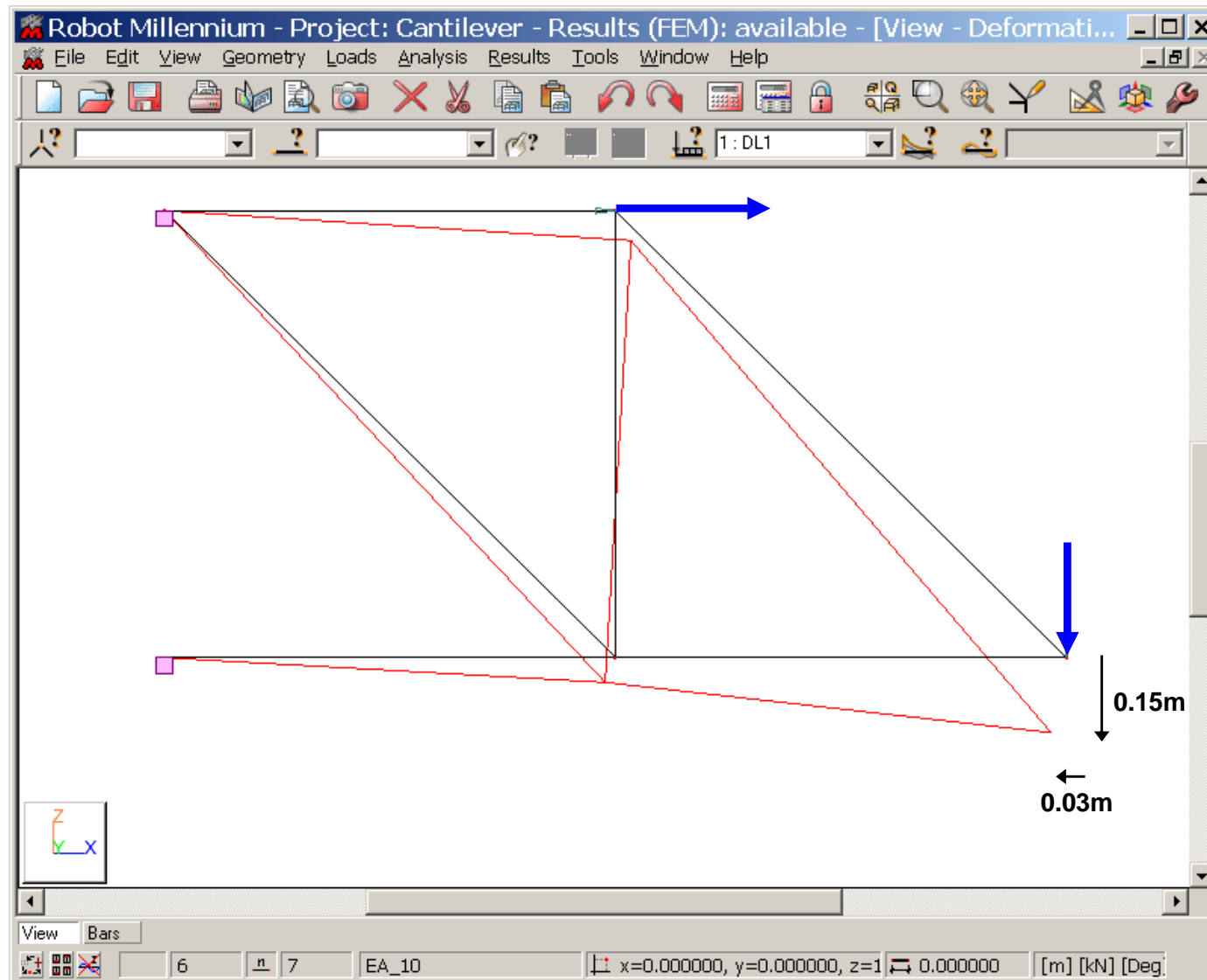


# Reading Input Files



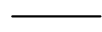
# Chris's Program - ZIP

2Dpinned.dev

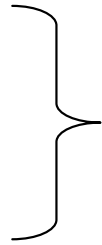
2Dpinned.cpp

MatrixInvert.cpp

size.h



Tells DevC++ which files to load



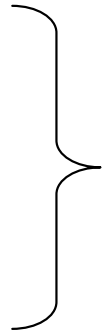
The files to load

nodes.txt

members.txt

loads.txt

supports.txt



Input files for the program

structure.dxf

MemberTensions.txt



Program creates these output files

# Paul's Program - Moodle

2Dpinned.cpp

nodes.txt

members.txt

loads.txt

supports.txt



Input files for the program

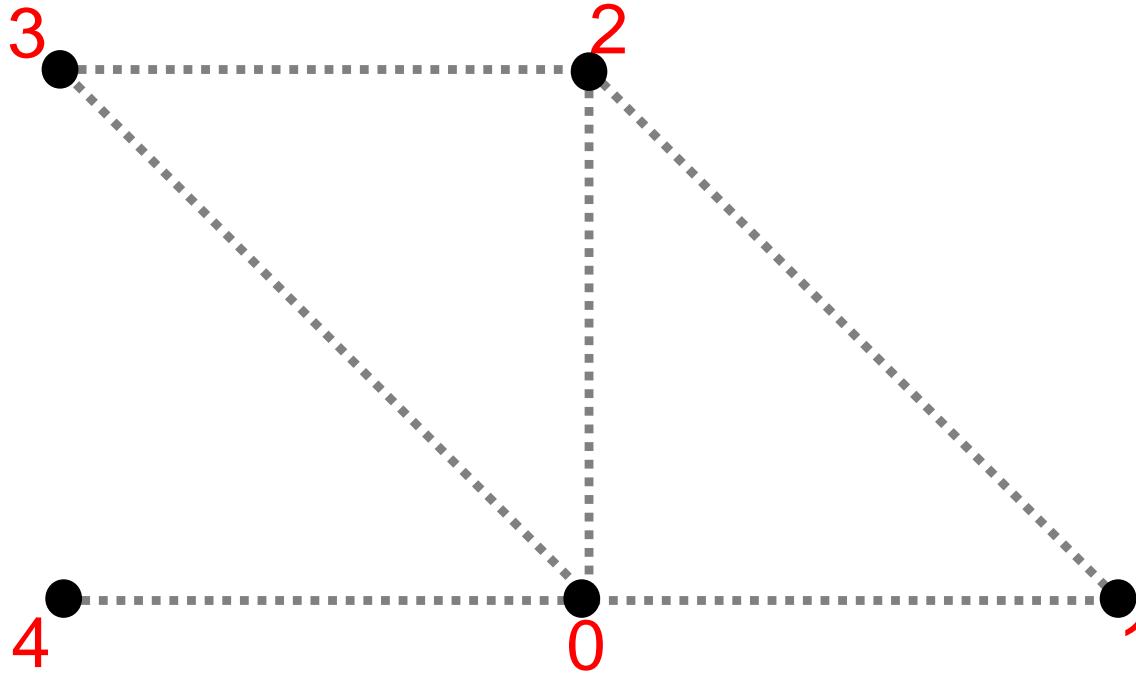
structure.dxf

MemberForces.txt



Program creates these output files

# Nodes



# Nodes - Text File Input

5

Number of Nodes

Chris uses "Last Node Number" here (i.e. = 4)

0 1.0 0.0

Node Number

X-Coordinate

Y-Coordinate

1 2.0 0.0

2 1.0 1.0

3 0.0 1.0

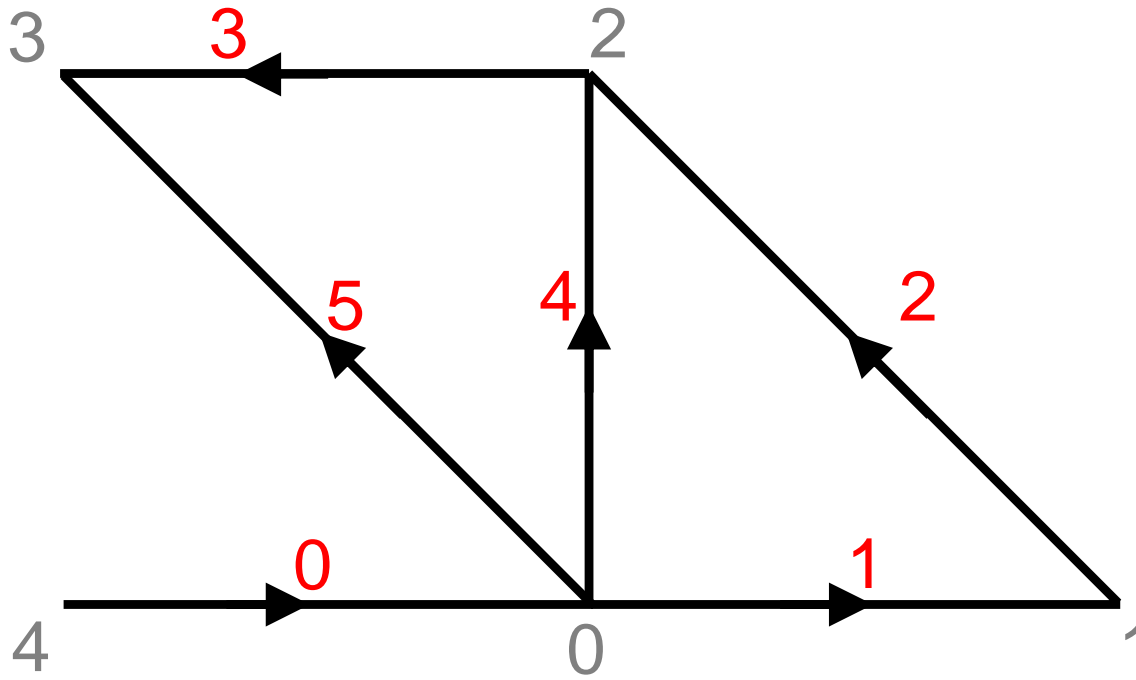
4 0.0 0.0



# Nodes - Internal Storage

double	<b>XCoord[n]</b>	X-Coordinate of n <sup>th</sup> Node
double	<b>YCoord[n]</b>	Y-Coordinate of nth Node
double	<b>Delta[i]</b>	Displacement of i <sup>th</sup> Degree Of Freedom

# Members



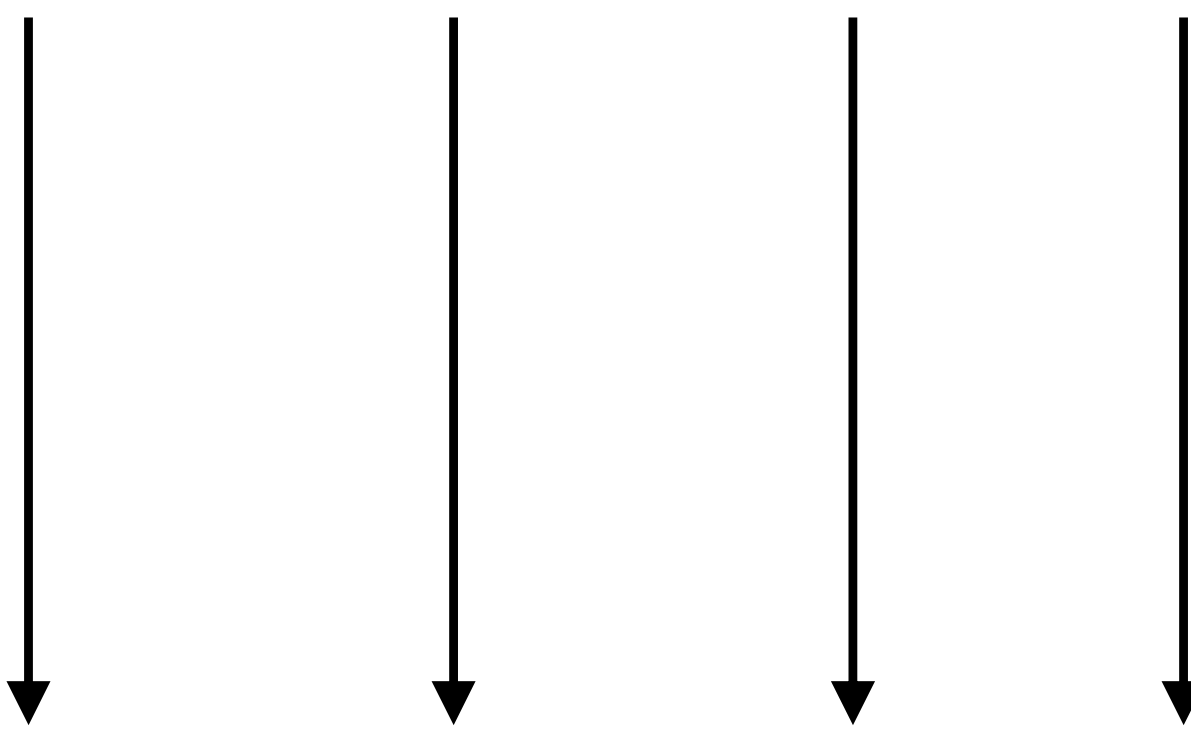
# Members - Text File Input

6

Number of Members

Chris uses "Last Member Number" here (i.e. = 5)

	Member N°	Start Node	End Node	EA
0	4	0	10	
1	0	1	10	
2	1	2	10	
3	2	3	10	
4	0	2	10	
5	0	3	10	

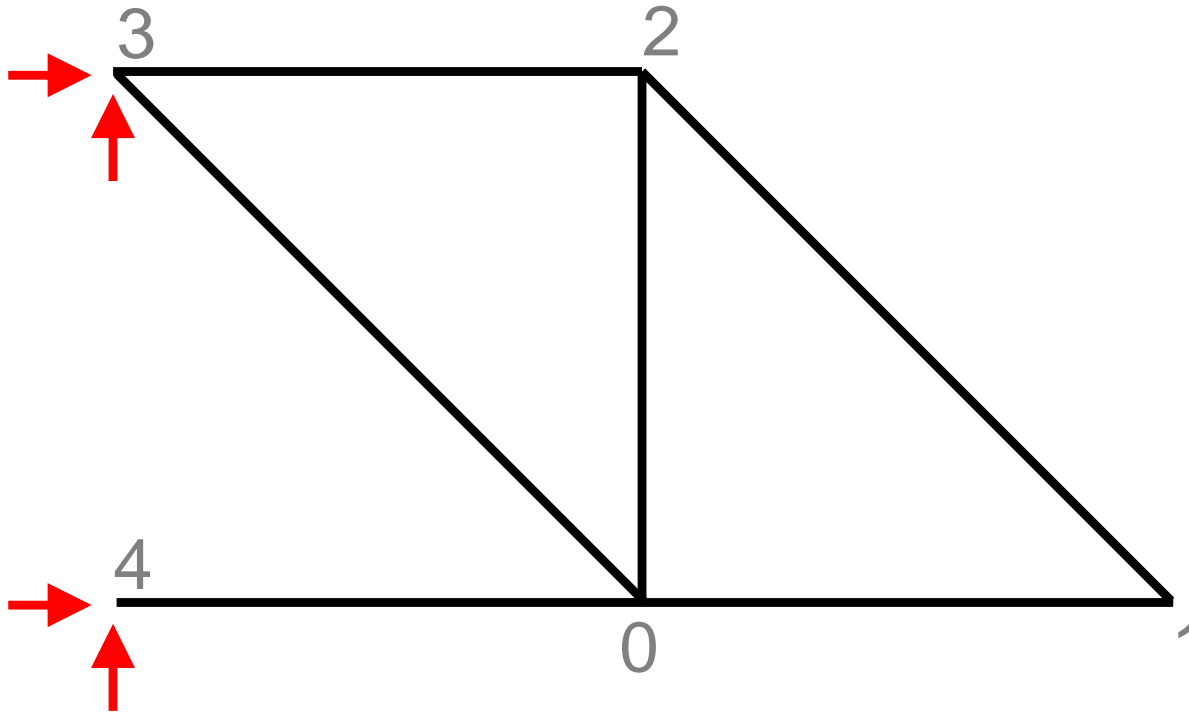




# Members - Internal Storage

int	<b>Start[m]</b>	$m^{\text{th}}$ Member's Start Node
int	<b>End[m]</b>	$m^{\text{th}}$ Member's End Node
double	<b>EA[m]</b>	EA of $m^{\text{th}}$ Member
double	<b>XLength[m]</b>	Original Length Projected onto X-Axis of $m^{\text{th}}$ Member
double	<b>YLength[m]</b>	Original Length Projected onto Y-Axis of $m^{\text{th}}$ Member
double	<b>Length[m]</b>	Original Straight-Line Length of $m^{\text{th}}$ Member

# Supports



# Supports - Text File Input

3 0

Node Number

Direction (0=X, 1=Y)

3 1

4 0

4 1

-1

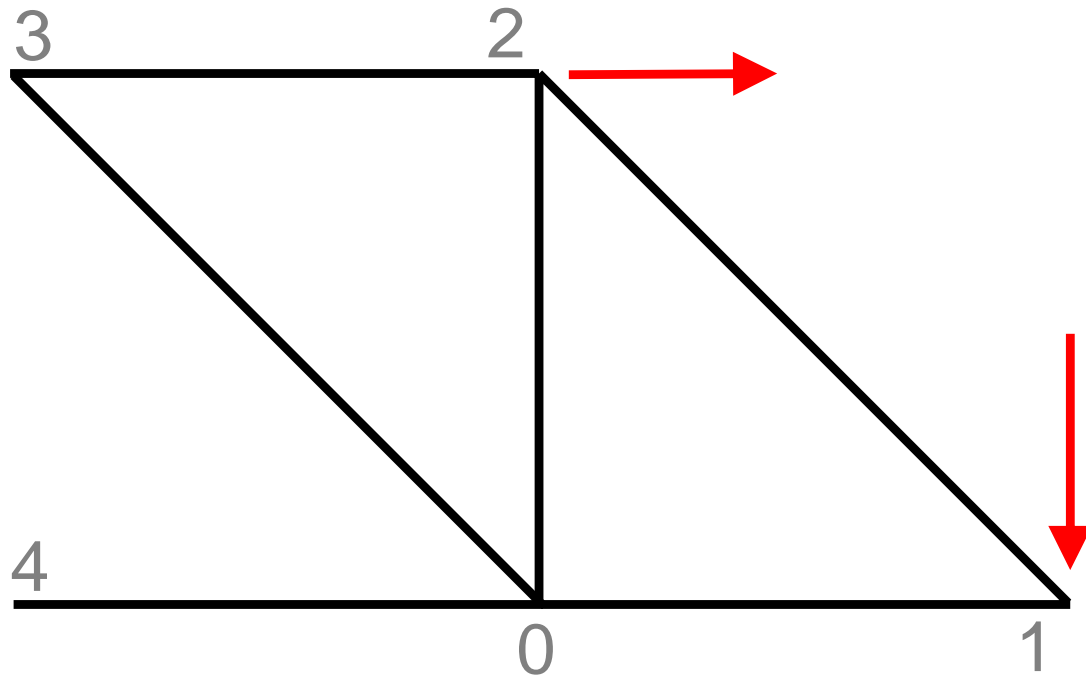


Any node number less-than 0 means stop

# Supports - Internal Storage

$$\begin{bmatrix}
 k_{x_1x_1} & \mathbf{0} & k_{x_1x_2} & k_{x_1y_2} & \cdots & k_{x_1x_n} & k_{x_1y_n} \\
 \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\
 k_{x_2x_1} & \mathbf{0} & k_{x_2x_2} & k_{x_2y_2} & \cdots & k_{x_2x_n} & k_{x_2y_n} \\
 k_{y_2x_1} & \mathbf{0} & k_{y_2x_2} & k_{y_2y_2} & \cdots & k_{y_2x_n} & k_{y_2y_n} \\
 \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
 k_{x_nx_1} & \mathbf{0} & k_{x_nx_2} & k_{x_ny_2} & \cdots & k_{x_nx_n} & k_{x_ny_n} \\
 k_{y_nx_1} & \mathbf{0} & k_{y_nx_2} & k_{y_ny_2} & \cdots & k_{y_nx_n} & k_{y_ny_n}
 \end{bmatrix}
 \begin{bmatrix}
 x_1 \\
 y_1 \\
 x_2 \\
 y_2 \\
 \vdots \\
 x_n \\
 y_n
 \end{bmatrix}
 =
 \begin{bmatrix}
 f_{x_1} \\
 \mathbf{0} \\
 f_{x_2} \\
 f_{y_2} \\
 \vdots \\
 f_{x_n} \\
 f_{y_n}
 \end{bmatrix}$$

# Loads



# Loads - Text File Input

1	1	-0.1	Node Number	Direction (0=X, 1=Y)	Load
2	0	0.2	↓	↓	↓

-1

Any node number less-than 0 means stop

# Loads - Internal Storage

double

**Load[i]**

Load Applied to  $i^{\text{th}}$  Degree Of Freedom

# Summary

- Read data in from a text file
- Store data in arrays (or matrices)
- Calculate arrays we need for later
  - XLength
  - YLength
  - Length