





Notación:

-  Módulo principal del mét. numérico (preimplementado, no programable por el usuario).
-  Módulo del método numérico programable por el usuario.
-  Módulo del método numérico preimplementado (no programable por el usuario)
-  Modelo de datos (constantes, esquema numérico, etc.)

`sample`: variable de entrada a un módulo.

`sample`: variable de salida de un módulo.

`dim1×dim2`: dimensiones de una variable (matriz, vector o escalar)

Resumen de variables:

- `xnode`: matriz de pares (x,y) representando cada nodo de la malla.
- `icone`: matriz de conectividad. Todos los nodos se conectan formando elementos rectangulares (pero el tratamiento general del método es por nodos)
- `K`: matriz del sistema.
- `Kd`: matriz del sistema luego de aplicar la condición de contorno Dirichlet.
- `F`: vector de fuerzas.
- `Fd`: vector de fuerzas luego de aplicar la condición de contorno Dirichlet.
- `U`: vector solución del método numérico (vectorial)
- `Fixnodes`: matriz de nodos con valores fijos.
- `Sideload`: matriz de pares de nodos frontera donde se aplican cargas sobre el lado.
- `Pointload`: matriz de nodos donde se aplican fuerzas puntuales.
- `model`: struct con todos los datos del modelo (constantes, esquema numérico, etc.)
- `Def`: matriz de deformaciones (tres valores por cada nodo).
- `Ten`: matriz de tensiones (tres valores por cada nodo).
- `Ten_VM`: tensiones de Von Misses (un valor por cada nodo).
- `reaction`: vector de reacciones (dos valores por cada nodo fijo).

Resumen de dimensiones de variables:

- `nnodes`: cantidad total de nodos de la malla.
- `nelem`: cantidad total de elementos de la malla.
- `nnod`: cantidad de nodos del elemento (3 o 4 nodos).
- `nfixed`: cantidad de nodos frontera con valores fijos.
- `nsload`: cantidad de pares de nodos (lado de un elemento) con cargas sobre el lado.
- `npload`: cantidad de nodos con cargas puntuales.



model

- `nnodes` [1×1 int] - Cantidad de nodos totales de la malla.
- `nelem` [1×1 int] - Cantidad de elementos totales de la malla.
- `young` [1×1 double] - Módulo de Young.
- `poiss` [1×1 double] - Coeficiente de Poisson.
- `gravity` [1×1 double] - Aceleración de la gravedad.
- `pstrs` [1×1 int] - Parámetro de selección de sistema: (1) Tensión plana; cualquier otro valor para deformación plana.
- `thick` [1×1 double] - Espesor de la malla.

```
xnode [nnodes×2 double]
icone [nelem×4 int]
fixnodes [nfixed×1 double]
sideload [nsload×1 double]
pointload [npload×1 double]
model [1×1 struct]
```

fem2d_pstr

U [2*nnodes×1 double]

```
% Inicialización de las variables ppales. del Sistema
1. [K,F] = fem2d_pstr_initialize(model.nnodes);

% Generación de la matriz constitutiva
2. [D] = fem2d_heat_const_mat(model);

% Ensamble de las matrices y vectores del sistema
3. [K,F] = fem2d_pstr_gen_system(K,F,D,xnode,icone,model);

% Ensamble de nodos frontera con carga distribuida
4. [F] = fem2d_pstr_sideload(F,sideload,xnode);

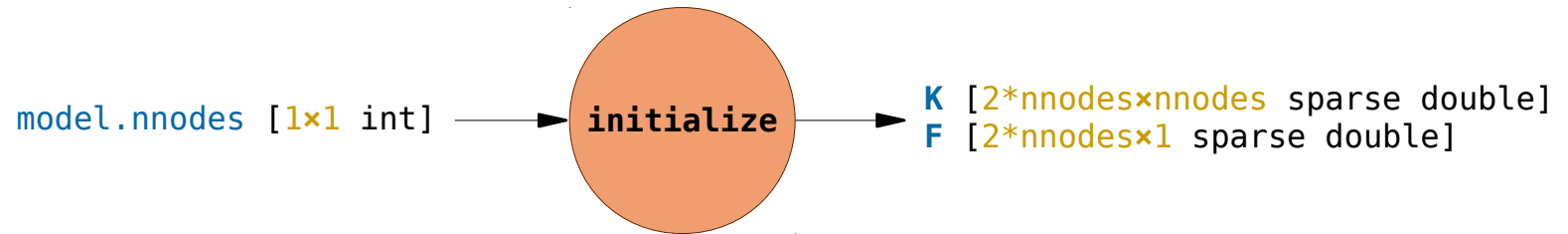
% Ensamble de nodos con fuerzas puntuales
5. [F] = fem2d_pstr_pointload(F,pointload,xnode,icone);

% Ensamble de nodos con condición de desplazamiento
6. [Kd,Fd] = fem2d_pstr_fixnodes(K,F,fixnodes);

% Resolución del sistema lineal de ecuaciones
7. [U] = Kd\Fd;

% Cálculo de deformaciones y tensiones
8. [Def,Ten,Ten_VM] = fem2d_pstr_DT(xnode,icone,model,D,U);

% Cálculo de reacciones
9. [reaction] = fem2d_pstr_reaction(K,F,U,fixnodes);
```



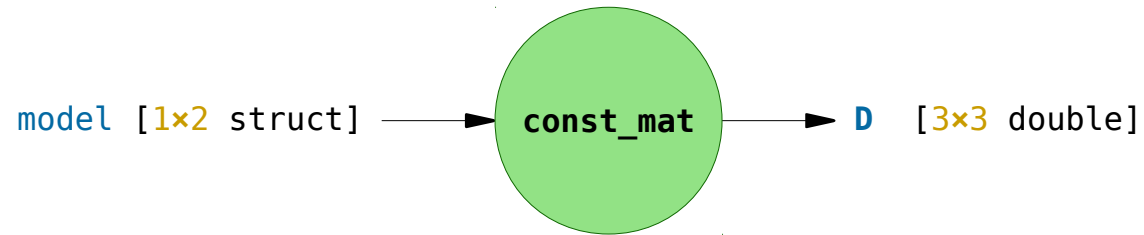
Descripción: módulo para inicializar las variables principales del sistema, las cuales se utilizarán para almacenar los datos calculados y ensamblados por el método numérico. Se inicializan como *sparse* para optimizar el rendimiento general del método numérico.

Entrada:

- `model.nnodes`: cantidad de nodos de la malla.

Salida:

- `K`: matriz del sistema.
- `F`: vector de fuerzas.



Descripción: módulo para calcular la matriz constitutiva del sistema. La misma depende de los valores del Módulo de Young y el Coeficiente de Poisson. La forma de esta matriz es siempre de 3x3, pero la selección de coeficientes determina el tratamiento del sistema como Deformación o Tensión Plana

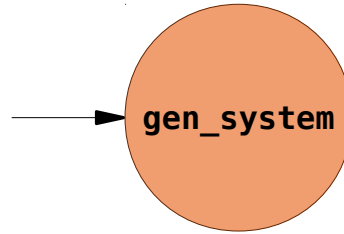
Entrada:

- `model`: struct con todos los datos del modelo (constantes, esquema numérico, etc.)

Salida:

- `D`: matriz constitutiva del sistema.

```
K [2*nnodes*2*nnodes double]
F [2*nnodes*1 double]
D [3*1 double]
xnode [nnodes*2 double]
icone [nelem*4 int]
model [1*1 struct]
```



```
K [2*nnodes*2*nnodes sparse double]
F [2*nnodes*1 sparse double]
```

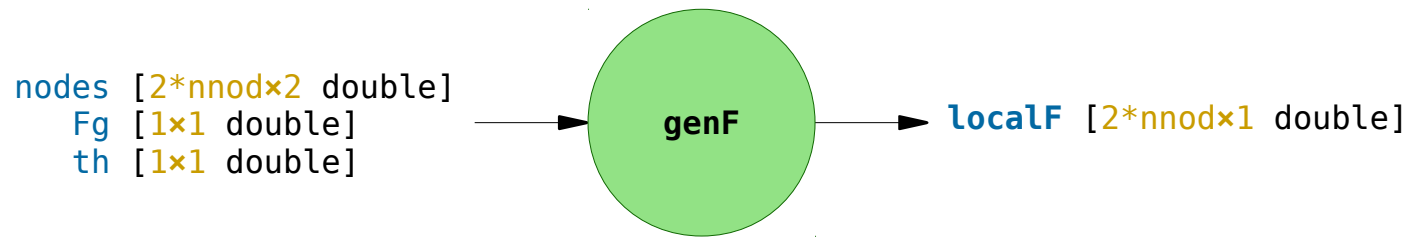
Descripción: módulo para realizar el ensamble de las variables globales del sistema. Es un *wrapper* con la estructura de bucle necesaria para llamar al módulo *assembly* e ir ensamblando las variables elemento a elemento.

Entrada:

- **K**: matriz global del sistema vacía.
- **F**: vector de fuerzas vacío.
- **D**: matriz constitutiva.
- **xnode**: matriz de pares (x,y) representando cada nodo de la malla.
- **icone**: matriz de conectividad. Cada fila de la matriz indica la conectividad de un elemento triangular o rectangular con sus índices enumerados en sentido antihorario. Los elementos triangulares tienen la cuarta columna en -1.
- **model**: struct con todos los datos del modelo (constantes, esquema numérico, etc.)

Salida:

- **K**: matriz global del sistema luego del ensamble.
- **F**: vector de fuerzas luego del ensamble.



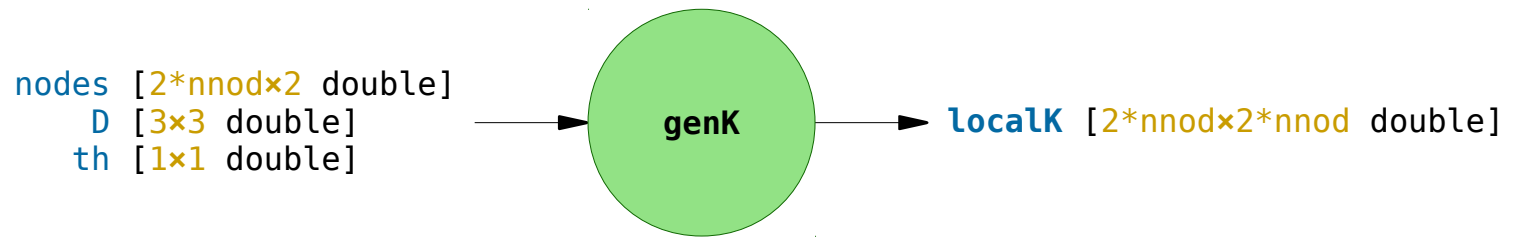
Descripción: módulo para calcular el vector de fuerzas **F** para cada elemento, producto de la presencia de una fuerza por unidad de volumen (generalmente la gravedad) en dicho elemento. La integral se resuelve mediante cuadratura de punto medio, y se requiere evaluar el área del elemento.

Entrada:

- **nodes**: nodos (x,y) del elemento. Los elementos admisibles son de 3 o 4 nodos.
- **Fg**: magnitud de la fuerza gravitatoria.
- **th**: espesor de la placa.

Salida:

- **localF**: vector local de fuerzas.



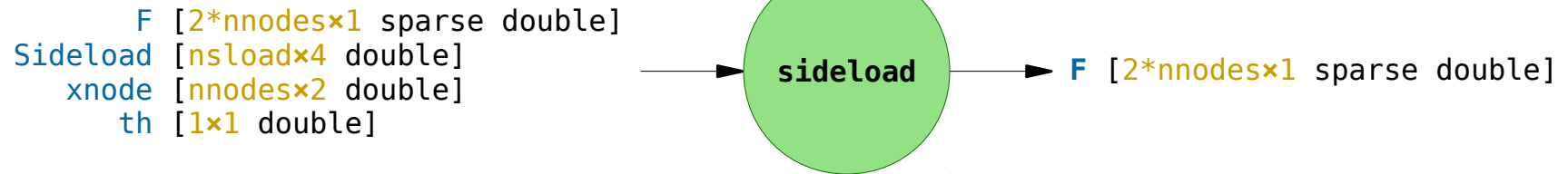
Descripción: módulo para calcular y evaluar de forma numérica la matriz de rigidez K . Se utilizan funciones de forma en coordenadas naturales y se resuelve la integral de forma numérica utilizando cuadratura de Gauss.

Entrada:

- **nodes**: nodos (x,y) del elemento. Los elementos admisibles son de 3 o 4 nodos.
- **D**: matriz constitutiva.
- **th**: espesor de la placa.

Salida:

- **localK**: matriz de rigidez del elemento (local).



Descripción: módulo para calcular y ensamblar las contribuciones de pares de nodos (lados) donde se aplican cargas distribuidas.

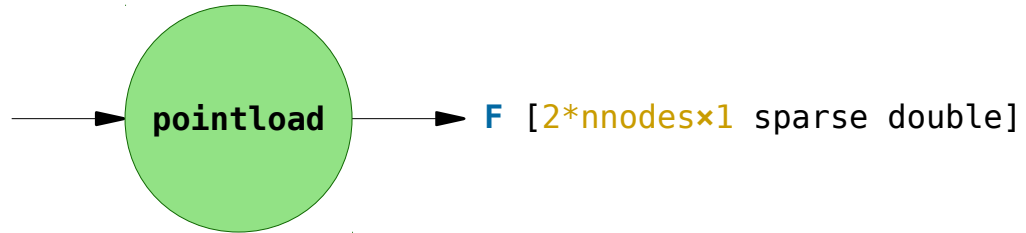
Entrada:

- **F**: vector de fuerzas.
- **Sideload**: matriz con la información sobre fronteras con cargas distribuidas.
 - Columnas 1-2: dos nodos contiguos formando un lado de un elemento.
 - Columna 3: valor de fuerza en sentido eje-x.
 - Columna 4: valor de fuerza en sentido eje-y.
- **xnode**: matriz de nodos con pares (x,y) representando las coordenadas de cada nodo de la malla.
- **th**: espesor de la placa.

Salida:

- **F**: vector de fuerzas. Presenta modificaciones luego de aplicar la condición de borde.

```
F [2*nnodes×1 sparse double]
Pointload [npload×5 double]
xnode [nnodes×2 double]
icone [nelem×4 int]
```



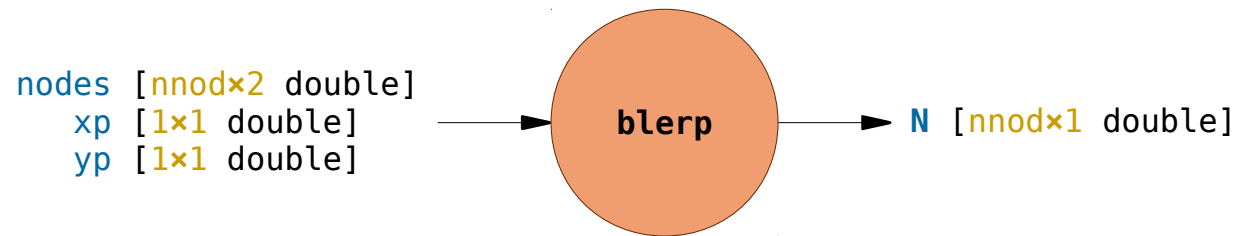
Descripción: módulo para calcular y ensamblar las contribuciones producidas por fuerzas puntuales aplicadas en posiciones (x,y) dentro de elemento(s) del dominio. Estos pares (x,y) también pueden ubicarse en los bordes del elemento, incluso en alguno de sus vértices.

Entrada:

- **F**: vector de fuerzas.
- **xnode**: matriz de nodos con pares (x,y) representando las coordenadas de cada nodo de la malla.
- **icone**: matriz de conectividad. Indica los 3 ó 4 nodos que integran el elemento, recorridos en cualquier orden pero en sentido antihorario. En caso de elementos triangulares, la cuarta columna siempre es -1.
- **Pointload**: matriz con la información sobre las fuentes puntuales aplicadas a elementos del dominio.
 - Columna 1: índice de elemento al que se aplica la fuerza.
 - Columna 2: valor de la fuerza aplicada en sentido eje-x.
 - Columna 3: valor de la fuerza aplicada en sentido eje-y.
 - Columnas 4-5: posición (x,y) dentro del elemento donde se aplica la fuerza.
 -

Salida:

- **F**: vector de fuerzas con modificaciones luego de aplicar la condición de borde.



Descripción: módulo para realizar una interpolación inversa dentro del elemento, a partir de conocer las coordenadas donde se aplica la interpolación. Se calculan las funciones de forma evaluadas en el punto de referencia (**xp**, **yp**). Para el caso de triángulos se aplica la definición de las funciones de forma y el cálculo es directo, mientras que para elementos cuadrangulares se debe realizar una interpolación bilineal inversa para obtener los parámetros (**s**, **t**) asociados al par (**xp**, **yp**).

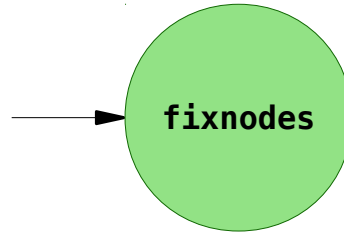
Entrada:

- **nodes**: nodos (x,y) del elemento. Los elementos admisibles son de 3 o 4 nodos.
- **xp**: coordenada-x del punto de referencia.
- **yp**: coordenada-y del punto de referencia.

Salida:

- **N**: funciones de forma para el elemento correspondientes al punto de referencia (**xp**, **yp**).

K [2*nnodes×2*nnodes sparse double]
F [2*nnodes×1 sparse double]
Fixnodes [nfixed×3 double]



K [2*nnodes×2*nnodes sparse double]
F [2*nnodes×1 sparse double]

Descripción: módulo para calcular y ensamblar las contribuciones de nodos pertenecientes a fronteras con desplazamientos fijos.

Entrada:

- **K**: matriz del sistema.
- **F**: vector de fuerzas.
- **Fixnodes**: matriz con la información sobre la frontera de tipo Dirchlet.
 - Columna 1: índice de nodo
 - Columna 2: dirección de desplazamiento:
 - 1 - sentido eje-x.
 - 2 - sentido eje-y.
 - Columna 3: valor del desplazamiento.

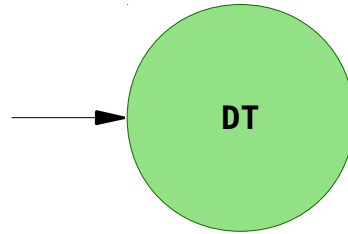
Salida:

- **K**: matriz del sistema luego de realizar las simplificaciones que surgen de aplicar la condición de borde.
- **F**: vector de fuerzs luego de realizar las simplificaciones que surgen de aplicar la condición de borde.

```

xnode [nnodes×2 double]
icone [nelems×4 int]
model [1×1 struct]
      D [3×3 double]
      U [2*nnodes×1 double]

```



```

Def    [nnodes×3 double]
Ten    [nnodes×3 double]
Ten_VM [nnodes×1 double]

```

Descripción: módulo para calcular las deformaciones y tensiones a partir de los desplazamientos calculados **U**. También se calculan las tensiones de Von Misses.

Entrada:

- **xnode**: matriz de nodos con pares (x,y) representando las coordenadas de cada nodo de la malla.
- **icone**: matriz de conectividad. Cada fila de la matriz indica la conectividad de un elemento rectangular, comenzando por el extremo inferior izquierdo y recorriendo el elemento en sentido antihorario.
- **model**: struct con todos los datos del modelo (constantes, esquema numérico, etc.).
- **D**: matriz constitutiva.
- **U**: vector solución (desplazamientos). Cada elemento del vector representa un valor vectorial (dos grados de libertad) asociado a cada nodo de la malla, y su posición dentro del vector depende de cómo se especificó cada nodo en **xnode**.

Salida:

- **Def**: deformaciones. Para cada nodo de la malla se calculan los tres valores típicos de deformaciones.
- **Ten**: tensiones. Para cada nodo de la malla se calculan los tres valores típicos de tensiones.
- **Ten_VM**: tensión de Von Misses. Para cada nodo de la malla se calcula el valor de tensión de Von Misses asociado a dicho nodo.

```
K [2*nnodes*2*nnodes sparse double]
F [2*nnodes*1 sparse double]
U [2*nnodes*1 double]
Fixnodes [nfixed*3 double]
```



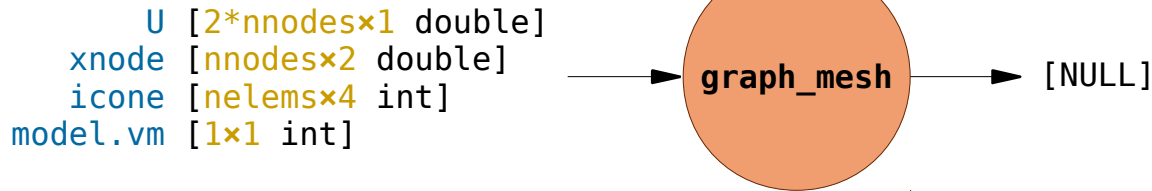
Descripción: módulo para calcular las reacciones presentes en nodos con condición de borde Dirichlet.

Entrada:

- `K`: matriz del sistema.
- `F`: vector de fuerzas.
- `Fixnodes`: matriz con la información sobre la frontera de tipo Dirichlet.
 - Columna 1: índice de nodo
 - Columna 2: valor del desplazamiento en x.
 - Columna 3: valor del desplazamiento en y.
- `U`: vector solución (desplazamientos). Cada elemento del vector representa un valor vectorial (dos grados de libertad) asociado a cada nodo de la malla, y su posición dentro del vector depende de cómo se especificó cada nodo en `xnode`.

Salida:

- `reaction`: vector de reacciones. Por cada nodo se computan las componentes x e y de la reacción en dicho nodo.



Descripción: módulo para graficar la solución del método numérico. Posee distintas formas de operación:

- Activada o desactivada.
- Vista en 2D (plano) y 3D (superficie)
- Visualización de magnitudes vectoriales (desplazamiento en x e y)

Entrada:

- **U**: vector solución. Cada elemento del vector representa un valor vectorial (dos grados de libertad) asociado a cada nodo de la malla, y su posición dentro del vector depende de cómo se especificó cada nodo en **xnode**.
- **xnode**: matriz de nodos con pares (x,y) representando las coordenadas de cada nodo de la malla.
- **icone**: matriz de conectividad. Cada fila de la matriz indica la conectividad de un elemento rectangular, comenzando por el extremo inferior izquierdo y recorriendo el elemento en sentido antihorario.
- **model.vm**: modo de visualización (2D, 3D, ninguna)

Salida: ninguna.