the function will also return the trace of the matrix. (Note: It will return the trace only if there is two variables on the left-hand side of the assignment.) You may assume that the **2**