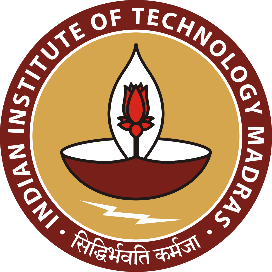
**ASSIGNMENT**

**ME 5204 – FINITE ELEMENT ANALYSIS**

****

**Department of Mechanical Engineering**

**Machine Design Section**

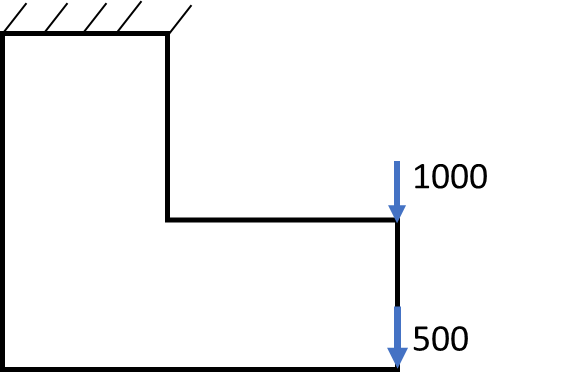
**Indian Institute of Technology, Madras - 600036**

**Submitted By,**

**N. SAI TEJA**

**ME22S084**

**Problem Statement 1:**

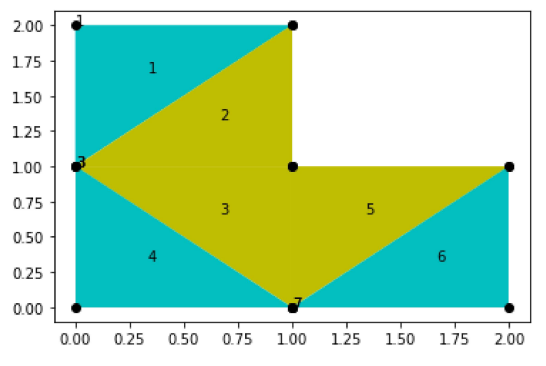
For a given 2-Dimensional structure as shown in figure 1, evaluate the displacements at each point and also evaluate the stresses, following a complete finite element procedure using python.

**Fig. 1: 2-D Structure**

**Solution Methodology:**

**1. Pre-Processing:**

i. Discretization: Dividing the given structure into finite element entities i.e., nodes and elements as shown in figure 2.



**Fig. 2: FEM Model**

The given structure is divided into 6 Constant Strain Triangular (CST) elements and 8 nodes.

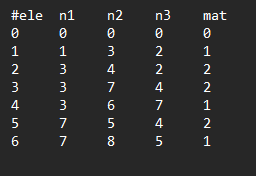
The structure is made up of two different kinds of elements, which is indicated in the figure with two different colors.

The following text documents are created for the extraction of the data of the structure.

1. COORD
2. NCA
3. MAT
4. LOAD\_BC
5. DISP\_BC

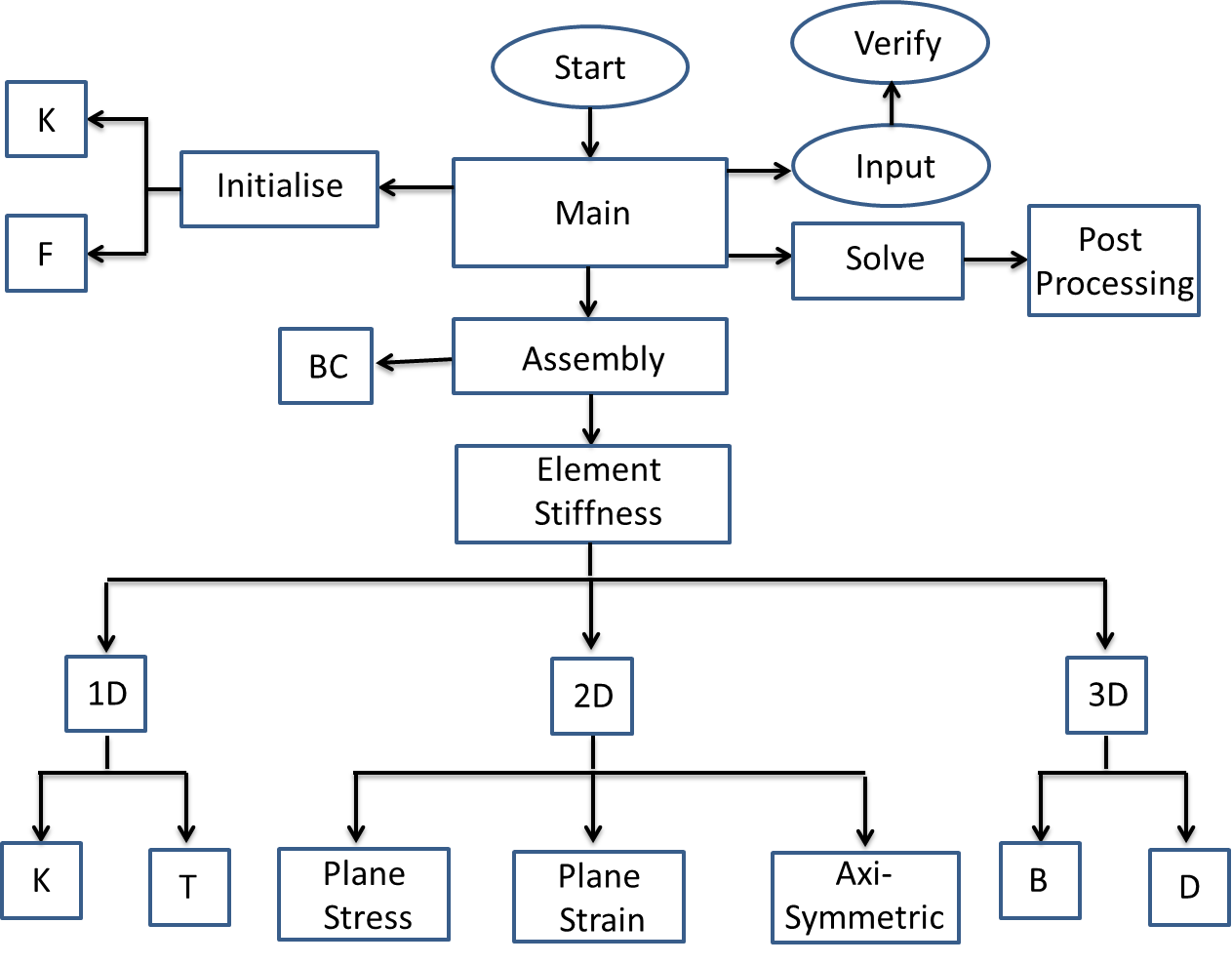
The first two documents, i.e., ‘COORD’ and ‘NCA’ contains the data related to the geometry of the structure, i.e., it contains information about the nodal coordinates and the element nodal connection array. ‘MAT’ contains information about the material properties i.e., it contains values of Young’s modulus and poison’s ratio. ‘LOAD\_BC’ & ‘DISP\_BC’ contains information about the external loads applied and supports given to the structure.

The figure below shows the data in the text document:



**Fig. 3: Nodal Connection Array (NCA)**

Following shows the flowchart for solving a Finite Element Problem:



**Fig. 4: Flow Chart**

The following shows the implemented python code for solving of the given 2D structure:

**Problem Statement 2:**

Using Galerkin approximation solve for approximate solution of ‘**sin(x)’**. Use complete finite element procedure and make use of monomial bases.

The element stiffness matrix can be calculated using:

The force matrix can be calculated using:

Where, f = sin(x)

The following is the implemented Python code for the approximate solution of **“sin(x)”:**