

# 4

## Analysis of Example Truss by a CAS

# **Computer Algebra Systems: What Are They?**

**Computer tools to help humans  
solve math problems**

**Especially: algebra  
calculus**

**Why do humans need help?**

## **The Three Big Ma's in order of appearance**

**Macsyma** (died 1999, continued as free source)

**Maple**

**Mathematica**

**Matlab is *not* a CAS**

**("Ma" in Matlab stands for Matrix, not Math)**

**And a host of minor players: Mathcad, Magma, ... etc**

## **Why Mathematica?**

### ***Strong points***

#### **Implemented on many platforms**

(Maple is primarily used under Unix, does not have a Mac version)

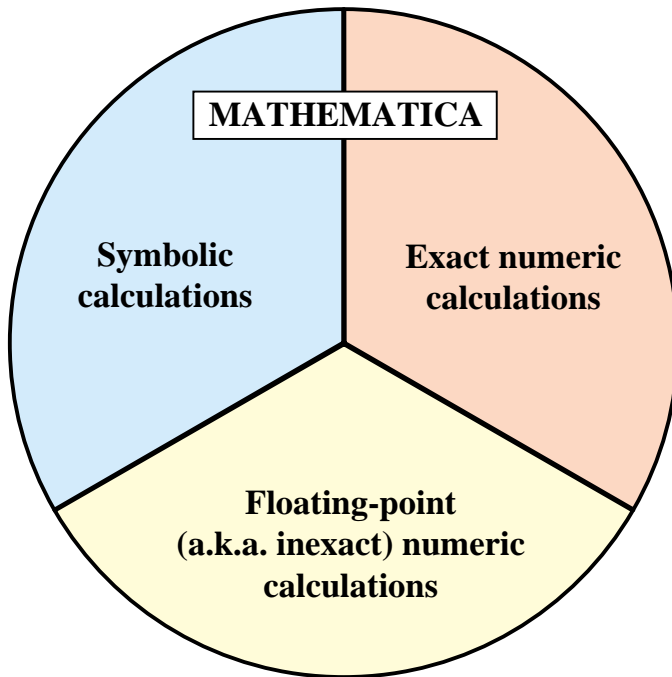
#### **Programming facilities**

**Graphics (greatly improved in Versions 6-7)**

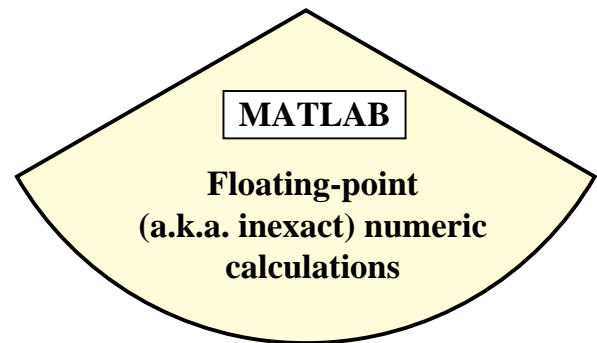
**Good documentation & many application books**

**Since 1 August 2007: 1-year renewable campus license, free to students, faculty and staff**

## Mathematica vs. Matlab Coverage



Source code (1991): C, then C++



Source code (1980s): Fortran

## **Popularity Contest by Number of "Hits" on www3.addall.com Book Search Engine (Sep 2003)**

	<b>addall</b>	<b>google (M)</b>
<b>Macsyma</b>	<b>9</b>	<b>.1</b>
<b>Maple</b>	<b>111</b>	<b>2</b>
<b>Mathematica</b>	<b>150</b>	<b>10</b>
<b>Matlab</b>	<b>148</b>	<b>21</b>

**Note: C++ 191, Java 189, Fortran 145 in addall**

## **Last Major Releases of *Mathematica***

**Version 2.2 1994**

**Version 3.0 1997**

**Version 4.0 1999**

**4.1 2001**

**4.2 2002**

**Version 5.0 2003**

**5.1 2004**

**5.2 2005**

**Version 6.0 2007**

**Version 7.0 2009**

**Version 8.0 2011**

**Eligibility:** students, faculty, staff and departments of all CU campuses

**Platforms:** Mac OSX, Windows, Linux, Solaris, AIX, HP-UX

### **How to Get It**

- 1) Download and install software as per instructions at <http://oit.colorado.edu/software-hardware/site-licenses/mathematica> or request an installation CD from the Site Licensing office: [sitelic@colorado.edu](mailto:sitelic@colorado.edu), 303-492-8995
- 2) Register your copy online at <http://register.wolfram.com/> using campus license number L2437-5121 plus your computer MathID number. Be sure to use your CU mail address.
- 3) A password (unique to your computer) will be forwarded to you via email from Site Licensing

### **ITS Support**

Question & tech support problems: send email to [sitelic@colorado.edu](mailto:sitelic@colorado.edu)  
Periodic hands-on workshops are available for those new to the software, click on [Workshops](#) on the Web page given above.  
Details (for example: what is a MathID?) about licensing & support are posted at the above web site.

## Class Demo Cell #1

Integration example

```
f[x_,α_,β_] := (1+β*x^2)/(1+α*x+x^2);  
F=Integrate[f[x,-1,2],{x,0,5}];  
F=Simplify[F];  
Print[F]; Print[N[F]];  
F=NIntegrate[f[x,-1,2],{x,0,5}];  
Print["F=",F//InputForm];
```

**Input Cell**

10 + Log[21]

13.0445

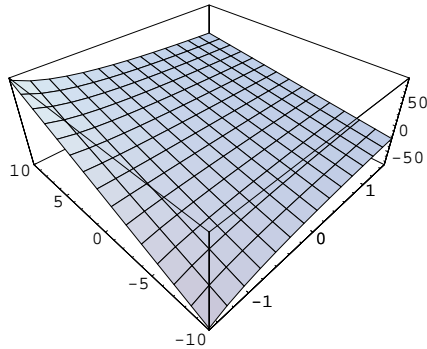
F=13.044522437723455

**Output Cells**

# Class Demo Cell #2

```
Fa=Integrate[f[z,a,b],{z,0,5}]; Fa=Simplify[Fa]; Print["Fa=",Fa];
Plot3D[Fa,{a,-1.5,1.5},{b,-10,10},ViewPoint->{-1,-1,1}];
Fa=FullSimplify[Fa]; (* very slow but you get *) Print["Fa=",Fa];
```

$$\begin{aligned}
 Fa = & \frac{1}{2\sqrt{4-a^2}} \left( 10\sqrt{4-a^2} b - a\sqrt{4-a^2} b \operatorname{Log}[26+5a] - i(2+(-2+a^2)b) \operatorname{Log}\left[1 - \frac{ia}{\sqrt{4-a^2}}\right] + \right. \\
 & 2i \operatorname{Log}\left[1 + \frac{ia}{\sqrt{4-a^2}}\right] - 2ib \operatorname{Log}\left[1 + \frac{ia}{\sqrt{4-a^2}}\right] + ia^2 b \operatorname{Log}\left[1 + \frac{ia}{\sqrt{4-a^2}}\right] + \\
 & 2i \operatorname{Log}\left[\frac{-10i - ia + \sqrt{4-a^2}}{\sqrt{4-a^2}}\right] - 2ib \operatorname{Log}\left[\frac{-10i - ia + \sqrt{4-a^2}}{\sqrt{4-a^2}}\right] + ia^2 b \operatorname{Log}\left[\frac{-10i - ia + \sqrt{4-a^2}}{\sqrt{4-a^2}}\right] - \\
 & \left. 2i \operatorname{Log}\left[\frac{10i + ia + \sqrt{4-a^2}}{\sqrt{4-a^2}}\right] + 2ib \operatorname{Log}\left[\frac{10i + ia + \sqrt{4-a^2}}{\sqrt{4-a^2}}\right] - ia^2 b \operatorname{Log}\left[\frac{10i + ia + \sqrt{4-a^2}}{\sqrt{4-a^2}}\right] \right)
 \end{aligned}$$



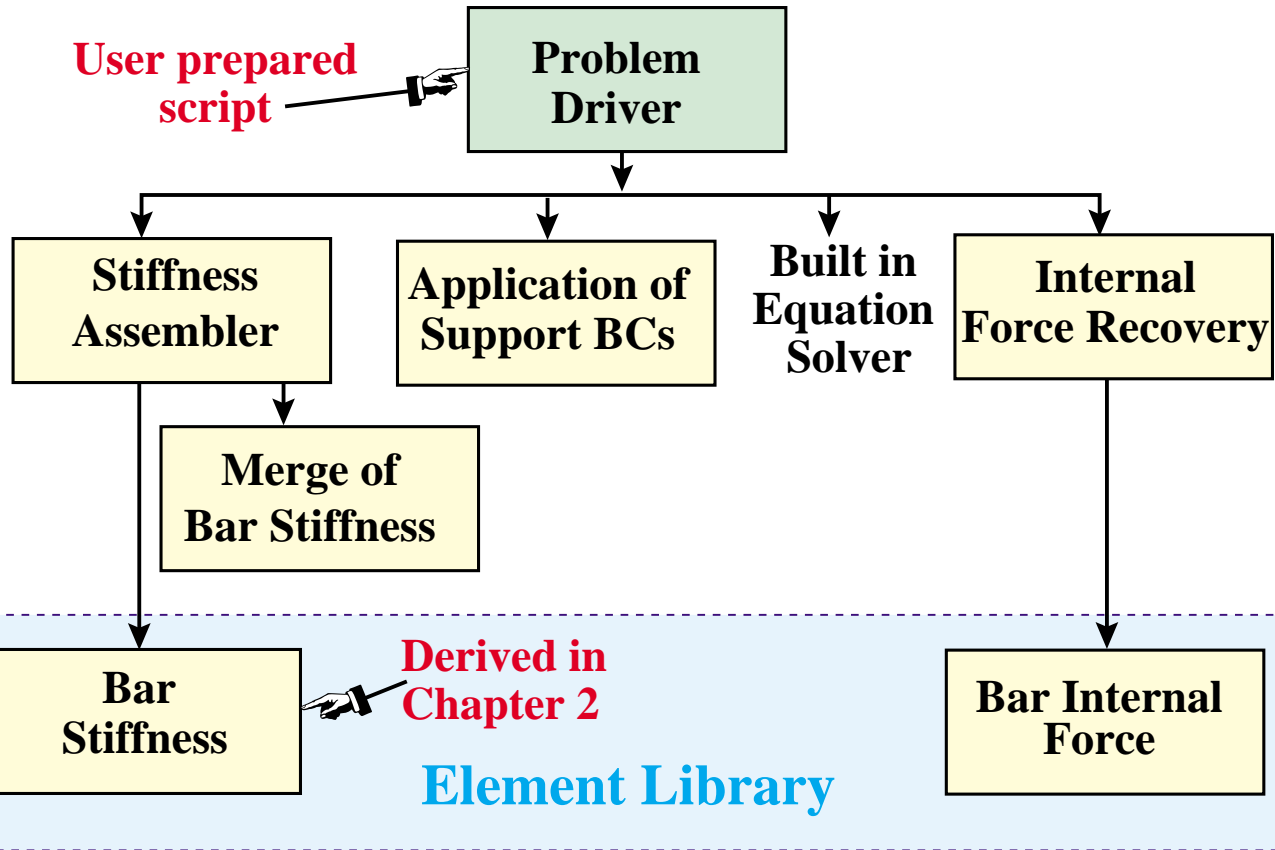
**Result after Simplify[ .. ]**

**Result after FullSimplify[ .. ]**

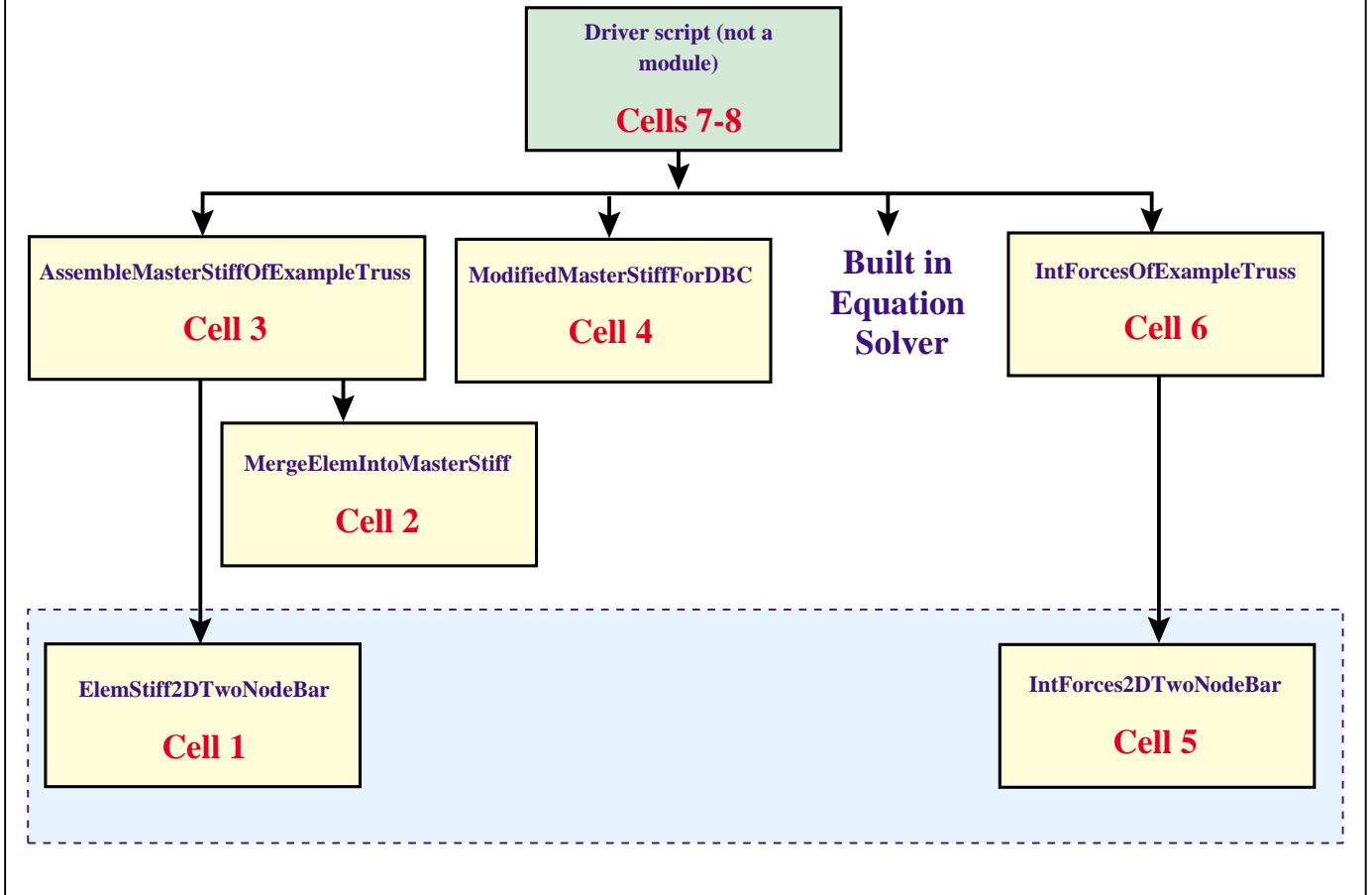
$$\begin{aligned}
 Fa = & 5b - \frac{1}{2} a b \operatorname{Log}[26+5a] - \\
 & \frac{i(2+(-2+a^2)b) \left( \operatorname{Log}\left[1 - \frac{ia}{\sqrt{4-a^2}}\right] - \operatorname{Log}\left[1 + \frac{ia}{\sqrt{4-a^2}}\right] - \operatorname{Log}\left[1 - \frac{i(10+a)}{\sqrt{4-a^2}}\right] + \operatorname{Log}\left[1 + \frac{i(10+a)}{\sqrt{4-a^2}}\right] \right)}{2\sqrt{4-a^2}}
 \end{aligned}$$

# Example Truss Program Configuration

User prepared script



# Example Truss Program Module Names



# Module to Form Element Stiffness Matrix of 2D Two-Node Bar

```

ElemStiff2DTwoNodeBar[{{x1_,y1_},{x2_,y2_}},{Em_,A_}] :=
Module[{c,s,dx=x2-x1,dy=y2-y1,L,Ke},
L=Sqrt[dx^2+dy^2]; c=dx/L; s=dy/L;
Ke=(Em*A/L)* {{ c^2, c*s,-c^2,-c*s},
               { c*s, s^2,-s*c,-s^2},
               {-c^2,-s*c, c^2, s*c},
               {-s*c,-s^2, s*c, s^2}};

Return[Ke]
];
Ke= ElemStiff2DTwoNodeBar[{{0,0},{10,10}},{100,2*Sqrt[2]}];
Print["Numerical elem stiff matrix:"]; Print[Ke//MatrixForm];
Ke= ElemStiff2DTwoNodeBar[{{0,0},{L,L}},{Em,A}];
Ke=Simplify[Ke,L>0];
Print["Symbolic elem stiff matrix:"]; Print[Ke//MatrixForm];

```

Numerical elem stiff matrix:

$$\begin{pmatrix} 10 & 10 & -10 & -10 \\ 10 & 10 & -10 & -10 \\ -10 & -10 & 10 & 10 \\ -10 & -10 & 10 & 10 \end{pmatrix}$$

Symbolic elem stiff matrix:

$$\begin{pmatrix} \frac{A Em}{2\sqrt{2} L} & \frac{A Em}{2\sqrt{2} L} & -\frac{A Em}{2\sqrt{2} L} & -\frac{A Em}{2\sqrt{2} L} \\ \frac{A Em}{2\sqrt{2} L} & \frac{A Em}{2\sqrt{2} L} & -\frac{A Em}{2\sqrt{2} L} & -\frac{A Em}{2\sqrt{2} L} \\ -\frac{A Em}{2\sqrt{2} L} & -\frac{A Em}{2\sqrt{2} L} & \frac{A Em}{2\sqrt{2} L} & \frac{A Em}{2\sqrt{2} L} \\ -\frac{A Em}{2\sqrt{2} L} & -\frac{A Em}{2\sqrt{2} L} & \frac{A Em}{2\sqrt{2} L} & \frac{A Em}{2\sqrt{2} L} \end{pmatrix}$$

# Module to Merge Element Stiffness into Master Stiffness

```

MergeElemIntoMasterStiff[Ke_,eftab_,Kin_]:=Module[
  {i,j,ii,jj,K=Kin},
  For [i=1, i<=4, i++, ii=eftab[[i]];
    For [j=i, j<=4, j++, jj=eftab[[j]];
      K[[jj,ii]]=K[[ii,jj]]+=Ke[[i,j]]
    ]
  ]; Return[K]
];
K=Table[0,{6},{6}];
Print["Initialized master stiffness matrix:"];
Print[K//MatrixForm]
Ke=ElemStiff2DTwoNodeBar[{{0,0},{10,10}},{100,2*Sqrt[2]}];
Print["Member stiffness matrix:"]; Print[Ke//MatrixForm];
K=MergeElemIntoMasterStiff[Ke,{1,2,5,6},K];
Print["Master stiffness after member merge:"];
Print[K//MatrixForm];

```

Initialized master stiffness matrix:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Member stiffness matrix:

$$\begin{pmatrix} 10 & 10 & -10 & -10 \\ 10 & 10 & -10 & -10 \\ -10 & -10 & 10 & 10 \\ -10 & -10 & 10 & 10 \end{pmatrix}$$

Master stiffness after member merge:

$$\begin{pmatrix} 10 & 10 & 0 & 0 & -10 & -10 \\ 10 & 10 & 0 & 0 & -10 & -10 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -10 & -10 & 0 & 0 & 10 & 10 \\ -10 & -10 & 0 & 0 & 10 & 10 \end{pmatrix}$$

## Module to Form Master Stiffness Matrix of Example Truss

```

AssembleMasterStiffOfExampleTruss[]:=
Module[{Ke,K=Table[0,{6},{6}]},
  Ke=ElemStiff2DTwoNodeBar[{{0,0},{10,0}},{100,1}];
  K= MergeElemIntoMasterStiff[Ke,{1,2,3,4},K];
  Ke=ElemStiff2DTwoNodeBar[{{10,0},{10,10}},{100,1/2}];
  K= MergeElemIntoMasterStiff[Ke,{3,4,5,6},K];
  Ke=ElemStiff2DTwoNodeBar[{{0,0},{10,10}},{100,2*Sqrt[2]}];
  K= MergeElemIntoMasterStiff[Ke,{1,2,5,6},K];
  Return[K]
];
K=AssembleMasterStiffOfExampleTruss[];
Print["Master stiffness of example truss:"]; Print[K/MatrixForm];

```

Master stiffness of example truss:

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

# Modules to Modify Master Stiffness & Forces for Displacement BC

*Introduction to FEM*

```

ModifiedMasterStiffForDBC[pdof_,K_] := Module[
  {i,j,k,nk=Length[K],np=Length[pdof],Kmod=K},
  For [k=1,k<=np,k++, i=pdof[[k]];
    For [j=1,j<=nk,j++, Kmod[[i,j]]=Kmod[[j,i]]=0];
    Kmod[[i,i]]=1];
  Return[Kmod]
];
ModifiedMasterForcesForDBC[pdof_,f_] := Module[
  {i,k,np=Length[pdof],fmod=f},
  For [k=1,k<=np,k++, i=pdof[[k]]; fmod[[i]]=0];
  Return[fmod]
];
K=Array[Kij,{6,6}]; Print["Assembled master stiffness:"];
Print[K//MatrixForm];
K=ModifiedMasterStiffForDBC[{1,2,4},K];
Print["Master stiffness modified for displacement B.C.:"];
Print[K//MatrixForm];
f=Array[fi,{6}]; Print["Force vector:"]; Print[f];
f=ModifiedMasterForcesForDBC[{1,2,4},f];
Print["Force vector modified for displacement B.C.:"]; Print[f];

```

```

Assembled master stiffness:
( Kij[1,1] Kij[1,2] Kij[1,3] Kij[1,4] Kij[1,5] Kij[1,6] )
( Kij[2,1] Kij[2,2] Kij[2,3] Kij[2,4] Kij[2,5] Kij[2,6] )
( Kij[3,1] Kij[3,2] Kij[3,3] Kij[3,4] Kij[3,5] Kij[3,6] )
( Kij[4,1] Kij[4,2] Kij[4,3] Kij[4,4] Kij[4,5] Kij[4,6] )
( Kij[5,1] Kij[5,2] Kij[5,3] Kij[5,4] Kij[5,5] Kij[5,6] )
( Kij[6,1] Kij[6,2] Kij[6,3] Kij[6,4] Kij[6,5] Kij[6,6] )
Master stiffness modified for displacement B.C.:
( 1 0 0 0 0 0 )
( 0 1 0 0 0 0 )
( 0 0 Kij[3,3] 0 Kij[3,5] Kij[3,6] )
( 0 0 0 1 0 0 )
( 0 0 Kij[5,3] 0 Kij[5,5] Kij[5,6] )
( 0 0 Kij[6,3] 0 Kij[6,5] Kij[6,6] )
Force vector:
{fi[1], fi[2], fi[3], fi[4], fi[5], fi[6]}
Force vector modified for displacement B.C.:
{0, 0, fi[3], 0, fi[5], fi[6]}

```

## Module to Compute Internal Force in 2D Two-Node Bar Element

```

IntForce2DTwoNodeBar[{{x1_,y1_},{x2_,y2_}},{Em_,A_},eftab_,u_]:=
Module[ {c,s,dx=x2-x1,dy=y2-y1,L,ix,iy,jx,jy,ubar,e},
  L=Sqrt[dx^2+dy^2]; c=dx/L; s=dy/L; {ix,iy,jx,jy}=eftab;
  ubar={c*u[[ix]]+s*u[[iy]],-s*u[[ix]]+c*u[[iy]],
        c*u[[jx]]+s*u[[jy]],-s*u[[jx]]+c*u[[jy]]};
  e=(ubar[[3]]-ubar[[1]])/L; Return[Em*A*e]
];
p =IntForce2DTwoNodeBar[{{0,0},{10,10}},{100,2*Sqrt[2]},
  {1,2,5,6},{0,0,0,0,0.4,-0.2}];
Print["Member int force (numerical):"]; Print[N[p]];
p =IntForce2DTwoNodeBar[{{0,0},{L,L}},{Em,A},
  {1,2,5,6},{0,0,0,0,ux3,uy3}];
Print["Member int force (symbolic):"]; Print[Simplify[p]];

```

```

Member int force (numerical):
2.82843
Member int force (symbolic):
  A Em (ux3 + uy3)
    2 L

```

## Module to Compute Internal Forces in Example Truss

```
IntForcesOfExampleTruss[u_]:= Module[{f=Table[0,{3}]},  
  f[[1]]=IntForce2DTwoNodeBar[{{0,0},{10,0}},{100,1},{1,2,3,4},u];  
  f[[2]]=IntForce2DTwoNodeBar[{{10,0},{10,10}},{100,1/2},{3,4,5,6},u];  
  f[[3]]=IntForce2DTwoNodeBar[{{0,0},{10,10}},{100,2*Sqrt[2]},  
    {1,2,5,6},u];  
  Return[f]  
];  
f=IntForcesOfExampleTruss[{0,0,0,0,0.4,-0.2}];  
Print["Internal member forces in example truss:"];Print[N[f]];
```

```
Internal member forces in example truss:  
{0., -1., 2.82843}
```

## Driver Program to Analyze Example Truss with All-Numerical Data

```
f={0,0,0,0,2,1};  
K=AssembleMasterStiffOfExampleTruss[];  
Kmod=ModifiedMasterStiffForDBC[{1,2,4},K];  
fmod=ModifiedMasterForcesForDBC[{1,2,4},f];  
u=Simplify[Inverse[Kmod].fmod];  
Print["Computed nodal displacements:"]; Print[u];  
f=Simplify[K.u];  
Print["External node forces including reactions:"]; Print[f];  
p=Simplify[IntForcesOfExampleTruss[u]];  
Print["Internal member forces:"]; Print[p];
```

```
Computed nodal displacements:  
{0, 0, 0, 0,  $\frac{2}{5}$ ,  $-\frac{1}{5}$ }  
External node forces including reactions:  
{-2, -2, 0, 1, 2, 1}  
Internal member forces:  
{0, -1,  $2\sqrt{2}$ }
```

## Driver Program to Analyze Example Truss with Symbolic Forces

```
f={0,0,0,0,fx3,fy3};
K=AssembleMasterStiffOfExampleTruss[];
Kmod=ModifiedMasterStiffForDBC[{1,2,4},K];
fmod=ModifiedMasterForcesForDBC[{1,2,4},f];
u=Simplify[Inverse[Kmod].fmod];
Print["Computed nodal displacements:"]; Print[u];
f=Simplify[K.u];
Print["External node forces including reactions:"]; Print[f];
p=Simplify[IntForcesOfExampleTruss[u]];
Print["Internal member forces:"]; Print[p];
```

Computed nodal displacements:

$\{0, 0, 0, 0, \frac{1}{10} (3 fx3 - 2 fy3), \frac{1}{5} (-fx3 + fy3)\}$

External node forces including reactions:

$\{-fx3, -fx3, 0, fx3 - fy3, fx3, fy3\}$

Internal member forces:

$\{0, -fx3 + fy3, \sqrt{2} fx3\}$

# Hierarchical Organization of Example Truss Program

**Driver program (samples in Cells 7 & 8)**

**AssembleMasterStiffOfExampleTruss**

**ElemStiff2DTwoNodeBar**

**MergeElemIntoMasterStiff**

**ModifiedMasterStiffForDBC**

**ModifiedMasterForcesForDBC**

**IntForcesOfExampleTruss**

**IntForce2DTwoNodeTruss**

**TASK**

**Assembly**

**Globalization**

**Merge**

**Apply BCs**

**[Solve: by library]**

**Postprocessing**