TASK 5

Q1) Utilize MATLAB's linear equation solvers (e.g., direct solvers, iterative methods) to solve the FEA system.

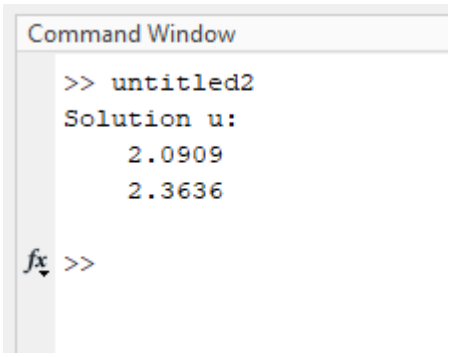
MATLAB'S Code using Direct Solver:

```
K = [4, -1; -1, 3];
f = [6; 5];

% Here I am using the direct solver method to solve fea
u = K \ f;

disp("Solution u:");
disp(u);
```

MATLAB COMMAND WINDOW OUTPUT:



MATLAB'S Code using Iterative Solver:

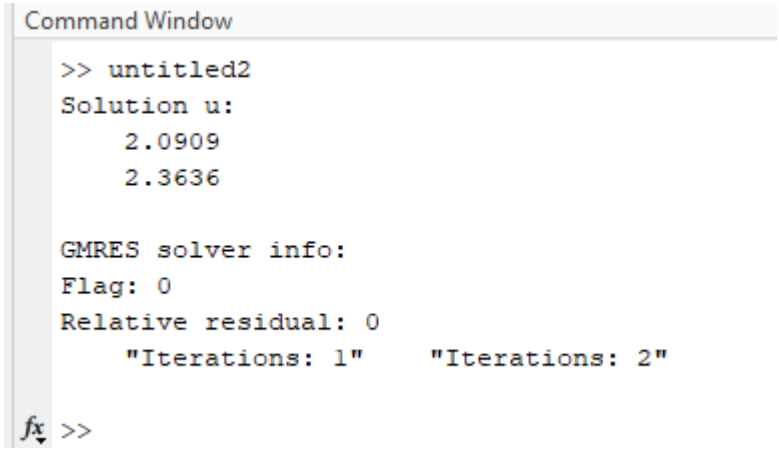
```
K = [4, -1; -1, 3];
f = [6; 5];

% Set up the GMRES solver parameters
tolerance = 1e-6;
maxIterations = 2;

% Solve the FEA system using GMRES iterative solver
[u, flag, relres, iter] = gmres(K, f, [], tolerance, maxIterations);

disp("Solution u:");
disp(u);
disp("GMRES solver info:");
disp("Flag: " + flag);
disp("Relative residual: " + relres);
disp("Iterations: " + iter);
```

MATLAB COMMAND WINDOW OUTPUT:



Q2) Check the accuracy of the results against known analytical solutions or commercial FEA software.

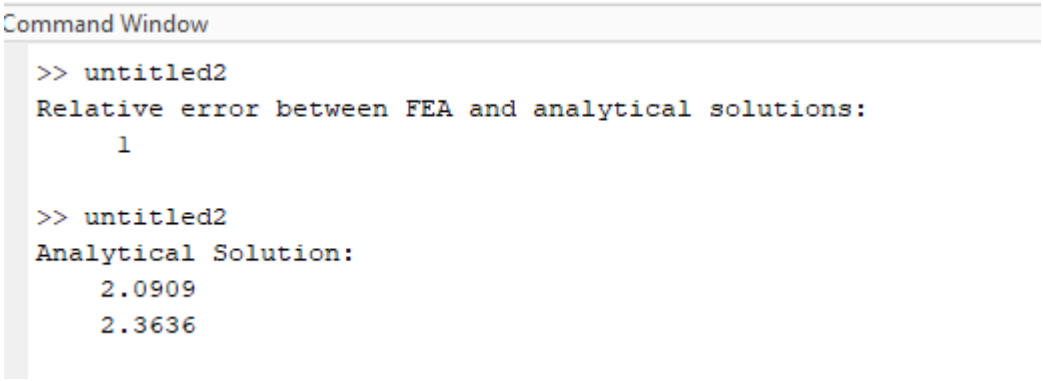
MATLAB'S Code:

```
K = [4, -1; -1, 3];
f = [6; 5];

% Analytical solution
u_analytical = inv(K) * f;

disp("Analytical Solution:");
disp(u_analytical);
```

The results for the analytically solved question is:



Relative Error between u_direct and u_analytical:

relative_error_direct = norm(u_direct - u_analytical) / norm(u_analytical)

For the first value of u_ direct, u_analytical that is **2.0909.**

= 2.0909-2.0909/ 2.0909

= 0% error

For 2nd value of u_direct and u_analytical is **2.3636.**

= 2.3636-2.3636/2.3636

= 0% error

Relative Error between u_iterative and u_analytical:

relative_error_iterative = norm(u_iterative - u_analytical) / norm(u_analytical)

For the first value of u_ iterative, u_analytical that is **2.0909.**

= 2.0909-2.0909/ 2.0909

= 0% error

For 2nd value of u_iterative and u_analytical is **2.3636.**

= 2.3636-2.3636/2.3636

= 0% error

Conclusion:

This means that FEA results (both direct and iterative) match the analytical solution perfectly in this case.