Credit Hours System **Cairo University**
Faculty of Engineering

Numerical Analysis Project

Lamees Mahmoud MohyEldeen	**1152023**
Aya Hossam	**1132125**
Eman Mohamed	**1170297**
Eman Gamal	**1170298**
Kamel Mohsen Kamel	**1162325**
Abdelrhman Ashraf	**1170060**
Hossam Hazem	**1170289**
Marwan Ehab	**1170057**
Abdelrahman mamdoh	**1170480**
Rabea Said Hamdy	**1155078**

Q1) ODE

```
clc;
clear;
Y= zeros(0,150);
Y1= zeros(0,150); % Y1=Y'
Y2= zeros(0,150); % Y2=Y1'=Y''
Y3= zeros(0,150); % Y3=Y2'=Y'''
Y4= zeros(0,150); % Y4=Y3'=Y''''
X= zeros(0,150);
% Set intial values
Y(1)=1;
Y1(1)=1;
Y2(1)=-1;
Y3(1)=1;
Y4(1)=-1;
X(1)=0;
h= 0.01; % step given
k = 2;% to Acess the value of the next state


for i=0:0.01:1.5
%set the peridictive values
X(k) = i+0.01;
Y(k) =Y(k-1)+h*Y1(k-1);
Y1(k) =Y1(k-1)+h*Y2(k-1);
Y2(k) =Y2(k-1)+h*Y3(k-1);
Y3(k) =Y3(k-1)+h*Y4(k-1);
Y4(k) =cos(2*X(k))-4*Y2(k); % given
% begin to calculate the corrective value with 10 iterations
for j=0:1:10
Y3(k)=Y3(k-1)+h*((Y4(k-1)+Y4(k))/2);
Y2(k)=Y2(k-1)+h*((Y3(k-1)+Y3(k))/2);
Y1(k)=Y1(k-1)+h*((Y2(k-1)+Y2(k))/2);
Y(k)=Y(k-1)+h*((Y1(k-1)+Y1(k))/2);
end
k=k+1;
end
plot(X,Y)
xlabel('X')                          %Labeling Horizontal Axis
ylabel('Y')                          %Labeling Vertical Axis
grid on
set(gca,'Fontsize',15)               %Changing the font size of the labels
```

Q2) PDE

```
%Numerical Project spring 2019
clear; clc                     %Clears workspace after every run
Nx=5;                          %Number of terms in X direction AKA n
Nt=20;                         %Number of terms in T direction AKA m
Lx=1;                          %The max X value
Lt=1;                          %The max T value
Nx=Nx+1;
Nt=Nt+1;
U=zeros(Nt,Nx);                %Intializing an empty array for Saving
the solution
x=linspace(0,Lx,Nx);           %Creating linear space for the x to draw
t=linspace(0,Lt,Nt);           %Creating linear space for the t to draw

%Using the given Boundry Conditions to fill the matrix
    U(:,1) = 0 ;                                %Left Boundary
Condition
    U(:,Nx) = 0;                                %Right Boundary
Condition
    U(1,2:(Nx-1))=sin(pi*0.2*((2:(Nx-1))-1));   %Bottom Boundary
Condition
    U(2,2:(Nx-1))=U(1,2:(Nx-1));                %From initial
condition ut=0(at t=0) 0<=x<=1
%Using finite difference formula we get this
%4*U((i+1),j)+4*U((i-1),j))-8*U(i,j)=-2*U(i,j)+U(i,(j+1))+U(i,(j-1))
% h=0.2 , k=0.05
for i=2:(Nt-1)
    for j=2:(Nx-1)
     U((i+1),j)=(6*U(i,j)+U(i,(j+1))+U(i,(j-1))-4*U((i-1),j))/4;
%Reorderd the equation above derived from the Finite Differnece formula
    end
end
%Plotting Calculated solution
figure('Name','Calculated');        %Naming Figure1 to draw the
calculated solution
surf(x,t,U);                        %Drawing the graph using x,t,and U
matrix
xlabel('X')                         %Labeling Horizontal Axis
ylabel('T')                         %Labeling Vertical Axis
zlabel('U')                         %Labeling Perp. Axis
set(gca,'Fontsize',10)              %Changing the font size of the
labels

%Exact solution
Exact=zeros(21,6);
for i=1:Nt
    for j=1:Nx
        Exact(i,j)=sin(pi*(Lx/(Nx-1))*(j-1))*cos(2*pi*(Lt/(Nt-1))*(i-
1)); %The Exact equation given to us
    end
end
%Plotting Exact solution
figure('Name','Exact');           %Naming Figure2 to draw the Exact
solution
surf(x,t,Exact);                  %Drawing the graph using x,t,and Exact
matrix
xlabel('X')                       %Labeling Horizontal Axis
ylabel('T')                       %Labeling Vertical Axis
zlabel('Exact')                   %Labeling Perp. Axis
```

```
set(gca,'Fontsize',10)           %Changing the font size of the labels
%Calculating ERROR

Error=abs((Exact-U)./Exact)*100;
Error(:,1)=1:21;                        %Giving default value 0 to first
column since 0/0 = Nan
Error(6,2:5)=Error(6,2:5)/(1.0e+17);
Error(16,2:5)=Error(16,2:5)/(1.0e+17);
disp(Error);
disp('Rows 6 & 16 are divided by (1.0e+17)');
disp('The shown data starts from column 2 since column 1 is all
zeroes');
```