Task 4

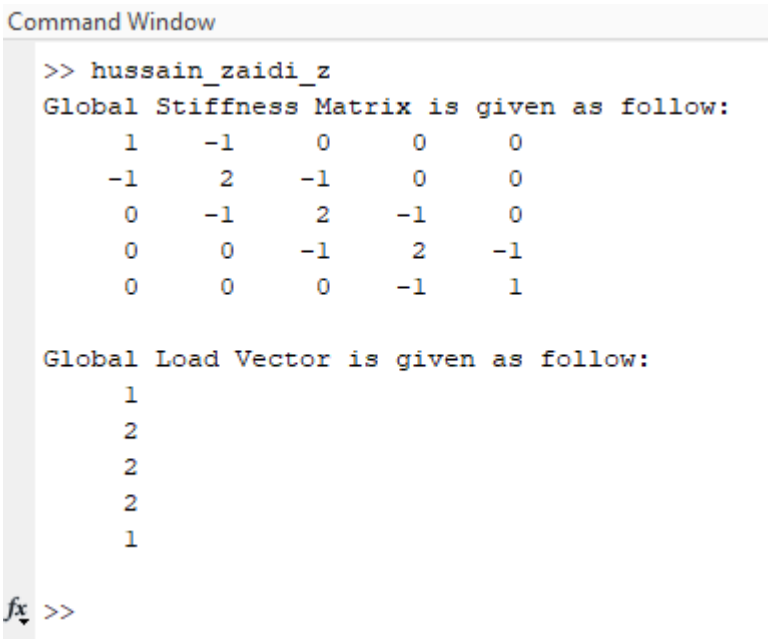
Shape Functions and Interpolation

Q) Create MATLAB functions to assemble the global stiffness matrix and load vectors from element matrices and vectors.

- **MATLAB Code:**

```
numNodes = 5;
numElements = 4;
elementNodes = [1, 2; 2, 3; 3, 4; 4, 5];


Q = zeros(numNodes, numNodes);
Z = zeros(numNodes, 1);

Qasim = zeros(2, 2, numElements);
for e = 1:numElements
   Qasim(:,:,e) = [1, -1; -1, 1];
end

zaidi = ones(2, numElements);


for e = 1:numElements
    nodes = elementNodes(e, :);

    for i = 1:length(nodes)
        for j = 1:length(nodes)
            Q(nodes(i), nodes(j)) = Q(nodes(i), nodes(j)) + Qasim(i, j, e);
        end
    end

    for i = 1:length(nodes)
        Z(nodes(i)) = Z(nodes(i)) + zaidi(i, e);
    end
end


disp("Global Stiffness Matrix is given as follow:");
disp(K);
disp("Global Load Vector is given as follow:");
disp(F);
```

- **Command Window Output:**



Q) Apply appropriate boundary conditions (e.g., essential and natural) to the global system.

- **MATLAB Code:**

```
numofNodes = 5;
numofElements = 4;
elementNodes = [1, 2; 2, 3; 3, 4; 4, 5];

Q = zeros(numofNodes, numofNodes);
Z = zeros(numofNodes, 1);

Qasim = zeros(2, 2, numofElements);
for e = 1:numofElements
    Qasim(:,:,e) = [1, -1; -1, 1];
end


zaidi = ones(2, numofElements);

for e = 1:numofElements
    nodes = elementNodes(e, :);

    for i = 1:length(nodes)
        for j = 1:length(nodes)
            Q(nodes(i), nodes(j)) = Q(nodes(i), nodes(j)) + Qasim(i, j, e);
```

```
        end
    end

    for i = 1:length(nodes)
        Z(nodes(i)) = Z(nodes(i)) + zaidi(i, e);
    end
end


fixedNodessuchas = [1];
forceNodes = [5];
forceMagnitude = 2.0;


for i = fixedNodessuchas
    Q(i, :) = 0;
    Q(i, i) = 1;
    Z(i) = 0;
end

for i = forceNodes
    Z(i) = Z(i) + forceMagnitude;
end

displacements = Q I am running a few minutes late; my previous meeting is running
over. Z;

disp("Displacements are shown as :");
disp(displacements);
```

- **Command Window Output:**