Experiment 1

Introduction to MATLAB

Question I.

- MATLAB stands for ***MATrix LABoratory***. It is developed by ***Mathworks***. It is an extremely useful tool for ***Scientists and Engineers***. It is a ***high-level language*** which provides environment for Numerical ***Computations and Visualizations***. Scripts are stored in ***'.m'*** format.

- MATLAB can be used as an advanced calculator. The below operators can be used:
  - \+ for addition
  - \- for subtraction
  - * for multiplication
  - / and \ for division. Example 5/4 = 1.25 and 5\4 = 0.8
  - ^ for power. Example 2^3 = 8

  Brackets can be used to give preference

  == stands for equals and ~= stands for not equals

  rem() function can be used for remainder. Example rem(5,4) = 1

  exp() function is used for exponential. Example exp(2) = 7.3891

  log() function is used for logarithmic (base e). Example log(10) = 2.3026

  cos(), sin(), tan(), asin(), acos() etc. functions are used for trigonometry.

- ***Arrays*** are ordered arrangement of numbers in MATLAB.
  For example, a vector can be thought of a 1-D array and matrix, a 2-D array.
  Hence, a 1-D array can be created as below:
  A = [n1 n2 n3 …… nm]
  A 3-D can be created as below:
  B = [n1 n2 n3;n4 n5 n6;n7 n8 n9] (i.e. use semi-colon for next row)

  ### *Matrix Algebra*

  - **Addition:** '+' Example: matrix1 + matrix2 or matrix1 + scalar

- **Subtraction:** Same as Addition (use of – instead of +)
- **Multiplication:** '*' when multiplication is possible(i.e., row dimension of one equals column dimension of other). Use '.*' when dimensions do not agree. Also known as element by element multiplication.
- **Division:** X = Same as multiplication (use of / and ./). Moreover, A\B solves for X in A*X = B and X = B/A solves for X in X*A = B.

- **2-D Plotting:** After declaring syms x (x as a symbol), and y = f(x), we can use directly ezplot(y) (easy plot) to plot y = f(x). Different attributes of ezplot can be used (using set command) to make graph more fancy or simple. plot(x,y) function can also be used to plot graph but before that we have to initialize x and y as arrays.
  3-D Plotting: 3-D plotting can be used in same way as above but functions change respectively as ezsurf(z) (where z is a function of x and y) and plot3(x,y,z) (where are x,y and z are initialized arrays)

- **Symbolic Math Toolbox** is used to plot, solve and manipulate symbolic math equations. A symbol is declared by using 'syms' keyword. In just above part, we can see how it can be used to plot 2-D and 3-D Functions. Solving can be done using 'solve' command ('vpasolve' solves numerically but 'solve' solves symbolically). Example: solve(x^2-1) will give x = -1,1 as solutions of x.

Question II.

A = [1 3 5 7 9 11 13 15;2.5 2 1.5 1 0.5 0 -0.5 -1;0.25 0.5 1 2 4 8 16 32]

Let A*B = C

Then C = [1 7 13;2.5 1 -0.5;0.25 2 16]

Hence, command to find B is:

B = A\C

Input and Output Images below:

Editor - D:\VIT\MATLAB\Refined\ans3.m

Command Window

```
>> A = [1 3 5 7 9 11 13 15;2.5 2 1.5 1 0.5 0 -0.5 -1;0.25 0.5 1 2 4 8 16 32]

A =

    1.0000    3.0000    5.0000    7.0000    9.0000   11.0000   13.0000   15.0000
    2.5000    2.0000    1.5000    1.0000    0.5000         0   -0.5000   -1.0000
    0.2500    0.5000    1.0000    2.0000    4.0000    8.0000   16.0000   32.0000

>> C = [1 7 13;2.5 1 -0.5;0.25 2 16]

C =

    1.0000    7.0000   13.0000
    2.5000    1.0000   -0.5000
    0.2500    2.0000   16.0000

>> B = A\C

B =

    1.0000    0.3445   -0.0766
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0    0.7943    0.7679
         0         0         0
         0   -0.1388    0.3086

fx >>
```

Question III.

MATLAB Code

Editor - D:\VIT\MATLAB\Refined\ans3.m

ans3.m | ans4.m

```
1    %Program for Question 3 - 16BCE0783
2    n = input('Enter the value of n = ');
3    list = [];  %Creating empty array
4    c = 0; %Initializing Counter
5    while n~=1
6        if rem(n,2)==0
7            c = c + 1;
8            f = n/2;
9            n = f;
10           list(c) = f;
11       else
12           c = c + 1;
13           f = (3*n) + 1;
14           n = f;
15           list(c) = f;
16       end
17   end
18   list
19   disp(['Number of steps is: ',num2str(c)])
```

Input and Output:

Editor - D:\VIT\MATLAB\Refined\ans3.m

Command Window

```
Enter the value of n = 783

list =

  Columns 1 through 9

       2350        1175        3526        1763        5290        2645        7936        3968        1984

  Columns 10 through 18

        992         496         248         124          62          31          94          47         142

  Columns 19 through 27

         71         214         107         322         161         484         242         121         364

  Columns 28 through 36

        182          91         274         137         412         206         103         310         155

  Columns 37 through 45

        466         233         700         350         175         526         263         790         395

  Columns 46 through 54

       1186         593        1780         890         445        1336         668         334         167

  Columns 55 through 63

        502         251         754         377        1132         566         283         850         425
```

Editor - D:\VIT\MATLAB\Refined\ans3.m

Command Window

```
         502         251         754         377        1132         566         283         850         425

  Columns 64 through 72

        1276         638         319         958         479        1438         719        2158        1079

  Columns 73 through 81

        3238        1619        4858        2429        7288        3644        1822         911        2734

  Columns 82 through 90

        1367        4102        2051        6154        3077        9232        4616        2308        1154

  Columns 91 through 99

         577        1732         866         433        1300         650         325         976         488

  Columns 100 through 108

         244         122          61         184          92          46          23          70          35

  Columns 109 through 117

         106          53         160          80          40          20          10           5          16

  Columns 118 through 121

           8           4           2           1

Number of steps is: 121
>>
```

Question IV.

Editor - D:\VIT\MATLAB\Refined\ans4.m

ans4.m

```
1    %Program for Question 4 Fibonacci Numbers - 16BCE0783
2    a = 0;b = 1;
3    n = input('Input the number of terms: ');
4    sum = 0; %Initializing Total Sum
5    for i = 1:n
6        sum = sum+a;
7        t = a;    %Lines 7 to 9 are for exchanging
8        a = b;
9        b = t+b;
10   end
11   disp(['The sum of first ',num2str(n),' Fibonacci Numbers is ',num2str(sum)])
```

Command Window

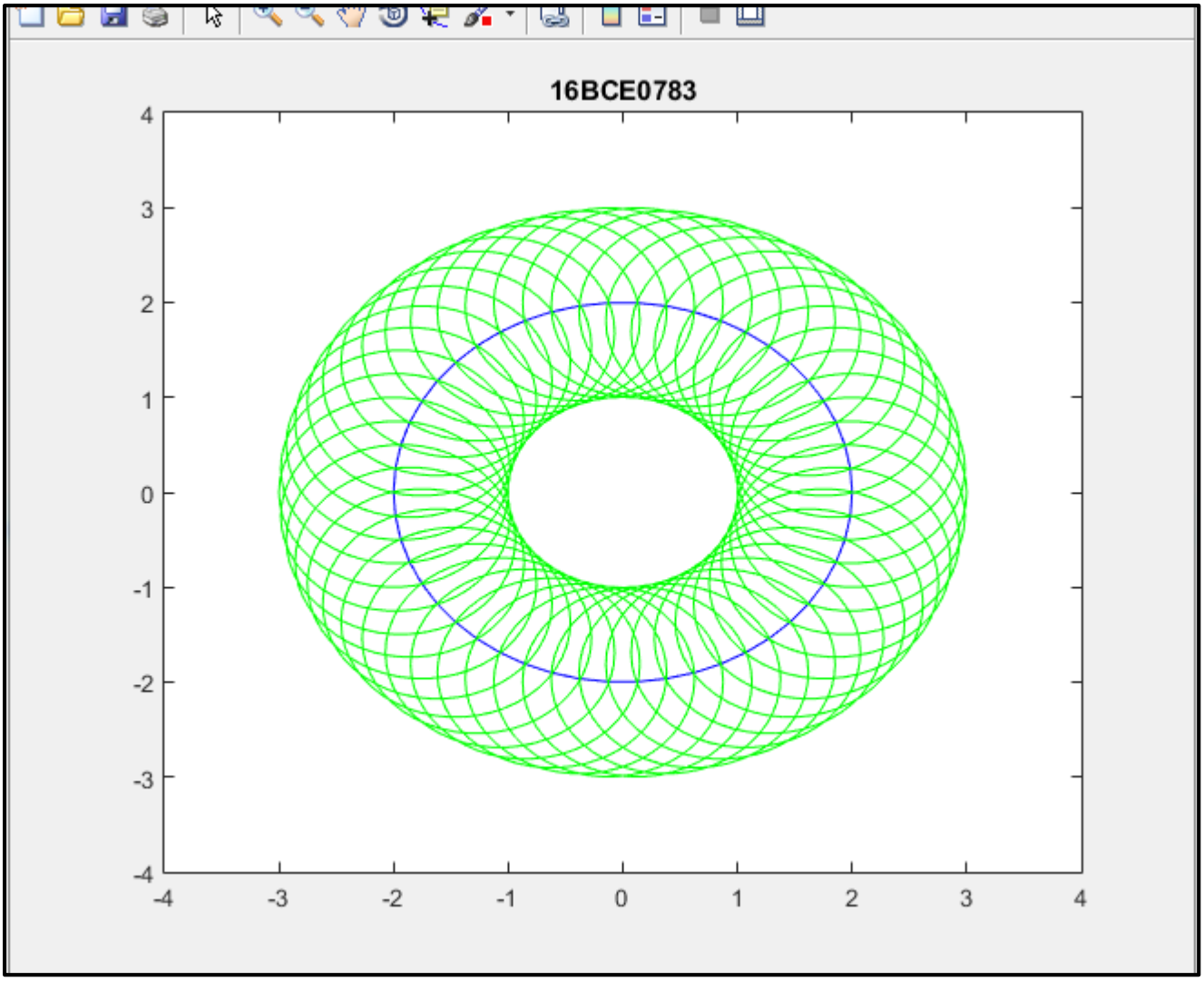
```
Input the number of terms: 783
The sum of first 783 Fibonacci Numbers is 3.139243139476952e+163
Input the number of terms: 18
The sum of first 18 Fibonacci Numbers is 4180
>>
```

Question V.

Editor - D:\VIT\MATLAB\Refined\ans5.m

ans5.m

```
1    %Program for Question 5 - 16BCE0783
2    th = 0:pi/25:2*pi;
3    x = 2 * cos(th);
4    y = 2 * sin(th);
5    h = plot(x,y,'color','blue');
6    axis([-4 4 -4 4])
7    title('16BCE0783')
8    hold on
9    for w=1:numel(th)
10   i = (1*cos(th))+x(w);
11   j = (1*sin(th))+y(w);
12   k = plot(i,j,'color','green');
13   end
```



Question VI.

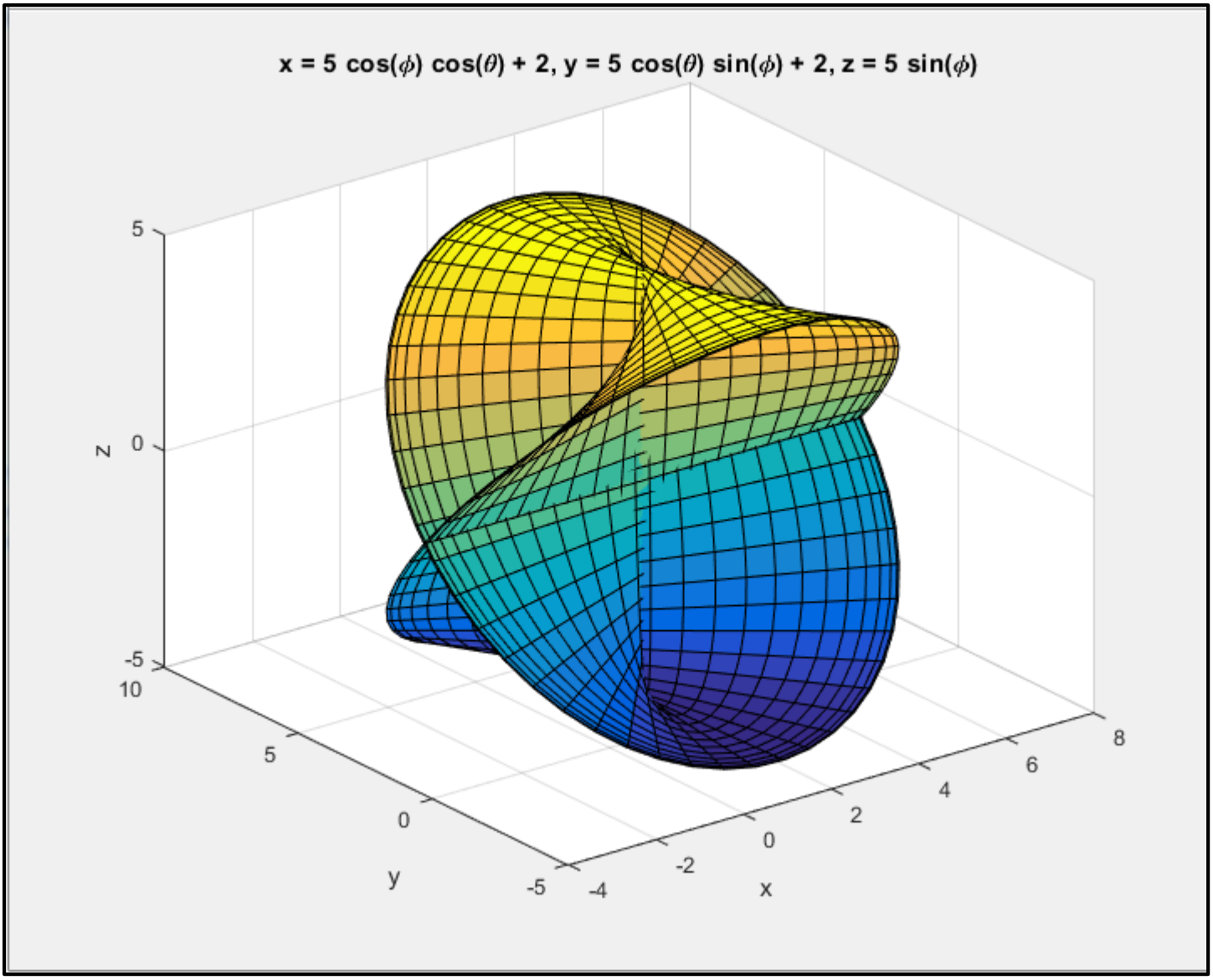
Editor - D:\VIT\MATLAB\Refined\ans6.m

ans6.m

```
%Program for Question 6 - 16BCE0783
syms theta phi
c = input('Enter the value of c: ');
a = input('Enter the value of a: ');
x = c+(a*cos(theta)*cos(phi));
y = c+(a*cos(theta)*sin(phi));
z = a*sin(phi);
ezsurf(x,y,z,[0 2*pi])
```

Command Window

```
Enter the value of c: 0
Enter the value of a: 4
>>
```


x = 5 cos(ϕ) cos(θ) + 2, y = 5 cos(θ) sin(ϕ) + 2, z = 5 sin(ϕ)
5
0
-5
z
10
5
0
-5
y
-4
-2
0
2
4
6
8
x