21

FEM Program for Space Trusses

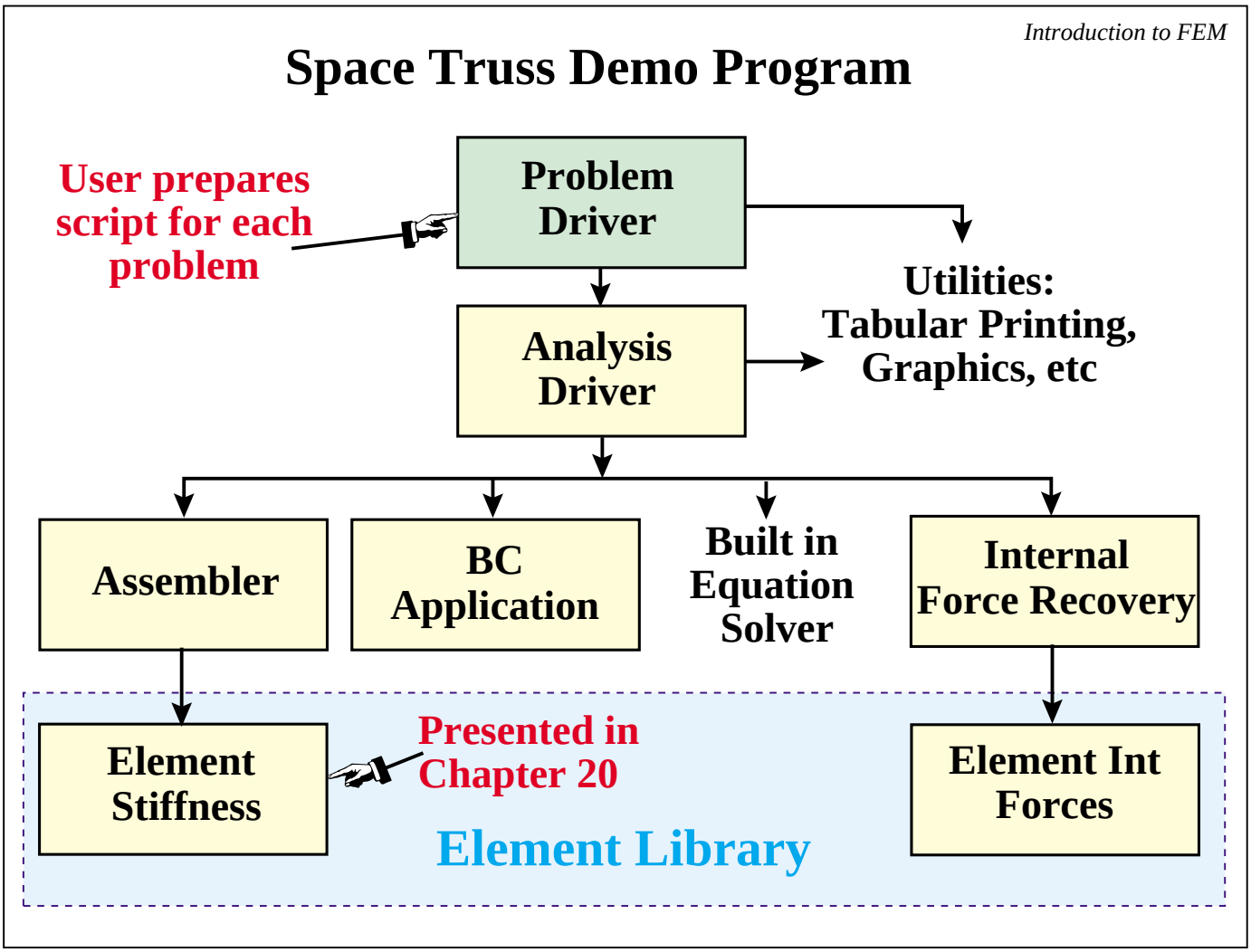
The Three Basic Stages of a FEM Program Based on the Direct Stiffness Method

Preprocessing **: defining the FEM model**

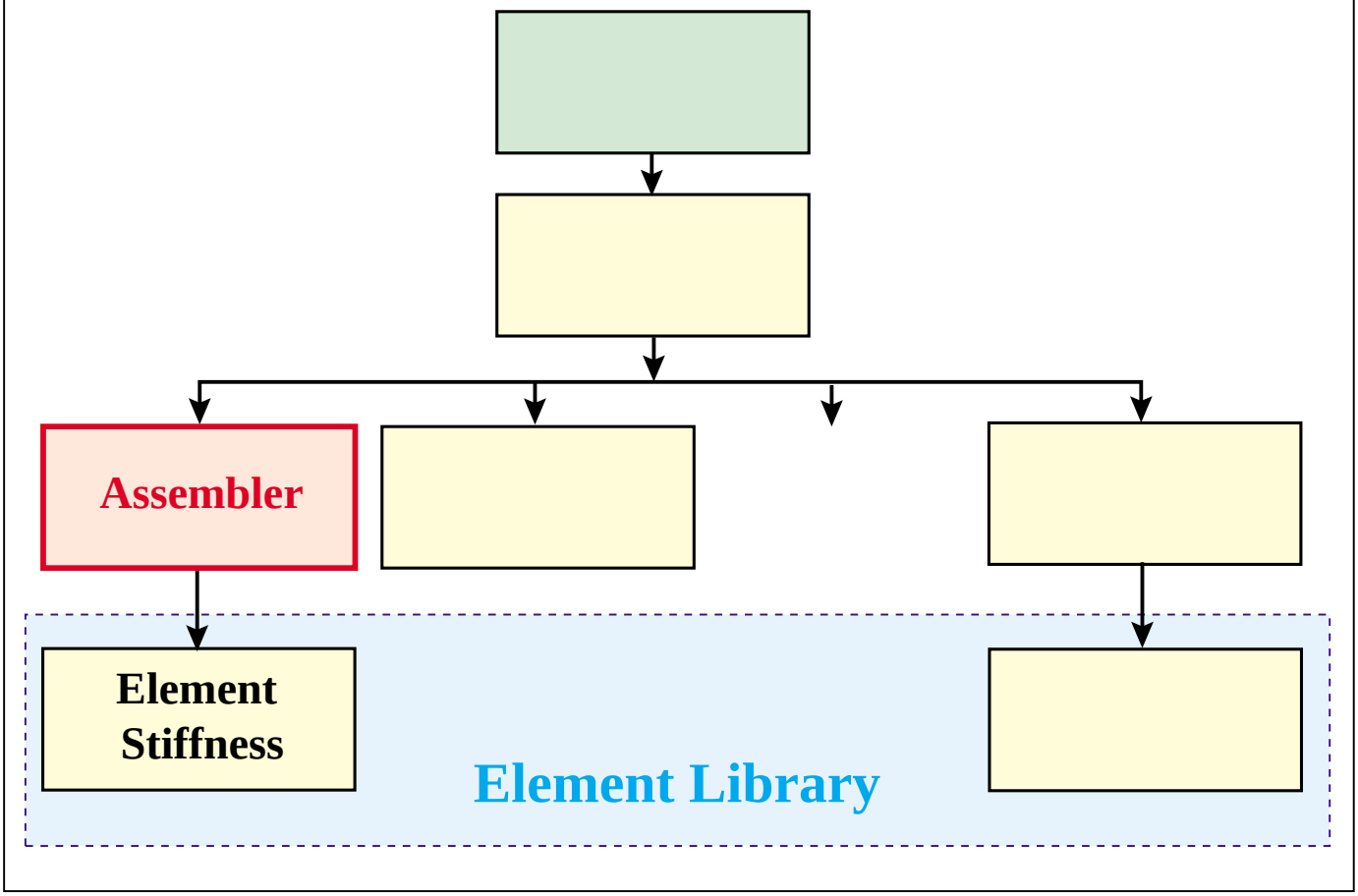
Processing **: setting up the stiffness equations and solving for displacements**

Postprocessing **: recovery of derived quantities and presentation of results**

Space Truss Demo Program



Next: Master Stiffness Assembler



Space Truss Assembler

```
SpaceTrussMasterStiffness[nodxyz_,elenod_,
   elemat_,elefab_,prcopt_]:=Module[
  {numele=Length[elenod],numnod=Length[nodxyz],neldof,
  e,eftab,ni,nj,i,j,ii,jj,ncoor,Em,A,options,Ke,K},
  K=Table[0,{3*numnod},{3*numnod}];
  For [e=1, e<=numele, e++, {ni,nj}=elenod[[e]];
      eftab={3*ni-2,3*ni-1,3*ni,3*nj-2,3*nj-1,3*nj};
      ncoor={nodxyz[[ni]],nodxyz[[nj]]};
      Em=elemat[[e]]; A=elefab[[e]]; options=prcopt;
      Ke=SpaceBar2Stiffness[ncoor,Em,A,options];
      neldof=Length[Ke];
      For [i=1, i<=neldof, i++, ii=eftab[[i]];
          For [j=i, j<=neldof, j++, jj=eftab[[j]];
              K[[jj,ii]]=K[[ii,jj]]+=Ke[[i,j]] ];
          ];
      ]; Return[K];
  ];
```

(Space truss element stiffness module omitted, presented in Ch 20)

Space Truss Assembler Test

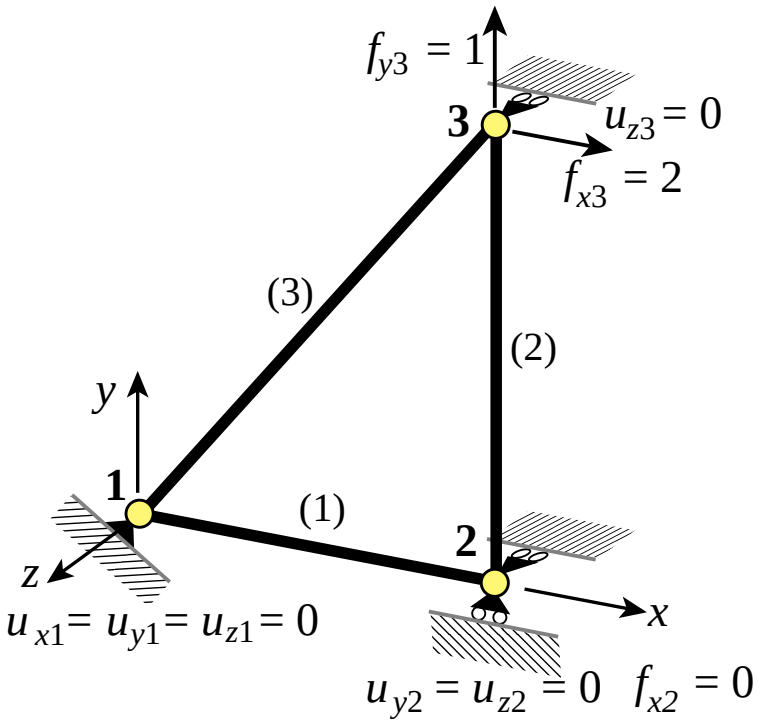
```
ClearAll[nodxyz,elemat,elefab,eleopt];
nodxyz={{0,0,0},{10,0,0},{10,10,0}};
elenod= {{1,2},{2,3},{1,3}};
elemat= Table[100,{3}]; elefab= {1,1/2,2*Sqrt[2]}; prcopt= {False};
K=SpaceTrussMasterStiffness[nodxyz,elenod,elemat,elefab,prcopt];
Print["Master Stiffness of Example Truss in 3D:\n",K//MatrixForm];
Print["eigs of K:",Chop[Eigenvalues[N[K]]]];
```

Master Stiffness of Example Truss in 3D:

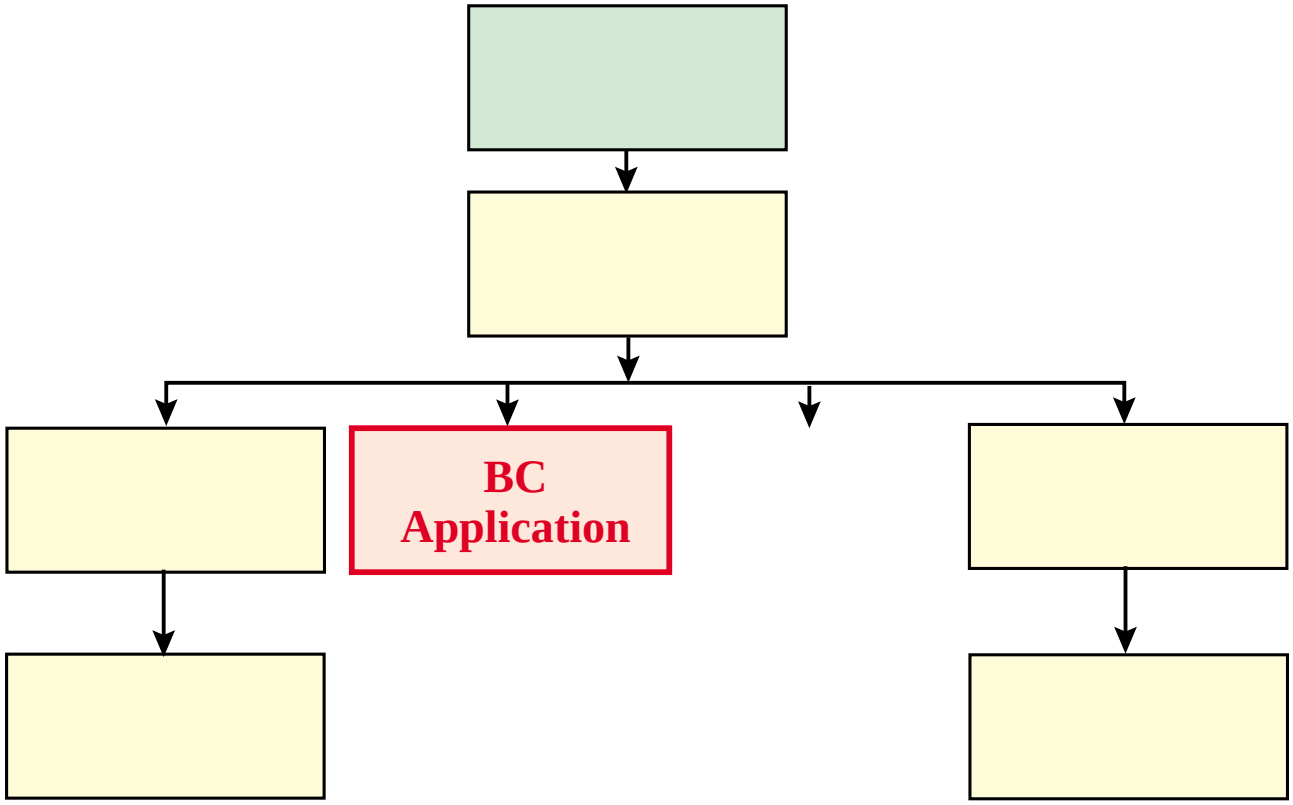
$$\begin{pmatrix} 20 & 10 & 0 & -10 & 0 & 0 & -10 & -10 & 0 \\ 10 & 10 & 0 & 0 & 0 & 0 & -10 & -10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -10 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 & 0 & 0 & -5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -10 & -10 & 0 & 0 & 0 & 0 & 10 & 10 & 0 \\ -10 & -10 & 0 & 0 & -5 & 0 & 10 & 15 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

eigs of K: {45.3577, 16.7403, 7.902, 0, 0, 0, 0, 0, 0}

Assembler Test Uses Example Plane Truss in 3D



Next: Application of Boundary Conditions



Application of Displacement BCs

```
ModifiedMasterStiffness[nodtag_,K_] := Module[
  {i,j,k,n=Length[K],pdof,np,Kmod=K},
  pdof=PrescDisplacementDOFTags[nodtag]; np=Length[pdof];
  For [k=1,k<=np,k++, i=pdof[[k]];
      For [j=1,j<=n,j++, Kmod[[i,j]]=Kmod[[j,i]]=0];
      Kmod[[i,i]]=1];
  Return[Kmod]];

ModifiedNodeForces[nodtag_,nodval_,K_,f_]:= Module[
  {i,j,k,n=Length[K],pdof,pval,np,d,c,fmod=f},
  pdof=PrescDisplacementDOFTags[nodtag]; np=Length[pdof];
  pval=PrescDisplacementDOFValues[nodtag,nodval]; c=Table[1,{n}];
  For [k=1,k<=np,k++, i=pdof[[k]]; c[[i]]=0];
  For [k=1,k<=np,k++, i=pdof[[k]]; d=pval[[k]];
      fmod[[i]]=d; If [d==0, Continue[]];
      For [j=1,j<=n,j++, fmod[[j]]-=K[[i,j]]*c[[j]]*d];
      ];
  Return[fmod]];
```

Restriction: single freedom constraints only. However, logic in **ModifiedNodeForces** accounts for nonzero prescribed displacements.

Test BC Applicator

```
ClearAll[K,f,v1,v2,v4]; Km=Array[K,{6,6}];
Print["Master Stiffness: ",Km//MatrixForm];
nodtag={{1,1},{0,1},{0,0}}; nodval={{v1,v2},{0,v4},{0,0}};
Kmod=ModifiedMasterStiffness[nodtag,Km];
Print["Modified Master Stiffness:",Kmod//MatrixForm];
fm=Array[f,{6}]; Print["Master Force Vector:",fm];
fmod=ModifiedNodeForces[nodtag,nodval,Km,fm];
Print["Modified Force Vector:",fmod//MatrixForm];
```

Master Stiffness:

$$\begin{pmatrix} K[1,1] & K[1,2] & K[1,3] & K[1,4] & K[1,5] & K[1,6] \\ K[2,1] & K[2,2] & K[2,3] & K[2,4] & K[2,5] & K[2,6] \\ K[3,1] & K[3,2] & K[3,3] & K[3,4] & K[3,5] & K[3,6] \\ K[4,1] & K[4,2] & K[4,3] & K[4,4] & K[4,5] & K[4,6] \\ K[5,1] & K[5,2] & K[5,3] & K[5,4] & K[5,5] & K[5,6] \\ K[6,1] & K[6,2] & K[6,3] & K[6,4] & K[6,5] & K[6,6] \end{pmatrix}$$

Modified Master Stiffness:

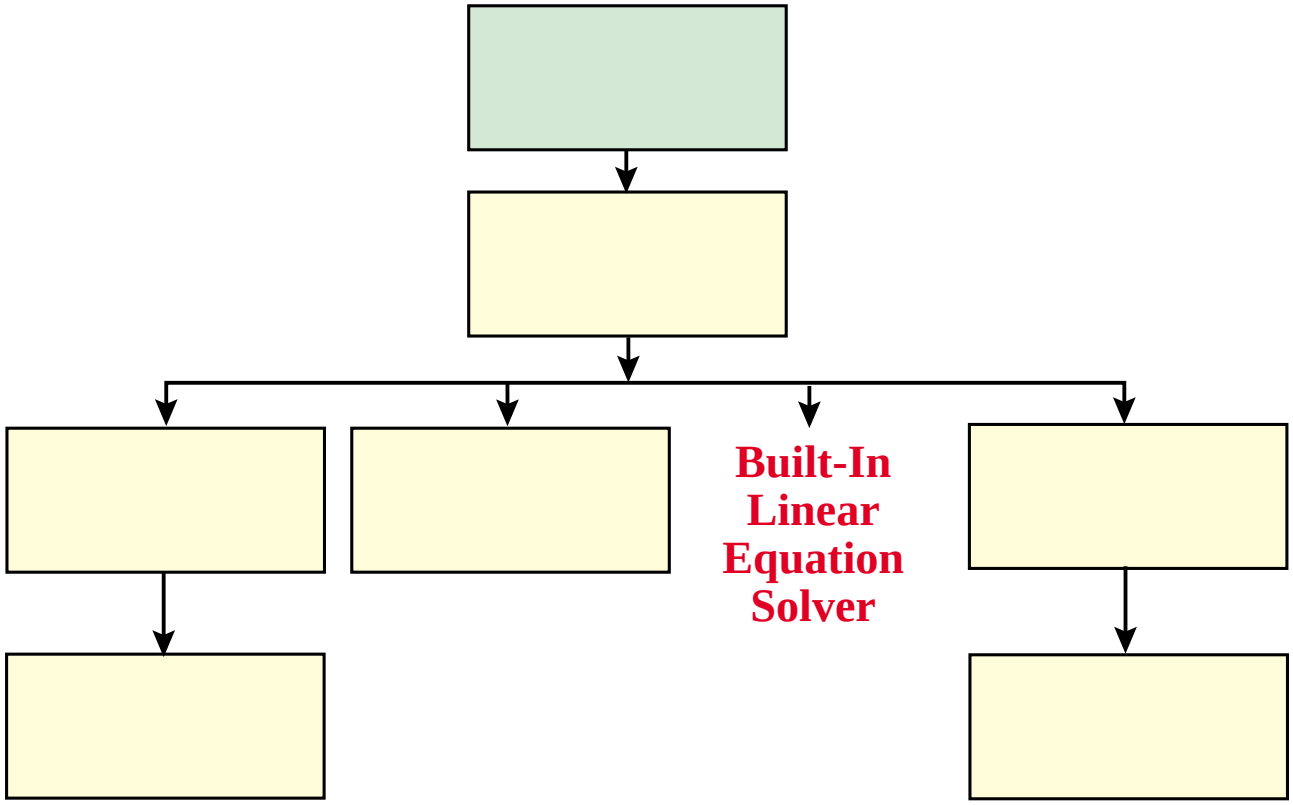
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & K[3,3] & 0 & K[3,5] & K[3,6] \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & K[5,3] & 0 & K[5,5] & K[5,6] \\ 0 & 0 & K[6,3] & 0 & K[6,5] & K[6,6] \end{pmatrix}$$

Master Force Vector: { f[1], f[2], f[3], f[4], f[5], f[6] }

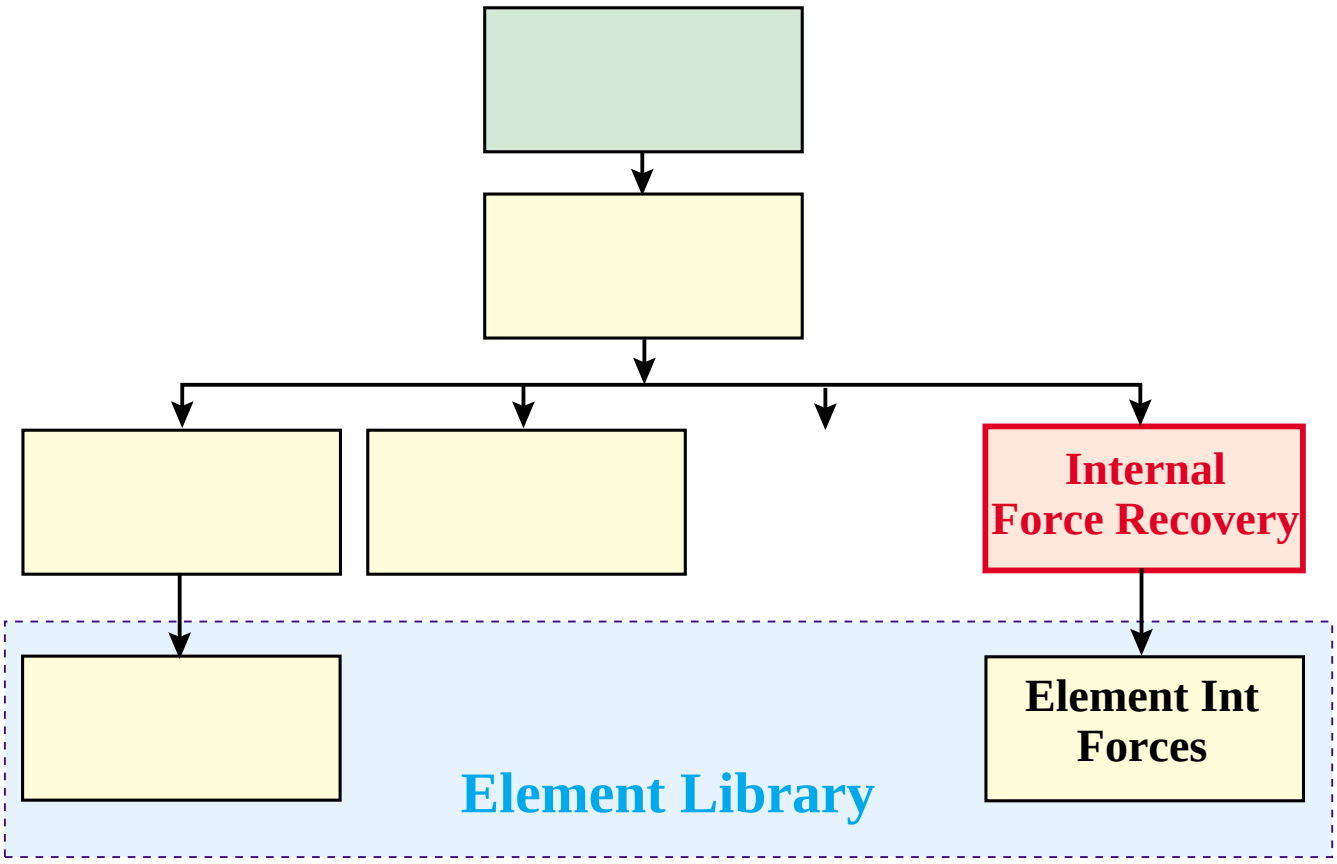
Modified Force Vector:

$$\begin{pmatrix} & v1 & & \\ & v2 & & \\ f[3] & -v1\,K[1,3] & -v2\,K[2,3] & -v4\,K[4,3] \\ & v4 & & \\ f[5] & -v1\,K[1,5] & -v2\,K[2,5] & -v4\,K[4,5] \\ f[6] & -v1\,K[1,6] & -v2\,K[2,6] & -v4\,K[4,6] \end{pmatrix}$$

Equation Solver Need Not be Discussed



Next: Internal Force Recovery



Internal Force Recovery

```
SpaceTrussIntForces[nodxyz_,elenod_,elemat_,elefab_,
  noddis_,prcopt_]:= Module[{  numnod=Length[nodxyz],
  numele=Length[elenod],e,ni,nj,ncoor,Em,A,options,ue,p},
  p=Table[0,{numele}];
  For [e=1, e<=numele, e++, {ni,nj}=elenod[[e]];
      ncoor={nodxyz[[ni]],nodxyz[[nj]]};
      ue=Flatten[{ noddis[[ni]],noddis[[nj]] }];
      Em=elemat[[e]]; A=elefab[[e]]; options=prcopt;
      p[[e]]=SpaceBar2IntForce[ncoor,Em,A,ue,options]
      ];
  Return[p]];
```

Element Level Internal Force Recovery

```
SpaceBar2IntForce[ncoor_,Em_,A_,ue_,options_]:= Module[
 {x1,x2,y1,y2,z1,z2,x21,y21,z21,EA,numer,LL,pe},
 {{x1,y1,z1},{x2,y2,z2}}=ncoor;{x21,y21,z21}={x2-x1,y2-y1,z2-z1};
  EA=Em*A; {numer}=options;  LL=x21^2+y21^2+z21^2;
  If [numer,{x21,y21,z21,EA,LL}=N[{x21,y21,z21,EA,LL}]];
  pe=(EA/LL)*(x21*(ue[[4]]-ue[[1]])+y21*(ue[[5]]-ue[[2]])+
             +z21*(ue[[6]]-ue[[3]]));
```

This is part of the element library

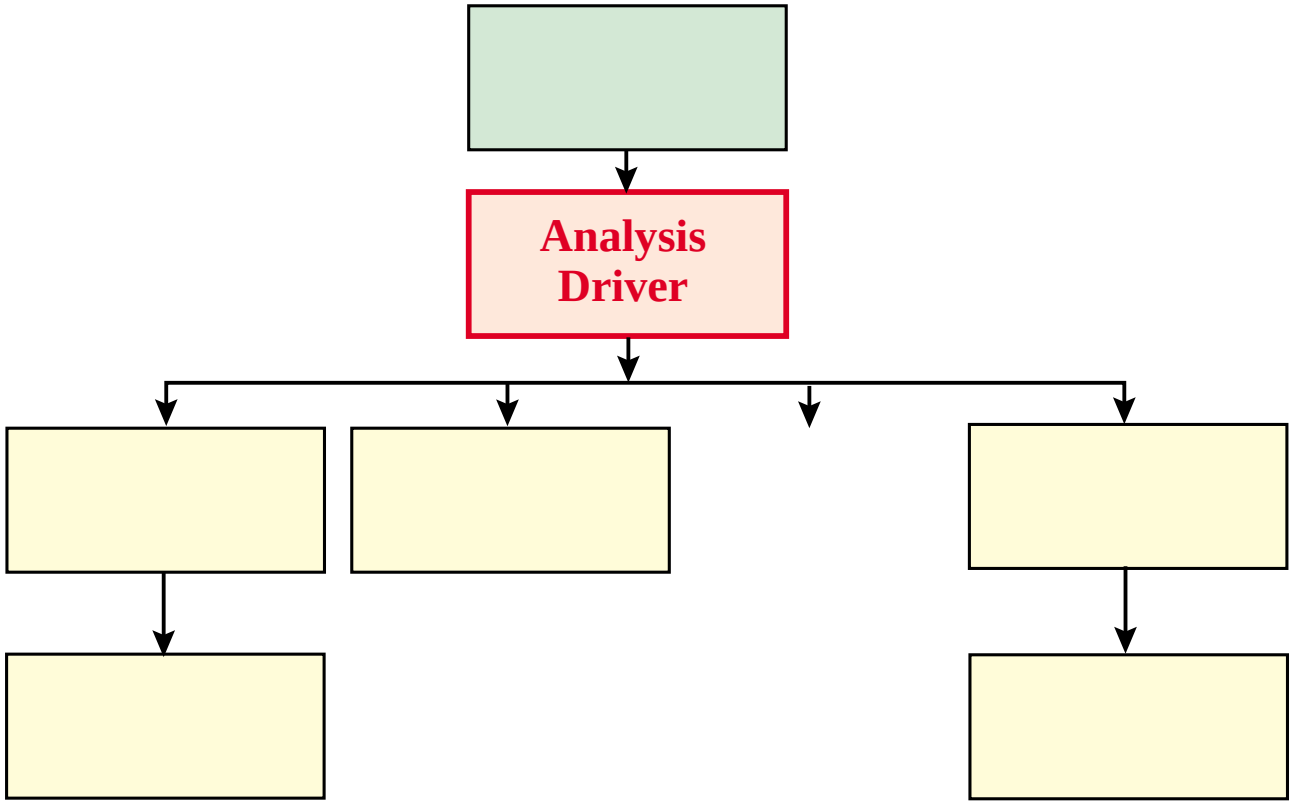
Space Truss Internal Force Recovery Tester

```
ClearAll[nodxyz,elenod,elemat,elefab,noddis];
nodxyz={{0,0,0},{10,0,0},{10,10,0}}; elenod={{1,2},{2,3},{1,3}};
elemat= Table[100,{3}]; elefab= {1,1/2,2*Sqrt[2]};
noddis={{0,0,0}, {0,0,0}, {4/10,-2/10,0}}; prcopt={False};
elefor=SpaceTrussIntForces[nodxyz,elenod,elemat,elefab,noddis,prcopt];
Print["Int Forces of Example Truss:",elefor];
Print["Stresses:",SpaceTrussStresses[elefab,elefor,prcopt]];
```

```
Int Forces of Example Truss: { 0, –1,  2*Sqrt[2]  }
Stresses: { 0, –2, 1 }
```

This script also tests module **SpaceTrussStresses**, not shown in these slides as it is very simple

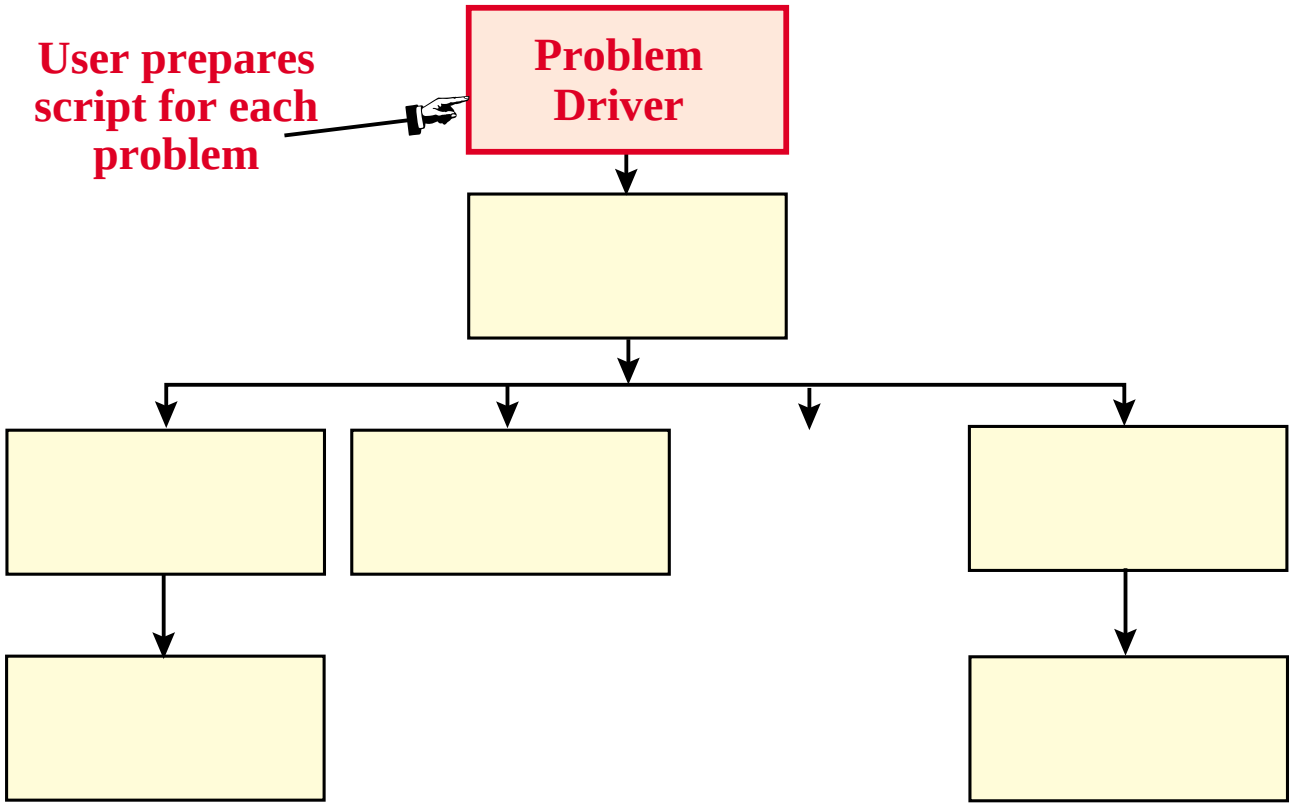
Next: Analysis Driver



Analysis Driver Module

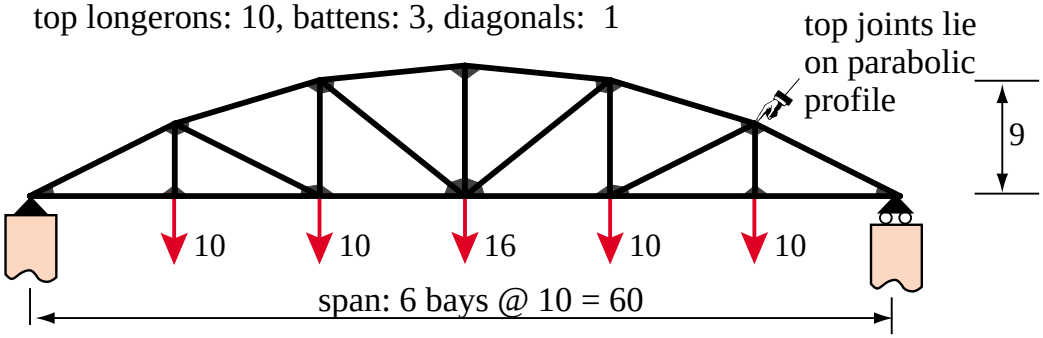
```
SpaceTrussSolution[nodxyz_,elenod_,elemat_,elefab_,nodtag_,nodval_,
   prcopt_]:= Module[{K,Kmod,f,fmod,u,noddis,nodfor,elefor,elesig},
   K=SpaceTrussMasterStiffness[nodxyz,elenod,elemat,elefab,prcopt];
   (* Print["eigs of K=",Chop[Eigenvalues[N[K]]]]; *)
   Kmod=ModifiedMasterStiffness[nodtag,K];
   f=FlatNodePartVector[nodval];
   fmod=ModifiedNodeForces[nodtag,nodval,K,f];
   (* Print["eigs of Kmod=",Chop[Eigenvalues[N[Kmod]]]]; *)
   u=LinearSolve[Kmod,fmod]; u=Chop[u]; f=Chop[K.u, 10.0^(-8)];
   nodfor=NodePartFlatVector[3,f]; noddis=NodePartFlatVector[3,u];
   elefor=Chop[SpaceTrussIntForces[nodxyz,elenod,elemat,elefab,
               noddis,prcopt]];
   elesig=SpaceTrussStresses[elefab,elefor,prcopt];
   Return[{noddis,nodfor,elefor,elesig}];
];
```

Next: Problem Driver

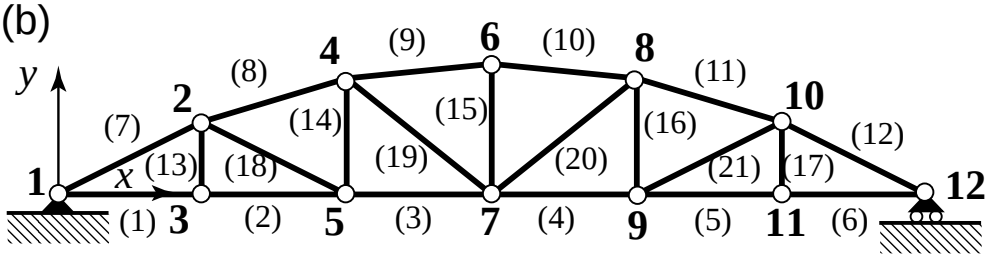


Six-bay Bridge Truss Example
(A Plane Truss, but Modeled in 3D)

(a) Elastic modulus E = 1000
Cross section areas of bottom longerons: 2,
top longerons: 10, battens: 3, diagonals: 1



(b)



z out of paper toward you

Bridge Problem Driver: PreProcessing Script (top of driver cell)

```
NodeCoordinates={{0,0,0},{10,5,0},{10,0,0},{20,8,0},{20,0,0},{30,9,0},
            {30,0,0},{40,8,0},{40,0,0},{50,5,0},{50,0,0},{60,0,0}};
ElemNodes={{1,3},{3,5},{5,7},{7,9},{9,11},{11,12},
           {1,2},{2,4},{4,6},{6,8},{8,10},{10,12},
           {2,3},{4,5},{6,7},{8,9},{10,11},
           {2,5},{4,7},{7,8},{9,10}};
PrintSpaceTrussNodeCoordinates[NodeCoordinates,"Node coordinates:",{}];
numnod=Length[NodeCoordinates];  numele=Length[ElemNodes];
Em=1000; Abot=2; Atop=10; Abat=3; Adia=1;
ElemMaterials= Table[Em,{numele}];
ElemFabrications={Abot,Abot,Abot,Abot,Abot,Abot,Atop,Atop,Atop,Atop,
                  Atop,Atop,Abat,Abat,Abat,Abat,Abat,Adia,Adia,Adia,Adia};
PrintSpaceTrussElementData[ElemNodes,ElemMaterials,ElemFabrications,
           "Element data:",{}];
ProcessOptions= {True};

(*  Plot statements omitted - interface being changed *)

NodeDOFTags=  Table[{0,0,1},{numnod}];
NodeDOFValues=Table[{0,0,0},{numnod}];
NodeDOFValues[[3]]={0,-10,0}; NodeDOFValues[[5]]= {0,-10,0};
NodeDOFValues[[7]]={0,-16,0};
NodeDOFValues[[9]]={0,-10,0}; NodeDOFValues[[11]]={0,-10,0};
NodeDOFTags[[1]]= {1,1,1};     (* fixed node 1 *)
NodeDOFTags[[numnod]]={0,1,1}; (* hroller @ node 12 *)
PrintSpaceTrussFreedomActivity[NodeDOFTags,NodeDOFValues,
       "DOF Activity:",{}];
```

Bridge Problem Driver: Preprocessing Print Output

Node coordinates:

node	x-coor	y-coor	z-coor
1	0.000000	0.000000	0.000000
2	10.000000	5.000000	0.000000
3	10.000000	0.000000	0.000000
4	20.000000	8.000000	0.000000
5	20.000000	0.000000	0.000000
6	30.000000	9.000000	0.000000
7	30.000000	0.000000	0.000000
8	40.000000	8.000000	0.000000
9	40.000000	0.000000	0.000000
10	50.000000	5.000000	0.000000
11	50.000000	0.000000	0.000000
12	60.000000	0.000000	0.000000

Element data:

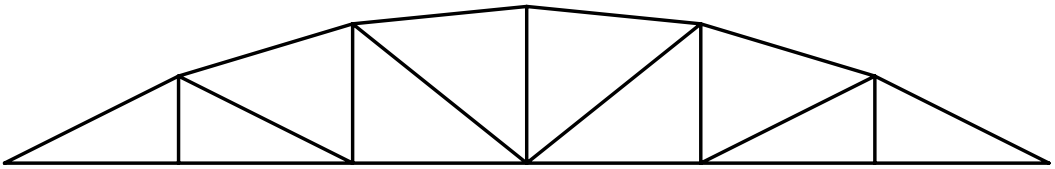
elem	nodes	modulus	area
1	{1, 3}	1000.00	2.00
2	{3, 5}	1000.00	2.00
3	{5, 7}	1000.00	2.00
4	{7, 9}	1000.00	2.00
5	{9, 11}	1000.00	2.00
6	{11, 12}	1000.00	2.00
7	{1, 2}	1000.00	10.00
8	{2, 4}	1000.00	10.00
9	{4, 6}	1000.00	10.00
10	{6, 8}	1000.00	10.00
11	{8, 10}	1000.00	10.00
12	{10, 12}	1000.00	10.00
13	{2, 3}	1000.00	3.00
14	{4, 5}	1000.00	3.00
15	{6, 7}	1000.00	3.00
16	{8, 9}	1000.00	3.00
17	{10, 11}	1000.00	3.00
18	{2, 5}	1000.00	1.00
19	{4, 7}	1000.00	1.00
20	{7, 8}	1000.00	1.00
21	{9, 10}	1000.00	1.00

DOF Activity:

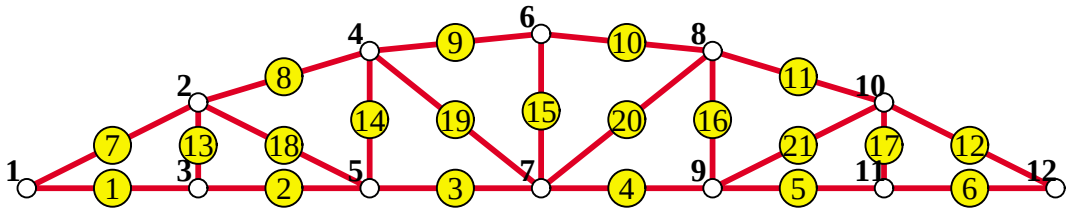
node	x-tag	y-tag	z-tag	x-value	y-value	z-value
1	1	1	1	0	0	0
2	0	0	1	0	0	0
3	0	0	1	0	-10	0
4	0	0	1	0	0	0
5	0	0	1	0	-10	0
6	0	0	1	0	0	0
7	0	0	1	0	-16	0
8	0	0	1	0	0	0
9	0	0	1	0	-10	0
10	0	0	1	0	0	0
11	0	0	1	0	-10	0
12	0	1	1	0	0	0

Bridge Problem Driver: Preprocessing Plots

bridge mesh



bridge mesh with elem & node labels



Bridge Problem Driver: Processing & PostProcessing Script
(bottom of driver cell)

```
{NodeDisplacements,NodeForces,ElemForces,ElemStresses}=
   SpaceTrussSolution[ NodeCoordinates,ElemNodes,ElemMaterials,
   ElemFabrications, NodeDOFTags, NodeDOFValues,ProcessOptions ];

PrintSpaceTrussNodeDisplacements[NodeDisplacements,
      "Computed node displacements:",{}];
PrintSpaceTrussNodeForces[NodeForces,
      "Node forces including reactions:",{}];
PrintSpaceTrussElemForcesAndStresses[ElemForces,ElemStresses,
      "Int Forces and Stresses:",{}];

(*  Plot statements omitted - interface being changed *)
```

Bridge Problem Driver: Result Print Output

Computed node displacements:

node	x-displ	y-displ	z-displ
1	0.000000	0.000000	0.000000
2	0.809536	-1.775600	0.000000
3	0.280000	-1.792260	0.000000
4	0.899001	-2.291930	0.000000
5	0.560000	-2.316600	0.000000
6	0.847500	-2.385940	0.000000
7	0.847500	-2.421940	0.000000
8	0.795999	-2.291930	0.000000
9	1.135000	-2.316600	0.000000
10	0.885464	-1.775600	0.000000
11	1.415000	-1.792260	0.000000
12	1.695000	0.000000	0.000000

Node forces including reactions:

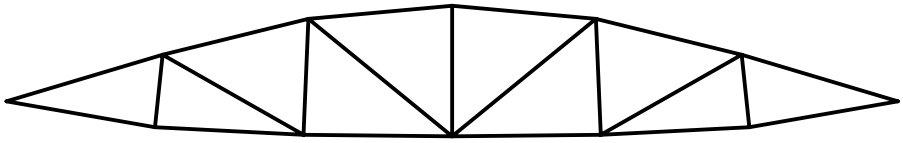
node	x-force	y-force	z-force
1	0.0000	28.0000	0.0000
2	0.0000	0.0000	0.0000
3	0.0000	-10.0000	0.0000
4	0.0000	0.0000	0.0000
5	0.0000	-10.0000	0.0000
6	0.0000	0.0000	0.0000
7	0.0000	-16.0000	0.0000
8	0.0000	0.0000	0.0000
9	0.0000	-10.0000	0.0000
10	0.0000	0.0000	0.0000
11	0.0000	-10.0000	0.0000
12	0.0000	28.0000	0.0000

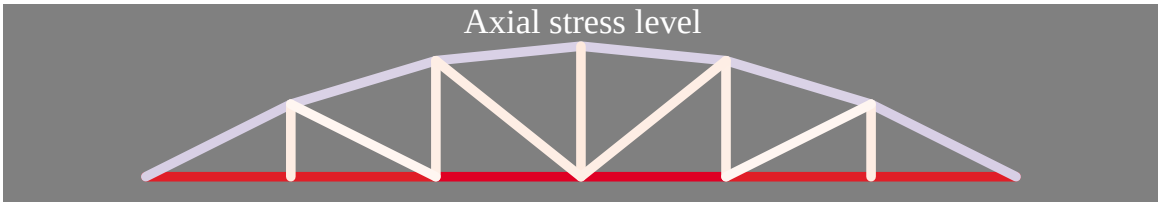
Int Forces and Stresses:

elem	axial force	axial stress
1	56.0000	28.0000
2	56.0000	28.0000
3	57.5000	28.7500
4	57.5000	28.7500
5	56.0000	28.0000
6	56.0000	28.0000
7	-62.6100	-6.2610
8	-60.0300	-6.0030
9	-60.3000	-6.0300
10	-60.3000	-6.0300
11	-60.0300	-6.0030
12	-62.6100	-6.2610
13	10.0000	3.3330
14	9.2500	3.0830
15	12.0000	4.0000
16	9.2500	3.0830
17	10.0000	3.3330
18	1.6770	1.6770
19	3.2020	3.2020
20	3.2020	3.2020
21	1.6770	1.6770

Bridge Problem Driver: Result Plot Output

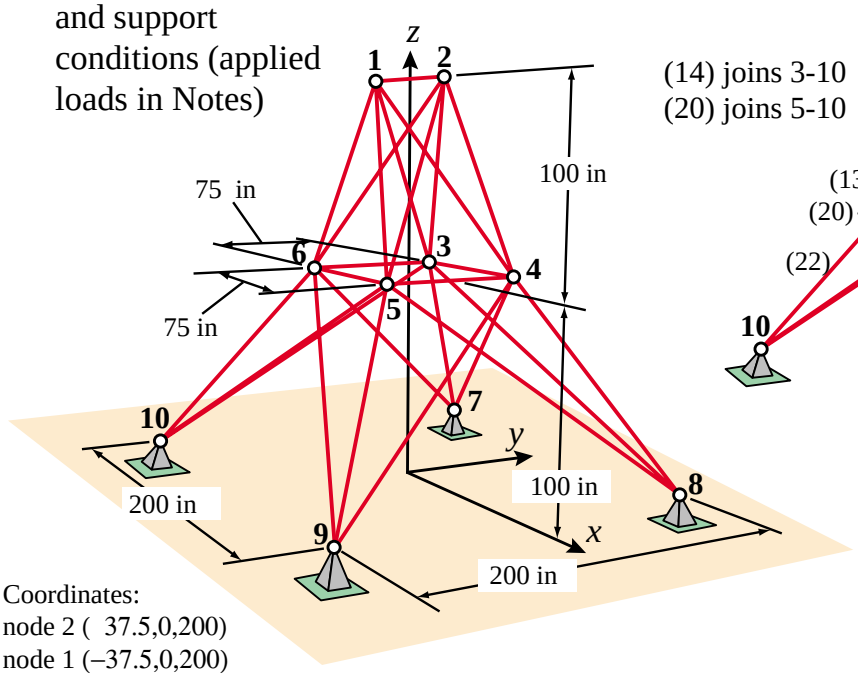
Deformed shape





Space Truss for Homework: 25-Member Transmission Tower

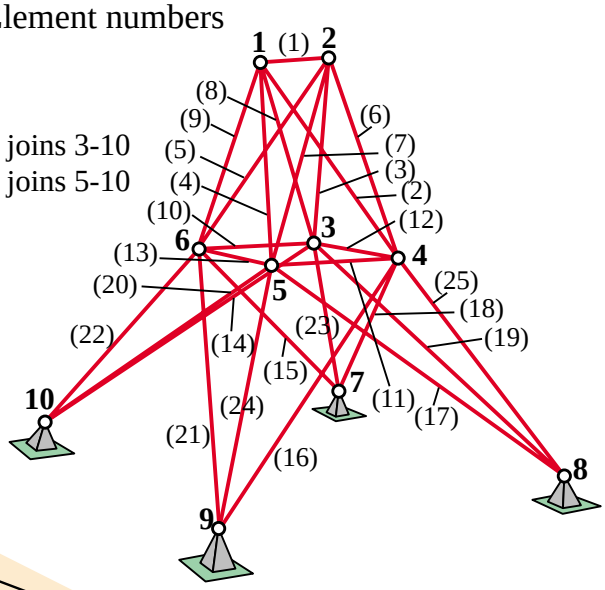
(a) Geometry, node numbers and support conditions (applied loads in Notes)



Coordinates:
node 2 (37.5,0,200)
node 1 (–37.5,0,200)

(b) Element numbers

(14) joins 3-10
(20) joins 5-10



Material: Aluminum.
Modulus and cross section properties in Notes