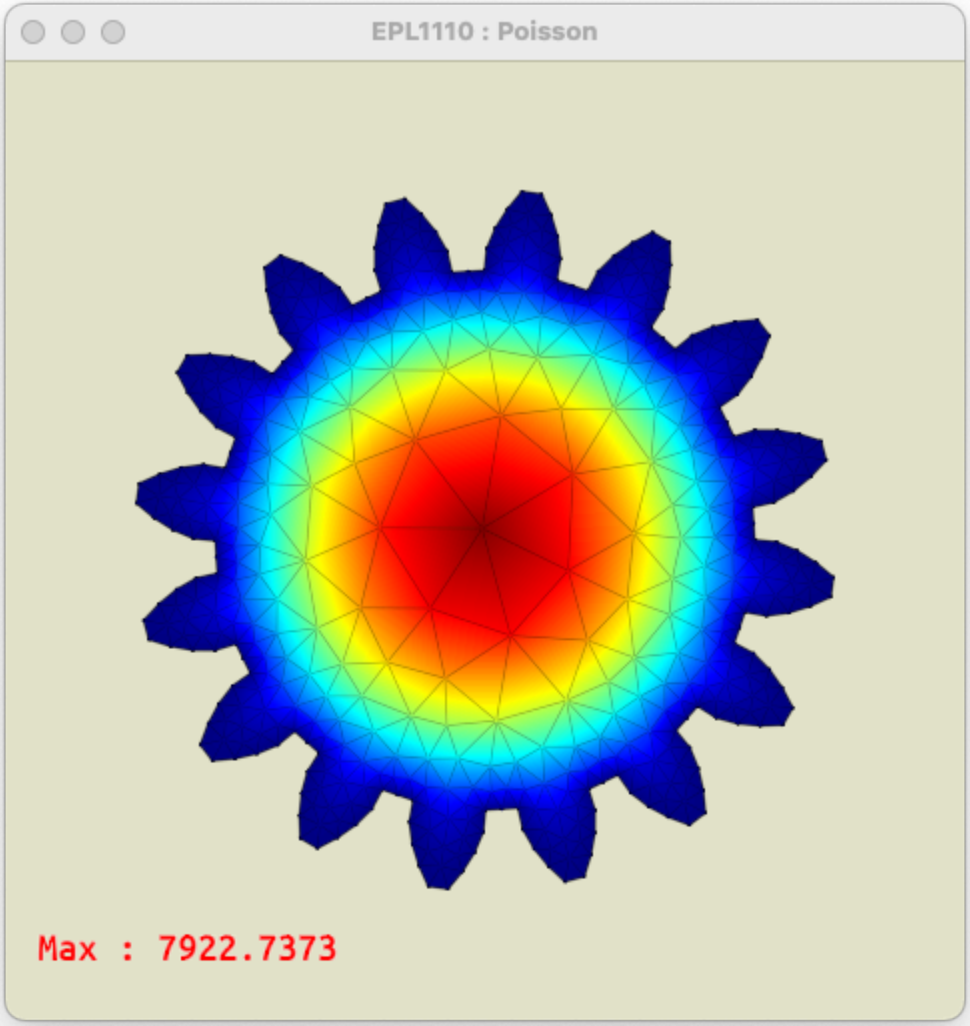


Nous allons faire notre premier
vrai programme d'éléments finis !

$$\langle\,\rangle \triangleq \int_\Omega \qquad \langle\,\rangle_e \triangleq \int_{\Omega_e} \qquad [\;]_e \triangleq [\;]_{X_1^e}^{X_2^e}$$

$$u'' + f = 0$$

$$\langle u'' \tau_i \rangle + \langle f \tau_i \rangle = 0$$

$$\sum_e -\langle u' \tau_i' \rangle_e + [\tau_i u']_e = 0$$

X_i

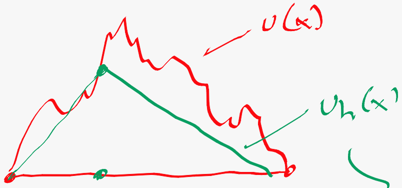
$$\sum_e \underbrace{\langle u \tau_i'' \rangle}_{=0} - \underbrace{[u \tau_i']_e}_{\frac{2}{h} u(X_i)}$$

CAR $\tau_i'' = 0$

VRAI EN 1D

$$\sum_e \langle -(u^*)' \tau_i' \rangle_e = \sum_j -\langle \tau_i' \tau_j' \rangle u(X_j)$$

□



$$U_i = u(X_i)$$

$$\sum U_i \tau_i$$

$$\boxed{\sum_j \langle \tau_i' \tau_j' \rangle u(X_j) = \langle f \tau_i \rangle}$$

FORMULATION DISCRETE □

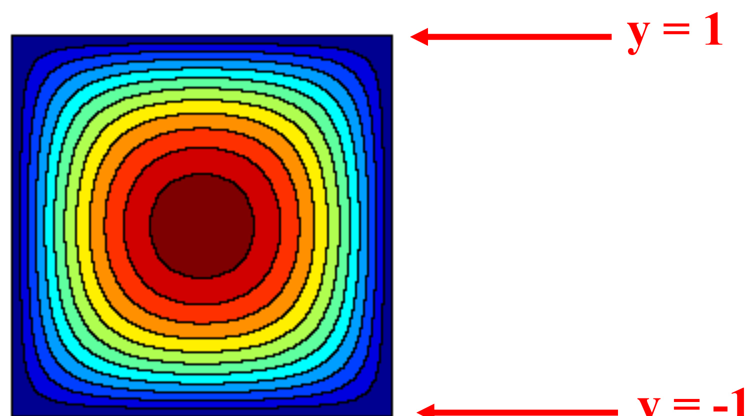
DEFINISSONS

$$u^*(x) = \sum u(X_i) \tau_i$$

L'INTERPOLATION DE LA SOLUTION EXACTE

Un exemple tout simple et bien connu :-)

$$
\begin{aligned}
\nabla^2 u(x,y) + 1 &= 0, \quad (x,y) \in \Omega, \\
u(x,y) &= 0, \quad (x,y) \in \partial\Omega,
\end{aligned}
$$



$$
u(x,y) = \sum_{i,j \text{ odd}} C_{ij} \sin\left(\frac{i\pi(x+1)}{2}\right) \sin\left(\frac{j\pi(y+1)}{2}\right)
$$

Solution analytique

$$
\begin{aligned}
\nabla^2 u(x,y) &= -1, \\
\sum_{i,j \text{ odd}} \pi^2 (i^2+j^2) C_{ij} \sin\left(\frac{i\pi(x+1)}{2}\right) \sin\left(\frac{j\pi(y+1)}{2}\right) &= 1,
\end{aligned}
$$

Because sin are orthogonal,

$$
\frac{\pi^2(i^2+j^2)}{4} C_{ij} = \underbrace{\int_\Omega \sin\left(\frac{i\pi(x+1)}{2}\right) \sin\left(\frac{j\pi(y+1)}{2}\right) d\Omega}_{16/(ij\pi^2)},
$$

$$
\boxed{u(x,y) = \sum_{i,j \text{ odd}} C_{ij} \sin\left(\frac{i\pi(x+1)}{2}\right) \sin\left(\frac{j\pi(y+1)}{2}\right)}
$$

On a ici une solution analytique !
Mais, ce n'est pas le cas dans la plupart des vrais problèmes.
Il s'agit juste d'un exemple pour présenter notre méthode !

Construction du système linéaire discret

$$J(u^h) = \frac{1}{2}\int_\Omega (\boldsymbol{\nabla} u^h)\cdot(\boldsymbol{\nabla} u^h)\, d\Omega - \int_\Omega f\, u^h\, d\Omega,$$

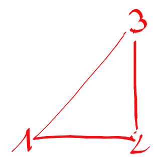
$$J(u^h) = \frac{1}{2}\int_\Omega (\sum_{i=1}^n U_i \boldsymbol{\nabla}\tau_i)\cdot(\sum_{j=1}^n U_j \boldsymbol{\nabla}\tau_j)\, d\Omega - \int_\Omega f\, (\sum_{i=1}^n U_i \tau_i)\, d\Omega,$$

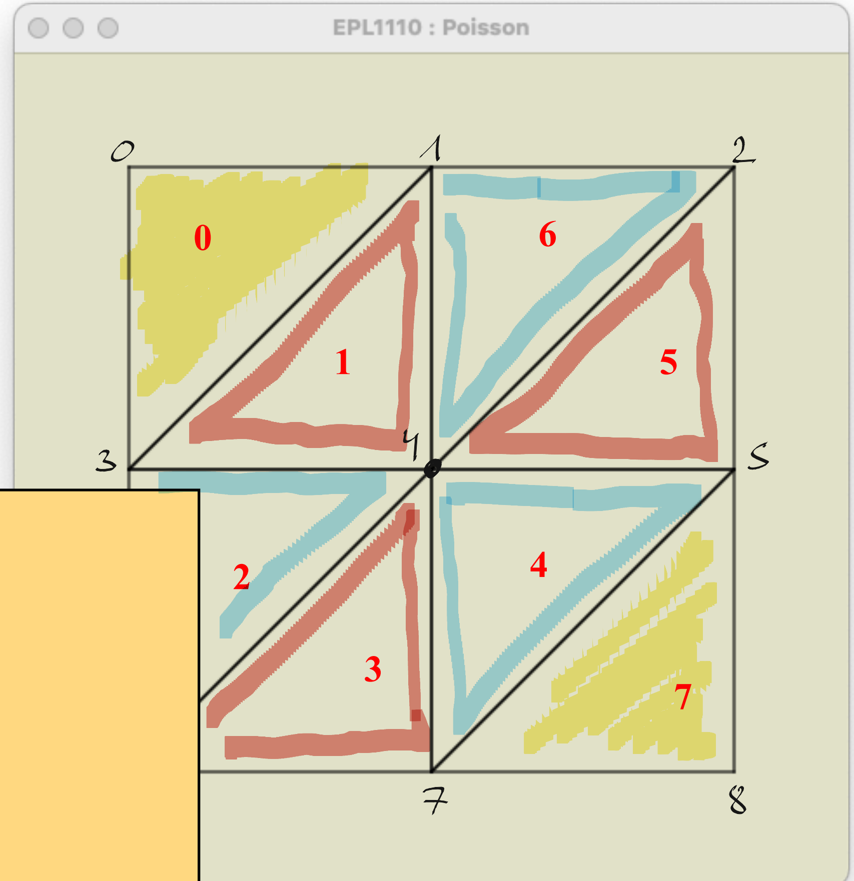
$$J(u^h) = \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n U_i\, U_j \boxed{\int_\Omega (\boldsymbol{\nabla}\tau_i)\cdot(\boldsymbol{\nabla}\tau_j)\, d\Omega} - \sum_{i=1}^n U_i \boxed{\int_\Omega f\, \tau_i\, d\Omega,}$$

$$J(u^h) = \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n U_i\, U_j \boxed{A_{ij}} - \sum_{i=1}^n U_i \boxed{B_i,}$$

$$0 = \frac{\partial J(u^h)}{\partial U_i} = \sum_{j=1}^n \boxed{A_{ij}}\, U_j - \boxed{B_i,} \quad i = 1, n.$$

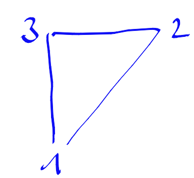
Faisons un petit exemple !



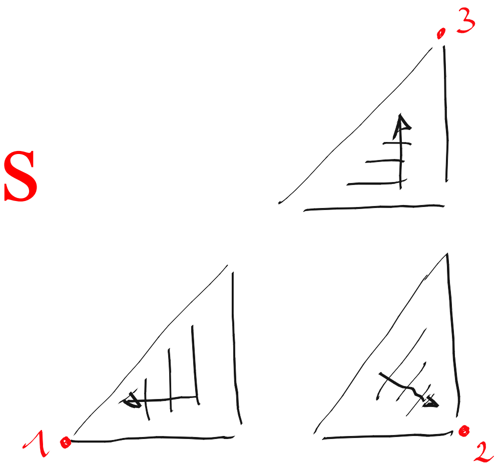


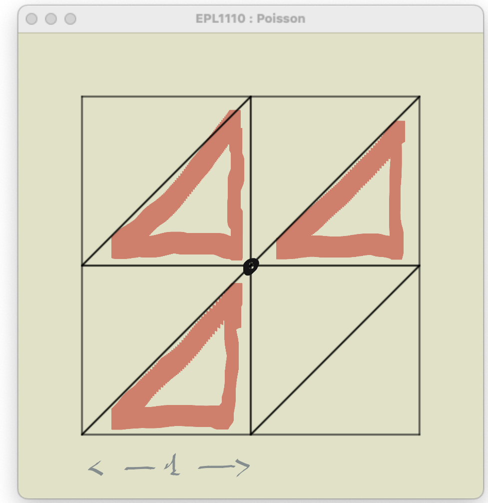

```
Number of nodes 9
     0 :  0.0000000e+00  2.0000000e+00
     1 :  1.0000000e+00  2.0000000e+00
     2 :  2.0000000e+00  2.0000000e+00
     3 :  0.0000000e+00  1.0000000e+00
     4 :  1.0000000e+00  1.0000000e+00
     5 :  2.0000000e+00  1.0000000e+00
     6 :  0.0000000e+00  0.0000000e+00
     7 :  1.0000000e+00  0.0000000e+00
     8 :  2.0000000e+00  0.0000000e+00
Number of triangles 8
     0 :         3      1      0
     1 :         3      4      1
     2 :         6      4      3
     3 :         6      7      4
     4 :         7      5      4
     5 :         4      5      2
     6 :         4      2      1
     7 :         7      8      5
```

$$A_{ij} = \oplus A^e_{ij}$$



Deux matrices locales





$$A^e_{ij} = \int_{\Omega_e} \phi^e_{i,x}\,\phi^e_{j,x} + \phi^e_{i,y}\,\phi^e_{j,y} \; dx\,dy$$

	$\phi_{i,x}$	$\phi_{i,y}$
$i=1$	-1	0
2	1	-1
3	0	1

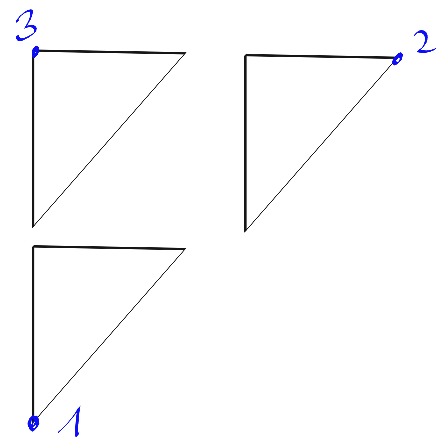
$$\sum \phi_i = 1$$

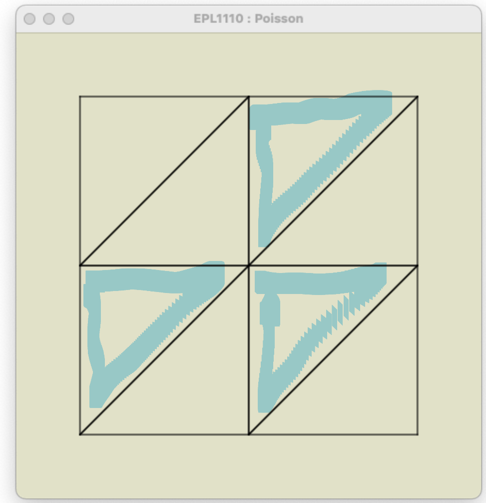
$$\sum \phi_{i,x} = 0$$

$$\sum \phi_{i,y} = 0$$

$$\underbrace{\frac{1}{2}\begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}}_{A^e_{ij}}$$

Deux matrices locales





$$A^e_{i,j} = \int_{\Omega_e} \phi^e_{i,x}\,\phi^e_{j,x} + \phi^e_{i,y}\,\phi^e_{j,y} \quad dx\,dy$$

	$\phi_{i,x}$	$\phi_{i,y}$
$i=1$	0	-1
2	1	0
3	-1	1

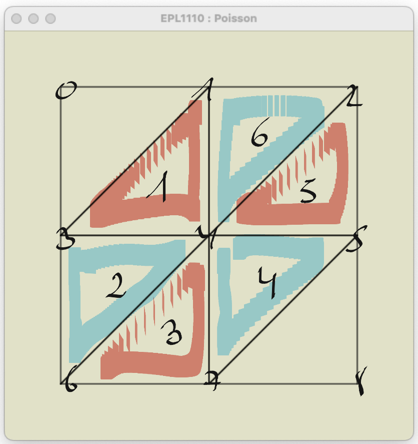
$$\sum \phi_i = 1$$

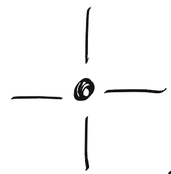
$$\sum \phi_{i,x} = 0$$

$$\sum \phi_{i,y} = 0$$

$$\underbrace{\frac{1}{2}\begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 2 \end{bmatrix}}_{A^e_{i,j}}$$

Assemblons les matrices





$\frac{1}{2}$

	1	2	3	4	5	6	7	8
1	1 + 2	−1		−1 −1				
2	−1	1 + 1			−1			
3			1 + 2	−1 −1		−1		
4	−1 −1		−1 −1	2 +1 +1 2 +1 +1	−1 −1		−1 −1	
5		−1		−1 −1	1 + 2			
6			−1			1 + 1	−1	
7				−1 −1		−1	2 + 1	
8								

$$\underbrace{\frac{1}{2}\begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}}_{A^e_{ij}} \qquad \underbrace{\frac{1}{2}\begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 2 \end{bmatrix}}_{A^e_{ij}}$$

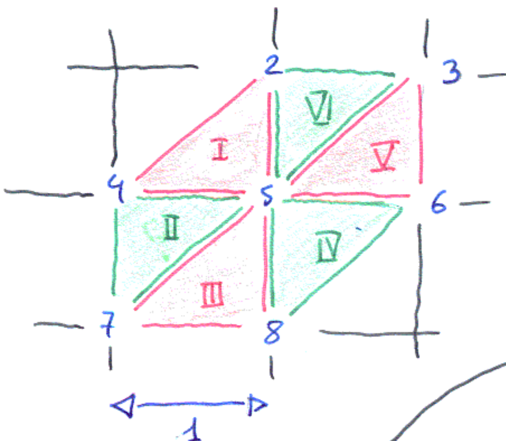
$$-2U_1 - 2U_3 + 8U_4 - 2U_5 - 2U_7$$

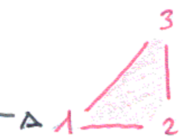
		1	2	3
0	:	3	1	0
1	:	3	4	1
2	:	6	4	3
3	:	6	7	4
4	:	7	5	4
5	:	4	5	2
6	:	4	2	1
7	:	7	8	5

STIFFNESS MATRIX CALCULATION

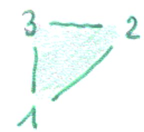
$$\underbrace{< \frac{\partial \phi_i^e}{\partial x}\frac{\partial \phi_j^e}{\partial x} + \frac{\partial \phi_i^e}{\partial y}\frac{\partial \phi_j^e}{\partial y} >_{\Omega_e}}_{A_{ij}^e}$$

$$A_{ij} = \bigoplus A_{ij}^e$$





$$\begin{array}{c|cc} & \phi_{i,x} & \phi_{i,y} \\ 1 & -1 & 0 \\ 2 & 1 & -1 \\ 3 & 0 & 1 \end{array} \longrightarrow \frac{1}{2}\begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$



A_{ij}^e

$$\begin{array}{c|cc} & \phi_{i,x} & \phi_{i,y} \\ 1 & 0 & -1 \\ 2 & 1 & 0 \\ 3 & -1 & 1 \end{array} \longrightarrow \frac{1}{2}\begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 2 \end{bmatrix}$$

CHECK !

$\sum \phi_i = 1$

$\sum \phi_{i,x} = 0$

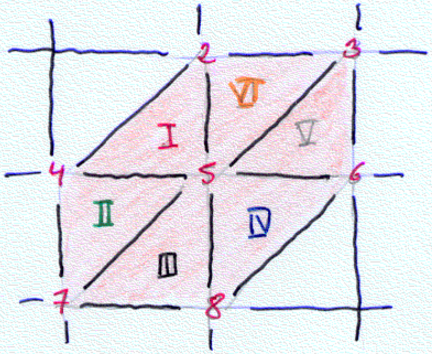
$\sum \phi_{i,y} = 0$

$\phi_1 = (1-x)$ + ct

$\phi_{1,x} = -1$

$\phi_{1,y} = 0$

ASSEMBLING LOCAL MATRICES



$$\begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix} \frac{1}{2}$$

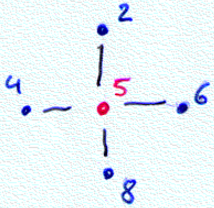
$$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 2 \end{bmatrix} \frac{1}{2}$$

LOCATION TABLE

I	4	5	2
II	7	5	4
III	7	8	5
IV	8	6	5
V	5	6	3
VI	5	3	2

	2	3	4	5	6	7	8
2	1 2	-1		-1 -1			
3	-1	1 1			-1		
4			1 2	-1 -1		-1	
5	-1 -1		-1 -1	2 1 1 2 1 1	-1 -1		-1 -1
6		-1		-1 -1	1 2		
7			-1			1 1	-1
8				-1 -1		-1	2 1

$$\frac{1}{2}\left(8U_5 - 2U_2 - 2U_4 - 2U_8 - 2U_6\right)$$



REGULAR LATTICE OF TURNER TRIANGLES = CENTRAL FINITE DIFFERENCES

Une structure pour un système linéaire Ax = b

```
typedef struct {
    double **A;
    double  *B;
    int size;
} femFullSystem;
```

```
theSystem = femFullSystemCreate(2);
double **A = theSystem->A;
double  *B = theSystem->B;
A[0][0] = 1.0; A[0][1] = 1.0; B[0] = 4;
A[1][0] = 1.0; A[1][1] = 2.0; B[1] = 7;
femFullSystemEliminate(theSystem);
femFullSystemPrint(theSystem);
```

Nous allons faire les tâches suivantes :

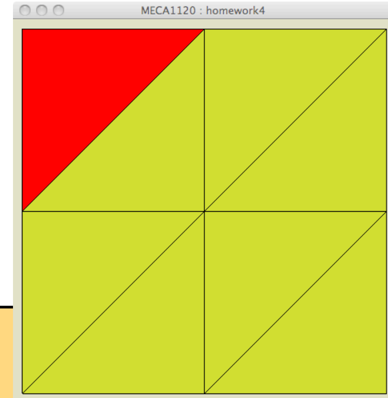
- Allouer la matrice et le vecteur
- Initialiser le système
- Imprimer le système
- Contraindre une valeur
- Résoudre le système

```
+1.0e+00 +1.0e+00 :  +1.0e+00
+1.0e+00 +1.0e+00 :  +3.0e+00
```

Résolution « en place »
par élimination de Gauss
de matrices symétriques
définies positives

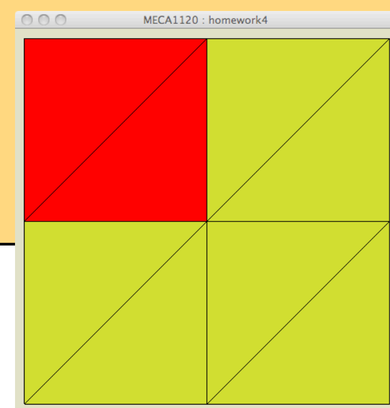
```
femFullSystem* femFullSystemCreate(int size);
void femFullSystemPrint(femFullSystem* mySystem);
void femFullSystemConstrain(femFullSystem* mySystem, int myNode, double myValue);
void femFullSystemEliminate(femFullSystem* mySystem);
```

Assemblage élement…



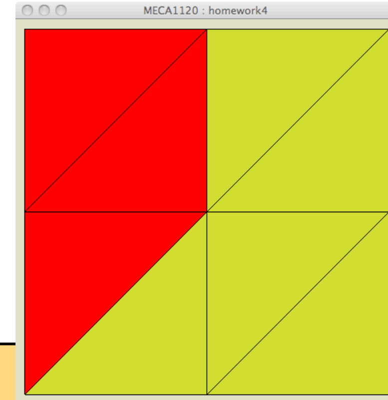

```
+1.0e+00 -5.0e-01           -5.0e-01                                                        :  +1.7e-01
-5.0e-01 +5.0e-01                                                                           :  +1.7e-01
                                                                                            :  +0.0e+00
-5.0e-01                    +5.0e-01                                                        :  +1.7e-01
                                                                                            :  +0.0e+00
                                                                                            :  +0.0e+00
                                                                                            :  +0.0e+00
                                                                                            :  +0.0e+00
                                                                                            :  +0.0e+00
```

```
+1.0e+00 -5.0e-01           -5.0e-01                                                        :  +1.7e-01
-5.0e-01 +1.0e+00                    -5.0e-01                                               :  +3.3e-01
                                                                                            :  +0.0e+00
-5.0e-01                    +1.0e+00 -5.0e-01                                               :  +3.3e-01
         -5.0e-01           -5.0e-01 +1.0e+00                                               :  +1.7e-01
                                                                                            :  +0.0e+00
                                                                                            :  +0.0e+00
                                                                                            :  +0.0e+00
                                                                                            :  +0.0e+00
```

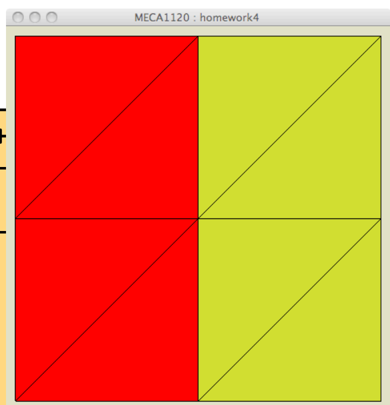


…par élément

Assemblage élément…



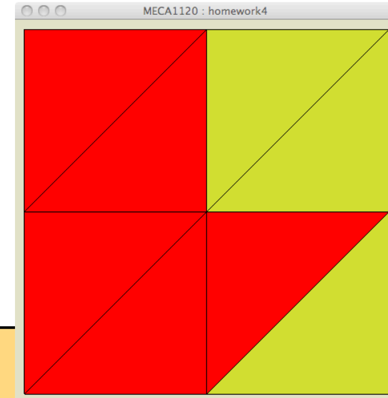
+1.0e+00	-5.0e-01	-5.0e-01						:	+1.7e-01
-5.0e-01	+1.0e+00		-5.0e-01					:	+3.3e-01
								:	+0.0e+00
-5.0e-01		+2.0e+00	-1.0e+00		-5.0e-01			:	+5.0e-01
	-5.0e-01	-1.0e+00	+1.5e+00					:	+3.3e-01
								:	+0.0e+00
		-5.0e-01			+5.0e-01			:	+1.7e-01
								:	+0.0e+00
								:	+0.0e+00



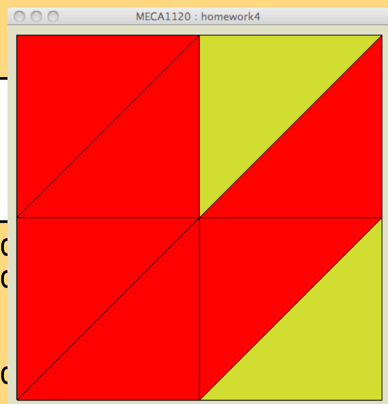
+		-5.0e-01						:	+1.7e-01
-			-5.0e-01					:	+3.3e-01
								:	+0.0e+00
-		+2.0e+00	-1.0e+00		-5.0e-01			:	+5.0e-01
		-1.0e+00	+2.0e+00			-5.0e-01		:	+5.0e-01
								:	+0.0e+00
		-5.0e-01			+1.0e+00	-5.0e-01		:	+3.3e-01
			-5.0e-01		-5.0e-01	+1.0e+00		:	+1.7e-01
								:	+0.0e+00

…par élément

Assemblage élément…




```
+1.0e+00 -5.0e-01          -5.0e-01                                      +1.7e-01
-5.0e-01 +1.0e+00                   -5.0e-01                             +3.3e-01
                                                                       : +0.0e+00
-5.0e-01                   +2.0e+00 -1.0e+00          -5.0e-01         : +5.0e-01
         -5.0e-01          -1.0e+00 +3.0e+00 -5.0e-01          -1.0e+00 : +6.7e-01
                                    -5.0e-01 +5.0e-01                  : +1.7e-01
                           -5.0e-01                   +1.0e+00 -5.0e-01 : +3.3e-01
                                         e+00         -5.0e-01 +1.5e+00 : +3.3e-01
                                                                       : +0.0e+00
```

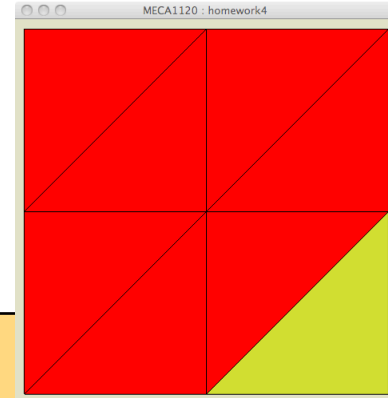



```
+1.0e+00 -5.0e-                                                        : +1.7e-01
-5.0e-01 +1.0e+                          e-01                          : +3.3e-01
                                             -5.0e-01                  : +1.7e-01
-5.0e-01                                 e+00         -5.0e-01         : +5.0e-01
         -5.0e-                          e+00 -1.0e+00          -1.0e+00 : +8.3e-01
                                         e+00 +1.5e+00                  : +3.3e-01
                           -5.0e-01                   +1.0e+00 -5.0e-01 : +3.3e-01
                                    -1.0e+00          -5.0e-01 +1.5e+00 : +3.3e-01
                                                                       : +0.0e+00
```

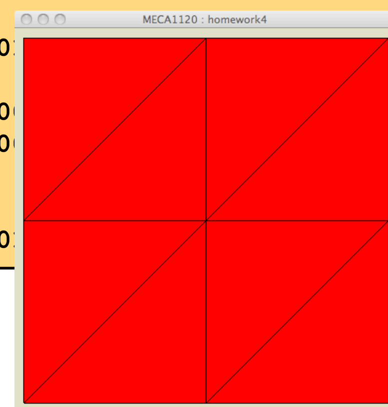
…par élément

Assemblage élément…

```
+1.0e+00 -5.0e-01          -5.0e-01                                                        :  +1.7e-01
-5.0e-01 +2.0e+00 -5.0e-01          -1.0e+00                                               :  +5.0e-01
         -5.0e-01 +1.0e+00                   -5.0e-01                                      :  +3.3e-01
-5.0e-01                   +2.0e+00 -1.0e+00          -5.0e-01                             :  +5.0e-01
         -1.0e+00          -1.0e+00 +4.0e+00 -1.0e+00          -1.0e+00                    :  +1.0e+00
                  -5.0e-01          -1.0e+00 +1.5e+00                                      :  +3.3e-01
                           -5.0e-01                   +1.0e+00 -5.0e-01                    :  +3.3e-01
                                    -1.0e+00          -5.0e-01 +1.5e+00                    :  +3.3e-01
                                                                                           :  +0.0e+00
```




```
+1.0e+00 -5.0e-01          -5.0e-01                                                        :  +1.7e-01
-5.0e-01 +2.0e+00 -5.0e-01          -1.0e+00                                               :  +5.0e-01
         -5.0e-01 +1.0e+00                   -5.0e-0                                       :  +3.3e-01
-5.0e-01                   +2.0e+00 -1.0e+00                                               :  +5.0e-01
         -1.0e+00          -1.0e+00 +4.0e+00 -1.0e+0                                       :  +1.0e+00
                  -5.0e-01          -1.0e+00 +2.0e+0                                  e-01 :  +5.0e-01
                           -5.0e-01                                                        :  +3.3e-01
                                    -1.0e+00                                          e-01 :  +5.0e-01
                                             -5.0e-0                                  e+00 :  +1.7e-01
```



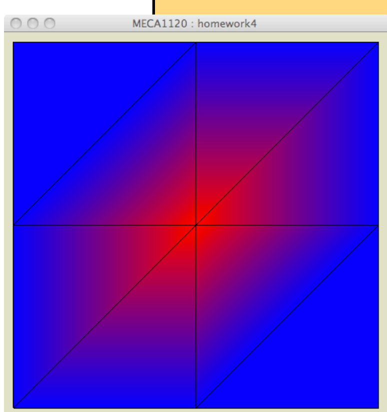
…par élément

Comment imposer les conditions frontières ?

```
def constrain(self,myNode,myValue):
    A = self.A; B = self.B; n = self.n
    for i in range(n):
      B[i] -= myValue * A[i,myNode]
      A[i,myNode] = 0
    for i in range(n):
      A[myNode,i] = 0
    A[myNode,myNode] = 1;
    B[myNode] = myValue;
```

```
for myEdge in theEdges.edges:
    if (myEdge[3] == -1) :
      theSystem.constrain(myEdge[0],0.0)
      theSystem.constrain(myEdge[1],0.0)
```

```
+1.0e+00                                                                                  :  +0.0e+00
          +1.0e+00                                                                        :  +0.0e+00
                    +1.0e+00                                                              :  +0.0e+00
                              +1.0e+00                                                    :  +0.0e+00
                                        +4.0e+00                                          :  +1.0e+00
                                                  +1.0e+00                                :  +0.0e+00
                                                            +1.0e+00                      :  +0.0e+00
                                                                      +1.0e+00            :  +0.0e+00
                                                                                +1.0e+00  :  +0.0e+00
```



Les éléments triangulaires, c'est bien :-)
Mais, ici des éléments quadrilatères, ce serait mieux !

$$\begin{bmatrix} 4 & 5 & 3 \\ 5 & 7 & 8 \\ 3 & 8 & 18 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

OPERATION COLONNE

$$\boxed{v_2 = 4}$$

$$\begin{bmatrix} 4 & 0 & 3 \\ 5 & 0 & 8 \\ 3 & 0 & 18 \end{bmatrix} \qquad \underbrace{\begin{bmatrix} 1 - 5v_2 \\ 2 - 7v_2 \\ 3 - 8v_2 \end{bmatrix}}_{\begin{bmatrix} -19 \\ -25 \\ -29 \end{bmatrix}} \quad v_2 = 4$$

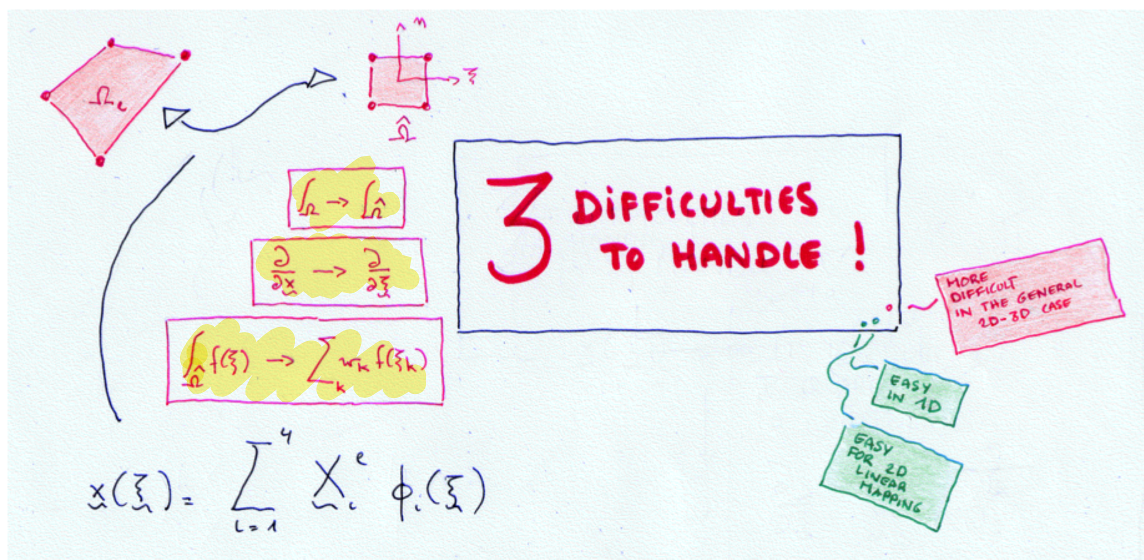
OPERATION LIGNE

$$\begin{bmatrix} 7 & 0 & 3 \\ 0 & 1 & 0 \\ 3 & 0 & 18 \end{bmatrix} \begin{bmatrix} -19 \\ 4 \\ -29 \end{bmatrix}$$

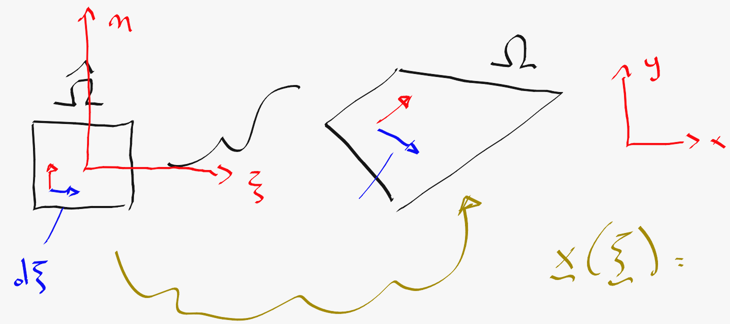
Et on fait comment pour des quadrilatères ?

$$A_{ij} = \int_\Omega (\nabla \tau_i) \cdot (\nabla \tau_j) \, d\Omega,$$

$$B_i = \int_\Omega f \tau_i \, d\Omega.$$



$$\int_{\Omega_e} \longrightarrow \int_{\hat{\Omega}}$$



$$\underline{x}(\underline{\xi}) = \sum_{i=1}^{n} \underline{X}_i^e \, \phi_i(\xi,\eta)$$

$$d\Omega_e = \left| \underbrace{\frac{\partial \underline{x}}{\partial \xi} d\xi} \times \frac{\partial \underline{x}}{\partial \eta} d\eta \right| = \overbrace{\left| \frac{\partial \underline{x}}{\partial \xi} \times \frac{\partial \underline{x}}{\partial \eta} \right|}^{J_e} \underbrace{d\xi \, d\eta}_{d\hat{\Omega}}$$

$$\left(\frac{\partial x}{\partial \xi} d\xi \, ; \, \frac{\partial y}{\partial \xi} d\xi \right)$$

$$\sum \underline{X}_i^e \frac{\partial \phi_i}{\partial \xi}(\xi,\eta)$$

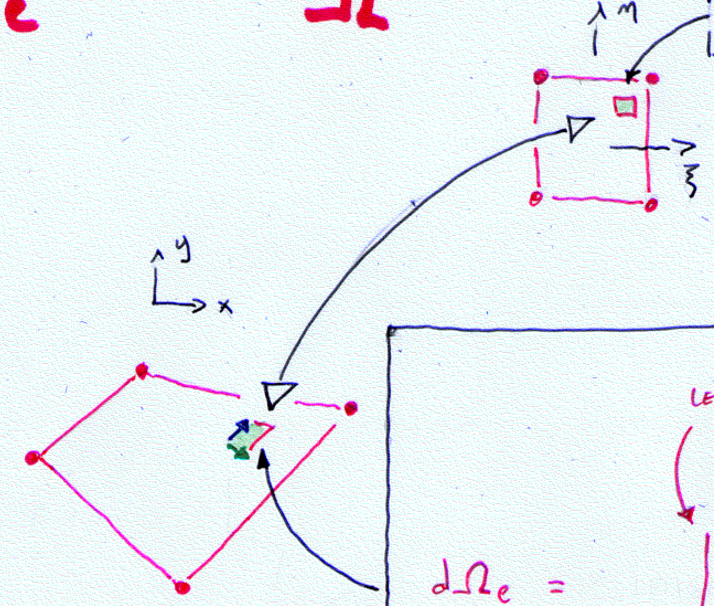
$$\sum \underline{X}_i^e \frac{\partial \phi_i}{\partial \eta}(\xi,\eta)$$

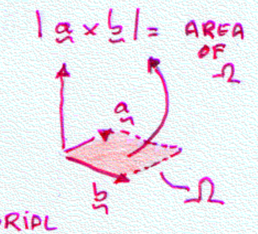
$$\boxed{J_e = \det \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{bmatrix}}$$

$$J_e = \det \left[\frac{\partial \underline{x}}{\partial \underline{\xi}} \right]$$

$$\int_{\Omega_e} \longrightarrow \int_{\hat{\Omega}}$$

$$d\hat{\Omega} = d\xi \, d\eta$$



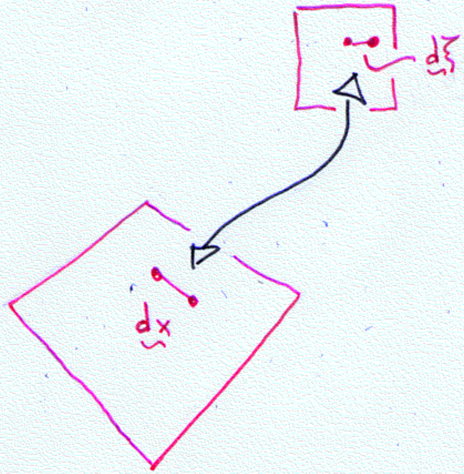


LENGTH

VECTORIAL PRODUCT

$$d\Omega_e = \left| \left(\frac{\partial \underline{x}}{\partial \xi} d\xi \times \frac{\partial \underline{x}}{\partial \eta} d\eta \right) \right|$$

$$= \underbrace{\left| \frac{\partial \underline{x}}{\partial \xi} \times \frac{\partial \underline{x}}{\partial \eta} \right|}_{\det\left(\frac{\partial \underline{x}}{\partial \underline{\xi}}\right)} \underbrace{d\xi \, d\eta}_{d\hat{\Omega}}$$



$$\underset{\sim}{x}(\underset{\sim}{\xi}) = \sum_{i=1}^{4} \underset{\sim}{X}_i^e \, \phi_i(\underset{\sim}{\xi})$$

$$d\underset{\sim}{x}(\underset{\sim}{\xi}) = \underbrace{\sum_{i=1}^{4} \underset{\sim}{X}_i^e \, \frac{\partial \phi_i}{\partial \underset{\sim}{\xi}}(\underset{\sim}{\xi})}_{\frac{\partial \underset{\sim}{x}}{\partial \underset{\sim}{\xi}}} \cdot d\underset{\sim}{\xi}$$

$$\begin{bmatrix} dx \\ dy \end{bmatrix} \qquad \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{bmatrix} \qquad \begin{bmatrix} d\xi \\ d\eta \end{bmatrix}$$

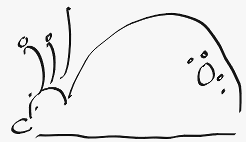
HOW TO CALCULATE J_e ?

$$J^e = \det\left(\frac{\partial \underset{\sim}{x}}{\partial \underset{\sim}{\xi}}\right)$$

$$\frac{\partial}{\partial x} \to \frac{\partial}{\partial \xi}$$

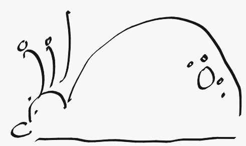
$$\frac{\partial \phi_i}{\partial x} = \frac{\partial \phi_i}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial \phi_i}{\partial \eta}\frac{\partial \eta}{\partial x}$$

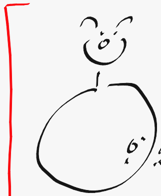
$$\frac{\partial \phi_i}{\partial x} = \frac{\partial \phi_i}{\partial \xi} \cdot \frac{\partial \xi}{\partial x}$$



$$\frac{\partial \phi_i}{\partial \xi} = \frac{\partial \phi_i}{\partial x} \cdot \frac{\partial x}{\partial \xi}$$

$$= \sum_{n_i} x_i^e \frac{\partial \phi_i}{\partial \xi}$$



$=$ $[$  $]^{-1}$

$$\begin{bmatrix} \partial\xi/\partial x & \partial\xi/\partial y \\ \partial\eta/\partial x & \partial\eta/\partial y \end{bmatrix} = \frac{1}{J_e}\begin{bmatrix} \partial y/\partial\eta & -\partial x/\partial\eta \\ -\partial y/\partial\xi & \partial x/\partial\xi \end{bmatrix}$$

$$\frac{\partial}{\partial \underline{x}} \to \frac{\partial}{\partial \underline{\xi}}$$

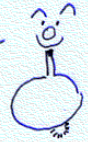
IT IS NOT NECESSARY TO CALCULATE $\underline{\xi}(\underline{x})$!

$\nabla \phi_i$

$$\frac{\partial \phi_i}{\partial \underline{x}} = \frac{\partial \phi_i}{\partial \underline{\xi}} \cdot \frac{\partial \underline{\xi}}{\partial \underline{x}}$$

$$\frac{\partial \phi_i}{\partial \underline{\xi}} = \frac{\partial \phi_i}{\partial \underline{x}} \cdot \frac{\partial \underline{x}}{\partial \underline{\xi}}$$

$$= \sum_{i=1}^{4} \underline{X}_i^e \frac{\partial \phi_i}{\partial \underline{\xi}}$$

 $=$ ( $)^{-1}$

$$\begin{bmatrix} \partial\xi/\partial x & \partial\xi/\partial y \\ \partial\eta/\partial x & \partial\eta/\partial y \end{bmatrix} = \frac{1}{J_e} \begin{bmatrix} \partial y/\partial\eta & -\partial x/\partial\eta \\ -\partial y/\partial\xi & \partial x/\partial\xi \end{bmatrix}$$

$$A_{ij}^{e} = \int_{\hat{\Omega}} J_e \left[\frac{\partial \phi_i}{\partial x} \cdot \frac{\partial \phi_j}{\partial x} + \frac{\partial \phi_i}{\partial y} \cdot \frac{\partial \phi_j}{\partial y} \right]$$

$$\frac{\partial \phi_i}{\partial x} = \underbrace{\frac{1}{J_e} \left[\frac{\partial y}{\partial \eta} \frac{\partial \phi_i}{\partial \xi} - \frac{\partial x}{\partial \xi} \frac{\partial \phi_i}{\partial \eta} \right]}_{f(\xi, \eta)}$$

QUOTIENT
DE POLYNOMES !

$$\begin{bmatrix} \dfrac{\partial \phi_i}{\partial x} & \dfrac{\partial \phi_i}{\partial y} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial \phi_i}{\partial \xi} & \dfrac{\partial \phi_i}{\partial \eta} \end{bmatrix} \cdot \begin{bmatrix} \dfrac{\partial \xi}{\partial x} & \dfrac{\partial \xi}{\partial y} \\[2ex] \dfrac{\partial \eta}{\partial x} & \dfrac{\partial \eta}{\partial y} \end{bmatrix}$$

$$\sum Y_i \frac{\partial \phi_i}{\partial \eta} \qquad\qquad \sum Y_i \frac{\partial \phi_i}{\partial \xi}$$

$$\frac{\partial \phi_i}{\partial x} = \underbrace{\frac{1}{J_e} \left(\frac{\partial y}{\partial \eta} \frac{\partial \phi_i}{\partial \xi} - \frac{\partial y}{\partial \xi} \frac{\partial \phi_i}{\partial \eta} \right)}_{\text{FCT}(\xi,\eta)}$$

$$\boxed{A^e_{ij} = \int_{\hat\Omega} \underbrace{\underbrace{\nabla \phi_i \cdot \nabla \phi_j}_{\text{FCT}(\xi,\eta)} \; J_e(\xi,\eta)}_{\text{FCT}(\xi,\eta)} \, d\hat\Omega}$$

YOU
HAVE JUST
TO INTEGRATE
NOW !

$$\int_{\hat{\Omega}} f(\underline{\xi}) \longrightarrow \sum_{k} \underbrace{w_k \, f(\underline{\xi}_k)}_{\text{TO BE SELECTED IN A CLEVER WAY}}$$

WEIGHTS

$\sum w_k = 1$! IF $\int_{\hat{\Omega}} = 1$

INTEGRATION POINTS

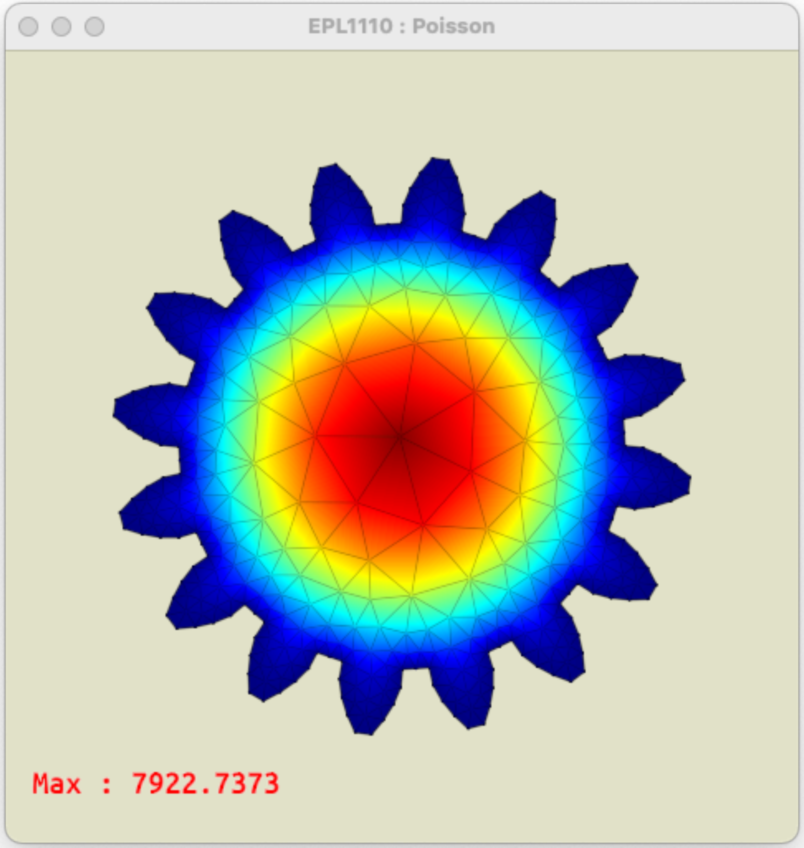
NUMERICAL INTEGRATION REQUIRED !

ANALYTICAL WAY IS IMPOSSIBLE !

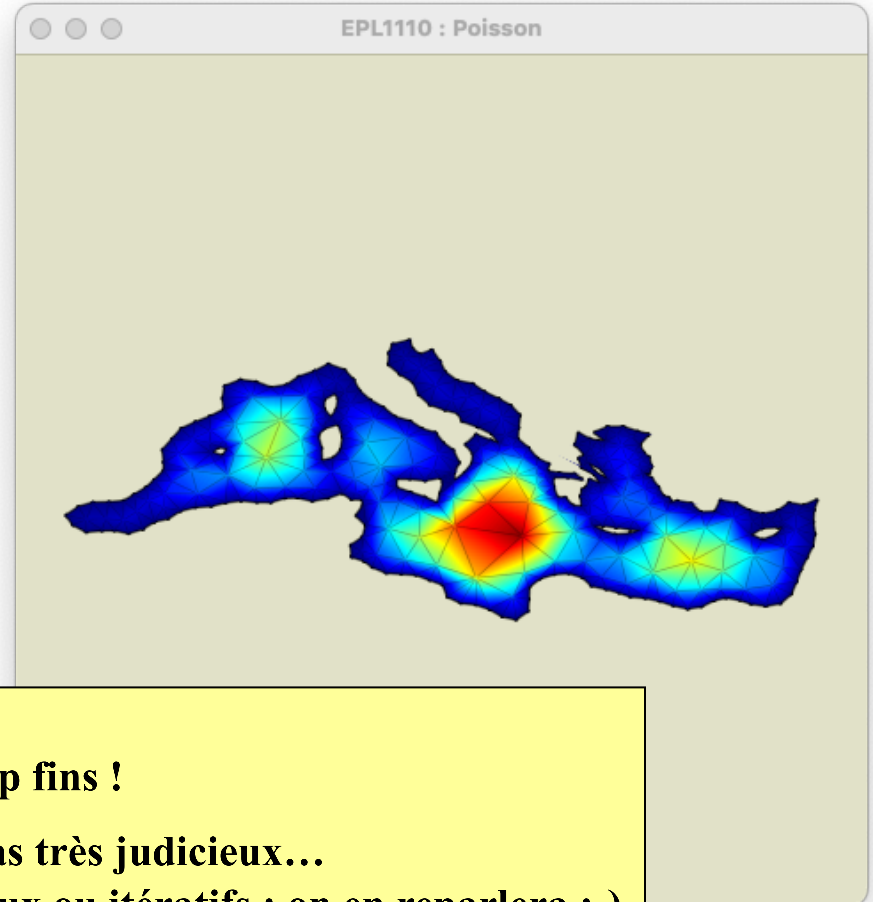
$$\boxed{\underbrace{\underline{\nabla}\phi_i \cdot \underline{\nabla}\phi_j \;\; J_e(\underline{\xi})}_{f(\underline{\xi})}}$$

GAUSS LEGENDRE QUADRATURE !

Et zou
On est prêt !



Calculer la solution du problème de Poisson avec des triangles linéaires ou des quads bilinéaires… Le jacobien n'est pas toujours constant !
Une code unique pour les deux cas !



Le code est très très lent…
Ne pas essayer des maillages trop fins !

Utiliser un solveur plein n’est pas très judicieux…
On fera appel à des solveurs creux ou itératifs : on en reparlera :-)

Un petit Poisson perdu dans la Méditerranée…