Final Exam
Vibhanshu Jain,
CS19B1027

Q1: Solve the given ODE using MATLAB.

$y' = 2x + y,\ y(0) = -1$

a) By Euler method.

Academic-Files ▸ Semester 7 ▸ ICME ▸ Exams ▸ code ▸

Editor - D:\Academic-Files\Semester 7\ICME\Exams\code\1\Q1eulerMethod.m

Q1eulerMethod.m

```
%% Define the y vector
y = zeros(n+1,1);

%% Define the initial condition
y(1,1) = y0;

%% Define the Euler method
for i = 1:n
    y(i+1,1) = y(i,1) + h*myfunction(x(i), y(i,1));
end

%% Print the result
fprintf('The result of the Euler method at xf: %f and with number of step %f is: %f', xf, n, y(end,1));

%% Printing a line break
fprintf('\n');
```

Command Window

```
>> Q1eulerMethod
The result of the Euler method at xf: 0.200000 and with number of step 20.000000 is: -1.179810
fx >>
```

b) By Runge-Kutta 4th Order method.

Academic-Files ▸ Semester 7 ▸ ICME ▸ Exams ▸ code ▸

Editor - D:\Academic-Files\Semester 7\ICME\Exams\code\1\Q1rungeKuttaMethod.m

Q1rungeKuttaMethod.m

```
%% Define the initial condition
y(1,1) = y0;

%% Runge-Kutta method of order 4
for i = 1:n
    k1 = myfunction(x(i), y(i));
    k2 = myfunction(x(i) + h/2, y(i) + h*k1/2);
    k3 = myfunction(x(i) + h/2, y(i) + h*k2/2);
    k4 = myfunction(x(i) + h, y(i) + h*k3);
    y(i+1,1) = y(i,1) + h*(k1 + 2*k2 + 2*k3 + k4)/6;
end

%% Print the result
fprintf('The result of the Runge-Kutta method at xf %f and with number of step %f is: %f', xf, n, y(end,1));

%% Printing a line break
fprintf('\n');
```

Command Window

```
>> Q1eulerMethod
The result of the Euler method at xf: 0.200000 and with number of step 20.000000 is: -1.179810
>> Q1rungeKuttaMethod
The result of the Runge-Kutta method at xf 0.200000 and with number of step 20.000000 is: -1.178597
fx >>
```

Q2: Solve the given ODE using MATLAB.

$y' = x^2 + 4y$, $y(0) = 1$

a) By Trapezoidal method.

b) By Runge-Kutta 4th Order method

Academic-Files ▸ Semester 7 ▸ ICME ▸ Exams ▸ code ▸

Editor - D:\Academic-Files\Semester 7\ICME\Exams\code\2\Q2rungeKuttaMethod.m

Q2rungeKuttaMethod.m

```
%% Final Exam Q2
%% Vibhanshu Jain - CS19B1027

%% Solve the given ODE using MATLAB
%% y'= x^2 + 4y'
%% y(0) = 1

%% b : By Runge-Kutta 4th Order method

%% Define the function
myFunction = @(x,y) x^2 + 4*y;

%% Define the initial conditions
x0 = 0;
y0 = 1;

%% the values where we have the find the function value
xf = 0.2;

%% Defining the number of steps
n = 20;

%% Define the step size
h = (xf - x0)/n;

```

Command Window

```
>> Q2rungeKuttaMethod
The result of the Runge-Kutta Method at xf 0.200000 and with number of step 20.000000 is: 2.228839
>> Q2rungeKuttaMethod
The result of the Runge-Kutta Method at xf 0.200000 and with number of step 20.000000 is: 2.228839
fx >>
```

Academic-Files ▸ Semester 7 ▸ ICME ▸ Exams ▸ code ▸

Editor - D:\Academic-Files\Semester 7\ICME\Exams\code\2\Q2trapezoidalMethod.m

Q2trapezoidalMethod.m

```matlab
%% Final Exam Q2
%% Vibhanshu Jain - CS19B1027


%% Solve the given ODE using MATLAB
%% y'= x^2 + 4y'
%% y(0) = 1

%% a: By Trapezoidal method

%% Define the function
myFunction = @(x,y) x^2 + 4*y;

%% Define the initial conditions
x0 = 0;
y0 = 1;

%% the values where we have the find the function value
xf = 0.2;

%% Defining the number of steps
n = 20;

%% Define the step size
h = (xf - x0)/n;
```

Command Window

```
>> Q2rungeKuttaMethod
The result of the Runge-Kutta Method at xf 0.200000 and with number of step 20.000000 is: 2.228839
>> Q2trapezoidalMethod
The result of the Trapezoidal Method at xf 0.200000 and with number of step 20.000000 is: 2.228379
fx >>
```

Q3:

Academic-Files ▸ Semester 7 ▸ ICME ▸ Exams ▸ code ▸ 3

Live Editor - D:\Academic-Files\Semester 7\ICME\Exams\code\3\StrainedBracketExample.mlx *

StrainedBracketExample.mlx * ✕ +

Deflection Analysis of Bracket

This example shows how to analyze a 3-D mechanical part under an applied load using finite element analysis (FEA) and determine the maximal deflection.

Create Structural Analysis Model

The first step in solving a linear elasticity problem is to create a structural analysis model. This is a container that holds the geometry, structural material properties, damping parame boundary loads, boundary constraints, superelement interfaces, initial displacement and velocity, and mesh.

```
model = createpde('structural','static-solid');
```

Import Geometry

Import an STL file of a simple bracket model using the `importGeometry` function. This function reconstructs the faces, edges and vertices of the model. It can merge some faces an numbers can differ from those of the parent CAD model.

```
importGeometry(model,'BracketWithHole.stl');
```

Plot the geometry, displaying face labels.

```
figure
pdegplot(model,'FaceLabels','on')
view(30,30);
title('Bracket with Face Labels')
```

Bracket with Face Labels

Academic-Files ▸ Semester 7 ▸ ICME ▸ Exams ▸ code ▸ 3

Live Editor - D:\Academic-Files\Semester 7\ICME\Exams\code\3\StrainedBracketExample.mlx

StrainedBracketExample.mlx ✕ +

Calculate Solution

Use the `solve` function to calculate the solution.

```
result = solve(model)
```

```
result = 
  StaticStructuralResults with properties:

      Displacement: [1×1 FEStruct]
            Strain: [1×1 FEStruct]
            Stress: [1×1 FEStruct]
    VonMisesStress: [5993×1 double]
              Mesh: [1×1 FEMesh]
```

Examine Solution

Find the maximal deflection of the bracket in the z-direction.

```
minUz = min(result.Displacement.uz);
fprintf('Maximal deflection in the z-direction is %g meters.', minUz)
```

```
Maximal deflection in the z-direction is -0.000132923 meters.
```

Plot Displacement Components

Plot the components of the solution vector. The maximal deflections are in the z-direction. Because the part and the loading are symmetric, the x-displacement and z-dis the y-displacement is antisymmetric with respect to the center line

Academic-Files ▸ Semester 7 ▸ ICME ▸ Exams ▸ code ▸ 3

Live Editor - D:\Academic-Files\Semester 7\ICME\Exams\code\3\StrainedBracketExample.mlx

StrainedBracketExample.mlx

Plot values of the von Mises stress at nodal locations. Use the same `jet` colormap.

```
figure
pdeplot3D(model,'ColorMapData',result.VonMisesStress)
title('von Mises stress')
colormap('jet')
```