S.N.	PROGRAM CODE	COMMENTS

1
2
 	
3
4
5
6
 	
7
 	
9
10
 	
11
12
 	
13
14
 	
15
 	
16
17
18
19
20
21
22
23
24
25
26
 	
27
28
29
30
31
32
33
34
35
36
 	
37
38
 	
39
	clc
	clear
	
	K=[5.5 -4.6 0 0 0;-4.6 14 -4.6 0 0;...
	.4 -4.6 11 -4.6 .4; 0 0 -4.6 14.2 -
	4.6;...
	0 0 4.0 -4.6 6.5];
	
	P=[50 200 100 200 90];
	
	K
	P
	
	nn=5;
	ne=4;
	
	loc=[1 5]';
	bc=[15 4]';
	
	c=length(bc);
	
	for i=1:c
	aa=K(loc(i),loc(i));
	P(loc(i))=aa*bc(i);
	for j=1:nn
	if j==loc(i)
	K(loc(i),j)=K(loc(i),j);
	else
	K(loc(i),j)=0;
	end
	end
	end
	
	for i=1:c
	for j=1:nn
	if j~=loc(i)
	sum(j)=K(j,loc(i))*bc(i);
	P(j)=P(j)-sum(j);
	K(j,loc(i))=0
	else
	end
	end
	end
	
	KM
	PM=P'
	
	U=inv(KM)*PM
		Note:
		1. *Important variables that store information for pre-processing and those that store results for post-processing will be introduced in the beginning. Other variables, of lesser importance, used in the program will be explained as and when it is observed.*
		2. It is assumed that reader is conversed with the symbols used in the theoretical formulation produced here in the earlier section.
		
		Variable information:
		`K` : Global stiffness matrix (*Input 1*)
		`P` : Load vector (*Input 2*)
		`nn` : Number of nodes (*Input 3*)
		`ne` : Number of elements (*Input 4*)
		`loc` : Boundary location matrix (*Input 5*)
		`bc` : Boundary value matrix (*Input 6*)
		`KM` : Modified global stiffness matrix (*Output 1*)
		`PM` : Modified global force matrix (Output 2)
		
		(Line 1-2): It clears the *command window*[1] screen and clears the *workspace*[2] of all its historical contents.
		
		(Line 3-6): Assign the global stiffness matrix to variable **`K`** without applying boundary conditions. **[INPUT 1]**
		
		(Line 7): Assign global force matrix to variable **`P`** without applying boundary conditions. **[INPUT 2]**
		
		(Line 9-10): Displays global stiffness matrix **`K`** and global force matrix **`P`** without applying boundary conditions.
		
		(Line 11-12): Assign number of nodes and number of elements to variable **`nn`** and **`ne`** respectively. **[INPUT 3,4]**
		
		(Line 13-14): Assign node number of boundary conditions to be specified in column form in variable **`loc`** and assign the boundary values to variable **`bc`** corresponding to **`loc`** matrix. **[INPUT 5,6]**
		
		(Line 15-26): The for loop runs **`c`** times corresponding

		to the number of boundary conditions to be specified. This is step one of the manipulation. If U_i and U_j are known boundary conditions then set row i and j equal to zero except K_{ii} and K_{jj}. At the same, corresponding row i and j of global force matrix is replaced by K_{ii} * U_i and K_{jj}* U_j respectively. **(Line 27-36):** The for loop runs **c** times corresponding to the number of boundary conditions to be specified. Initially replace global force matrix such that $P(:,i) = P(:,i) - K(:,i) * U_i - K(:,j) * U_j$ except $P(i,1)$ and $P(j,1)$. U_i Corresponds to the displacement at node i. Then set column i and j equal to zero except K_{ii} and K_{jj}. **(Line 37-38):** Displays modified global stiffness matrix and global force matrix. **[OUTPUT 1,2]** **(Line 39):** Displays displacements.

Terms:

1. Command window:

2. Editor:

3. Workspace: