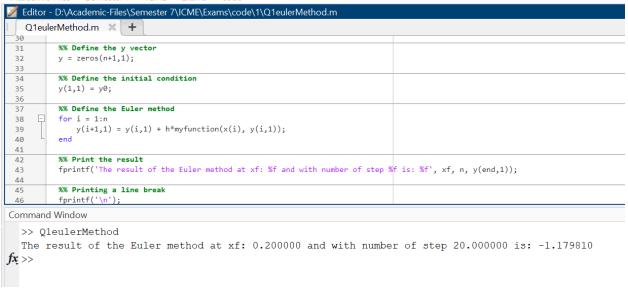
## Final Exam Vibhanshu Jain, CS19B1027

**Q1:** Solve the given ODE using MATLAB.

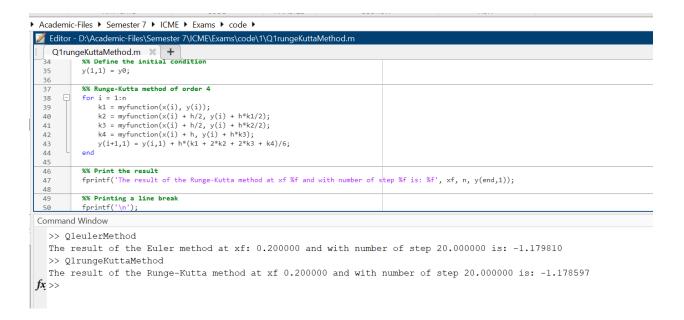
y' = 2x + y, y(0) = -1

a) By Euler method.

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



b) By Runge-Kutta 4th Order method.



**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

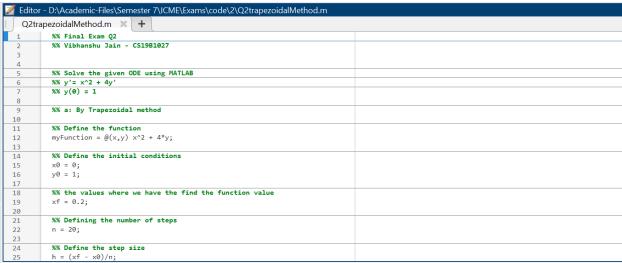
Editor - D:\Academic-Files\Semester 7\(CME\Exams\code\2\Q2\rungeKuttaMethod.m\)	Academic-Files ▶ Semester 7 ▶ ICME ▶ Exams ▶ code ▶	
1	Zelitor - D:\Academic-Files\Semester 7\ICME\Exams\code\2\Q2rungeKuttaMethod.m	
1	O2rungeKuttaMethod.m × +	
2	1	
3	2	
## Solve the given ODE using MATLAB    S		No violatista sait - Colobio.
%% y(0) = 1  %% b: By Runge-Kutta 4th Order method  %% Define the function myFunction = @(x,y) x^2 + 4*y;  %% Define the initial conditions  4		%% Solve the given ODE using MATLAB
7 8	5	%% y'= x^2 + 4y'
<pre>8</pre>	6	%% y(0) = 1
9 10	7	
10	8	%% b : By Runge-Kutta 4th Order method
myFunction = @(x,y) x^2 + 4*y;    13	9	
12 13	10	%% Define the function
13	11	$myFunction = @(x,y) x^2 + 4*y;$
14	12	
15	13	
16 17		· ·
17		$y\theta = 1;$
18		
19 20		
20		$x \neq 0.2;$
21		NV D. S. L.
22 23		
23		n = 20;
24 h = (xf - x0)/n; 25		9W Define the star size
25		
		11 - (x1 - x0)/11,
		nd 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 ▶



## 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  $\mathbf{fx} >>$ 

## 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.
   1
             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 and
           numbers can differ from those of the parent CAD model.
   2
            importGeometry(model, 'BracketWithHole.stl');
            Plot the geometry, displaying face labels.
             pdegplot(model, 'FaceLabels', 'on')
   4
             view(30,30);
title('Bracket with Face Labels')
   6
                                           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.
  19
             result = solve(model)
                 {\bf Static Structural Results} \ \ {\bf with} \ \ {\bf properties:}
                     Displacement: [1×1 FEStruct]
                   Strain: [1x1 FEStruct]
Stress: [1x1 FEStruct]
VonMisesStress: [5993x1 double]
Mesh: [1x1 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)
  21
               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 \nu-displacement is antisymmetric with respect to the center line
```

