**DESIGN DOCUMENT**

**Node Class Constructor**

**Arguments**

node\_coord: 1x3 vector

node\_number: double

**Computes/Stores**

node\_coord: 1x3 vector

node\_number: double

**Calls**

AssignDOF()

**Returns**

N/A

**GetNodeCoord()**

**Arguments**

N/A

**Computes/Stores**

N/A

**Returns**

node\_coord: 1x3 vector

**Node Class**

**Properties**

node\_coord: 1x3 vector

node\_dof: 6x1 vector

node\_number: double

**Public Methods**

Constructor

GetNodeCoord()

GetNodeDOF()

**Private Methods**

AssignDOF()

**Note:** If the only argument in the function is the Object itself, then the argument has been presented as ‘N/A’ for those functions.

**AssignDOF()**

**Arguments**

node\_number: double

**Computes/Stores**

node\_dof: 6x1 vector

**Returns**

N/A

**GetNodeDOF()**

**Arguments**

N/A

**Computes/Stores**

N/A

**Returns**

node\_dof: 6x1 vector

**Element Class Constructor**

**Arguments**

elementNodes: 1x2 vector

A: double

Izz: double

Iyy: double

J: double

Zzz: double

Zyy: double

E: double

v: double

webdir: 1x3 vector

w: 1x3 vector

**Computes/Stores**

elementNodes: 1x2 vector

A: double

Izz: double

Iyy: double

J: double

Zzz: double

Zyy: double

E: double

v: double

webdir: 1x3 vector

distribLoad: 1x3 vector

length: double

gamma: 12x12 matrix

localStiffness: 12x12 matrix

globalStiffness: 12x12 matrix

element\_dof: 12x1 vector

fixedEndForcesLocal: 12x1 vector

fixedEndForcesGlobal: 12x1 vector

**Calls**

ComputeLength

ComputeTransformationMatrix

ComputeElasticStiffnessMatrix

RetrieveDOF

ComputeFixedEndForces

**Returns**

N/A

**GetFixedEndForcesGlobal()**

**Arguments**

N/A

**Computes/Stores**

N/A

**Returns**

fixedEndForcesGlobal: 12x1 vector

**GetGlobalStiffness ()**

**Arguments**

N/A

**Computes/Stores**

N/A

**Returns**

globalStiffness: 12x12 matrix

**GetElementDOF()**

**Arguments**

N/A

**Computes/Stores**

N/A

**Returns**

element\_dof: 12x1 vector

**ComputeForces()**

**Arguments**

eleDelta: 1x12 vector

**Computes/Stores**

elementForces: 12x1 vector

**Returns**

elementForces: 12x1 vector

**Element Class**

**Properties**

elementNodes: 1x2 vector

A: double

Izz: double

Iyy: double

J: double

Zzz: double

Zyy: double

E: double

v: double

webdir: 1x3 vector

distribLoad: 1x3 vector

length: double

gamma: 12x12 matrix

localStiffness: 12x12 matrix

globalStiffness: 12x12 matrix

element\_dof: 12x1 vector

fixedEndForcesLocal: 12x1 vector

fixedEndForcesGlobal: 12x1 vector

**Public Methods**

Constructor

GetGlobalStiffness

GetElementDOF

GetFixedEndForcesGlobal

ComputeForces

**Private Methods**

ComputeLength

ComputeTransformationMatrix

ComputeElasticStiffnessMatrix

RetrieveDOF

ComputeFixedEndForces

**ComputeLength()**

**Arguments**

N/A

**Computes/Stores**

Local-

firstNode: 1x3 vector

secondNode: 1x3 vector

Global-

length: double

**Calls**

GetNodeCoord

**Returns**

length: double

**ComputeElasticStiffnessMatrix()**

**Arguments**

N/A

**Computes/Stores**

Local-

etay: double

etaz: double

twoy: double

twoz: double

foury: double

fourz: double

sixy: double

sixz: double

twelvey: double

torsion: double

Global-

globalStiffness: 12x12 matrix

localStiffness: 12x12 matrix

**Calls**

N/A

**Returns**

globalStiffness: 12x12 matrix

localStiffness: 12x12 matrix

**ComputeTransformationMatrix()**

**Arguments**

N/A

**Computes/Stores**

Local-

firstNode: 1x3 vector

secondNode: 1x3 vector

xprime: 1x3 vector

zprime: 1x3 vector

Global-

transformationMatrixGamma: 12x12 matrix

(Stored as- gamma: 12x12 matrix

**Calls**

GetNodeCoord

**Returns**

transformationMatrixGamma: 12x12 matrix

**ComputeFixedEndForces()**

**Arguments**

N/A

**Computes/Stores**

fixedEndForcesLocal: 12x1 vector

fixedEndForcesGlobal: 12x1 vector

**Calls**

N/A

**Returns**

fixedEndForcesLocal: 12x1 vector

fixedEndForcesGlobal: 12x1 vector

**RetrieveDOF()**

**Arguments**

N/A

**Computes/Stores**

elementDOF: 12x1 vector

**Calls**

GetNodeDOF

**Returns**

elementDOF: 12x1 vector

**Analysis Class**

**Properties**

nnodes: double

coordinates: nnodesx3 matrix

concen: nnodesx6 matrix

fixity: nnodesx6 matrix

nele: double

ends: nnodesx2 matrix

A: double

Izz: double

Iyy: double

J: double

Zzz: double

Zyy: double

Azz: double

Ayy: double

E: double

v: double

**Public Methods**

Constructor

RunAnalysis

GetMastan2Returns

**Private Methods**

CreateNodes

CreateElements

ComputeStiffnessSubMatrices

ClassifyDOF

ComputeError

CheckKff

ComputeDisplacementReactions

CreateLoadVectors

RecoverElementForces

webdir: 1x3 vector

distribLoad: 1x3 vector

Nodes: nnodesx1 vector

Elements: nelex1 vector

Kff: freeDOFxfreeDOF matrix

Kfn: freeDOFxknownDOF matrix

Knf: knownDOFxfreeDOF matrix

Knn: knownDOFxknownDOF matrix

Ksf: fixedDOFxfreeDOF matrix

Ksn: fixedDOFxknownDOF matrix

FeFf: freeDOFx1 vector

AFLAG: double

DEFL: nnodesx6 matrix

REACT: nnodesx6 matrix

ELE\_FOR: nelex12 matrix

error: double

**Analysis Class Constructor**

**Arguments**

nnodes: double

coordinates: nnodesx3 matrix

concen: nnodesx6 matrix

fixity: nnodesx6 matrix

nele: double

ends: nnodesx2 matrix

A: double

Izz: double

Iyy: double

J: double

Zzz: double

Zyy: double

Azz: double

Ayy: double

E: double

v: double

webdir: 1x3 vector

distribLoad: 1x3 vector

**Computes/Stores**

nnodes: double

coordinates: nnodesx3 matrix

concen: nnodesx6 matrix

fixity: nnodesx6 matrix

nele: double

ends: nnodesx2 matrix

A: double

Izz: double

Iyy: double

J: double

Zzz: double

Zyy: double

Azz: double

Ayy: double

E: double

v: double

webdir: 1x3 vector

distribLoad: 1x3 vector

Nodes: nnodesx1 vector

Elements: nelex1 vector

Kff: freeDOFxfreeDOF matrix

Kfn: freeDOFxknownDOF matrix

Knf: knownDOFxfreeDOF matrix

Knn: knownDOFxknownDOF matrix

Ksf: fixedDOFxfreeDOF matrix

Ksn: fixedDOFxknownDOF matrix

**Calls**

CreateNodes

CreateElements

ComputeStiffnessSubMatrices

**Returns**

N/A

**RunAnalysis()**

**Arguments**

N/A

**Computes/Stores**

AFLAG: double

DEFL: nnodesx6 matrix

REACT: nnodesx6 matrix

ELE\_FOR: nelex12 matrix

error: freeDOFx1 vector

**Calls**

CheckKffMatrix

ComputeDisplacementReactions

RecoverElementForces

ComputeError

**Returns**

AFLAG: double

DEFL: nnodesx6 matrix

REACT: nnodesx6 matrix

ELE\_FOR: nelex12 matrix

error: freeDOFx1 vector

**CreateNodes()**

**Arguments**

N/A

**Computes/Stores**

Nodes: nnodesx1 vector

**Calls**

Node Class Constructor

**Returns**

Nodes: nnodesx1 vector

**GetMastan2Returns()**

**Arguments**

N/A

**Computes/Stores**

N/A

**Returns**

AFLAG: double

DEFL: nnodesx6 matrix

REACT: nnodesx6 matrix

ELE\_FOR: nelex12 matrix

error: double

**RecoverElementForces()**

**Arguments**

N/A

**Computes/Stores**

ELE\_FOR: nelex12 matrix

**Calls**

ComputeForces

**Returns**

ELE\_FOR: nelex12 matrix

**CreateElements()**

**Arguments**

N/A

**Computes/Stores**

Elements: nelex1 vector

**Calls**

Element Class Constructor

**Returns**

Elements: nelex1 vector

**CheckKffMatrix()**

**Arguments**

N/A

**Computes/Stores**

Local-

kappa: double

lostDigits: double

Global-

AFLAG: double

**Calls**

N/A

**Returns**

AFLAG: double

**ComputeError()**

**Arguments**

N/A

**Computes/Stores**

Local-

deltaF: freeDOFx1 vector

deltaN: knownDOFx1 vector

backPf: freeDOFx1 vector

realLoads: freeDOFx1 vector

Global-

error: freeDOFx1 vector

**Calls**

ClassifyDOF

**Returns**

error: freeDOFx1 vector

**ClassifyDOF()**

**Arguments**

N/A

**Computes/Stores**

Local-

fixityTrans: 6xnnodes matrix

Global-

freeDOF: freeDOFx1 vector

fixedDOF: fixedDOFx1 vector

knownDOF: knownDOFx1 vector

**Calls**

N/A

**Returns**

freeDOF: freeDOFx1 vector

fixedDOF: fixedDOFx1 vector

knownDOF: knownDOFx1 vector

**CreateLoadVectors()**

**Arguments**

freeDOF: freeDOFx1 vector

fixedDOF: fixedDOFx1 vector

knownDOF: knownDOFx1 vector

**Computes/Stores**

Local-

concen\_t: 6xnnodes matrix

fixityTrans: 6xnnodes matrix

Global-

Pf: freeDOFx1 vector

Pn: knownDOFx1 vector

Ps: fixedDOFx1 vector

FeFf: freeDOFx1 vector

FeFn: knownDOFx1 vector

FeFs: fixedDOFx1 vector

deltaN: knownDOFx1 vector

**Calls**

GetElementDOF

GetFixedEndForcesGlobal

**Returns**

Pf: freeDOFx1 vector

Pn: knownDOFx1 vector

Ps: fixedDOFx1 vector

FeFf: freeDOFx1 vector

FeFn: knownDOFx1 vector

FeFs: fixedDOFx1 vector

deltaN: knownDOFx1 vector

**ComputeStiffnessSubMatrices()**

**Arguments**

N/A

**Computes/Stores**

Local-

K: nnodes\*6xnnodes\*6 matrix

Global-

Kff: freeDOFxfreeDOF matrix

Kfn: freeDOFxknownDOF matrix

Knf: knownDOFxfreeDOF matrix

Knn: knownDOFxknownDOF matrix

Ksf: fixedDOFxfreeDOF matrix

Ksn: fixedDOFxknownDOF matrix

**Calls**

GetElementDOF

GetGlobalStiffness

ClassifyDOF

**Returns**

Kff: freeDOFxfreeDOF matrix

Kfn: freeDOFxknownDOF matrix

Knf: knownDOFxfreeDOF matrix

Knn: knownDOFxknownDOF matrix

Ksf: fixedDOFxfreeDOF matrix

Ksn: fixedDOFxknownDOF matrix

**ComputeDisplacementReactions()**

**Arguments**

N/A

**Computes/Stores**

Local-

deltaF: freeDOFx1 vector

Rs: fixedDOFx1 vector

Rn: knownDOFx1 vector

Global-

DEFL: nnodesx6 matrix

REACT: nnodesx6 matrix

**Calls**

ClassifyDOF

CreateLoadVectors

**Returns**

DEFL: nnodesx6 matrix

REACT: nnodesx6 matrix

Creating ‘**Analysis**’ Object in ud\_3d1el Class

**ComputeStiffnessSubMatrices()**

**GetElementDOF()**

**GetGlobalStiffness()**

Elements

K

**ClassifyDOF()**

Knf

Knn

Kfn

Kff

Ksf

Ksn

FreeDOF, FixedDOF, KnownDOF

**CreateElements()**

**CreateNodes()**

**Node Class Constructor**

Nodes

**AssignDOF()**

**Element Class Constructor**

Vector of Node Objects

‘Nodes’

**GetNodeCoord()**

**ComputeTransformationMatrix()**

**GetNodeCoord()**

**ComputeLength()**

**RetrieveDOF()**

**ComputeElasticStiffnessMatrix()**

**ComputeFixedEndForces()**

length

fixedEndForcesLocal

fixedEndForcesGlobal

**GetNodeDOF()**

fixedEndForcesLocal

fixedEndForcesGlobal

length

transformationMatrixGamma

‘gamma’

elementDOF

Vector of Element Objects

‘Elements’

**Analysis** Object is now created

Analysis.**RunAnalysis()**

**CheckKffMatrix()**

AFLAG

**If** AFLAG == 1

**If** AFLAG == 0

Elements

DEFL

**ComputeForces()**

**RecoverElementForces()**

**ComputeError ()**

Halt Analysis!

DEFL= [0]nnodesx6

REACT= [0]nnodesx6

ELE\_FOR= [0]nelex12

error: Not Applicable

**ComputeDisplacementReactions()**

FeFf

realPf

FreeDOF

AFLAG = 1

FreeDOF, FixedDOF, KnownDOF

**MASTAN2**

**ud\_3d1el Class**

**MASTAN2**

**ud\_3d1el Class**

REACT

error

realLoads

backPf

Kfn

Kff

deltaF

deltaN

DEFL

knownDOF

DEFL

freeDOF

**ClassifyDOF()**

elementForces

ELE\_FOR

Elements

gamma

localStiffness

fixedEndForcesLocal

Rf=0

DEFL

deltaS=0

deltaN

Rn

Knn

Knf

Rs

Ksn

Ksf

deltaF

Kfn

Kff

FeFf

deltaN

Ps

FeFs

Pn

FeFn

Pf

**ClassifyDOF()**

**CreateLoadVectors()**