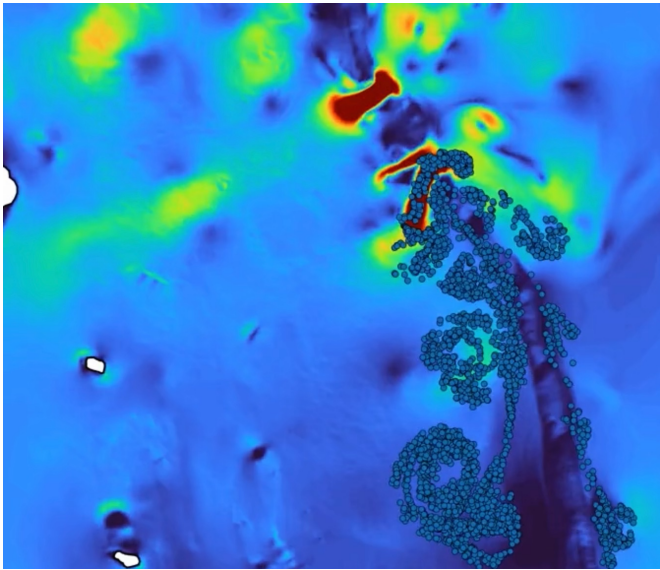
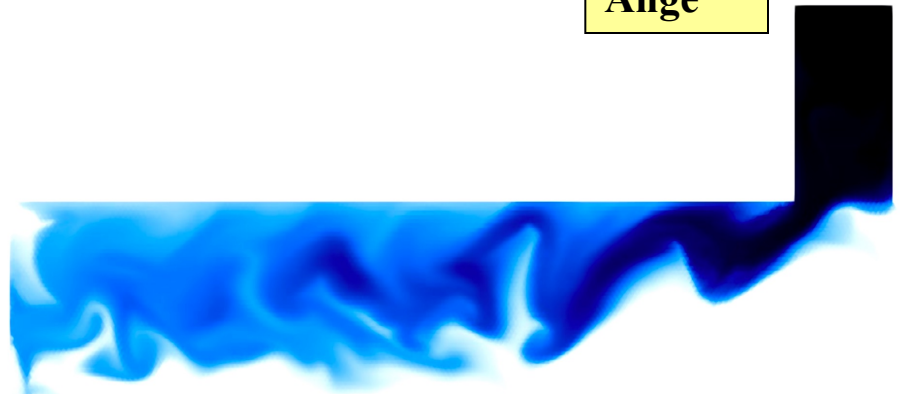


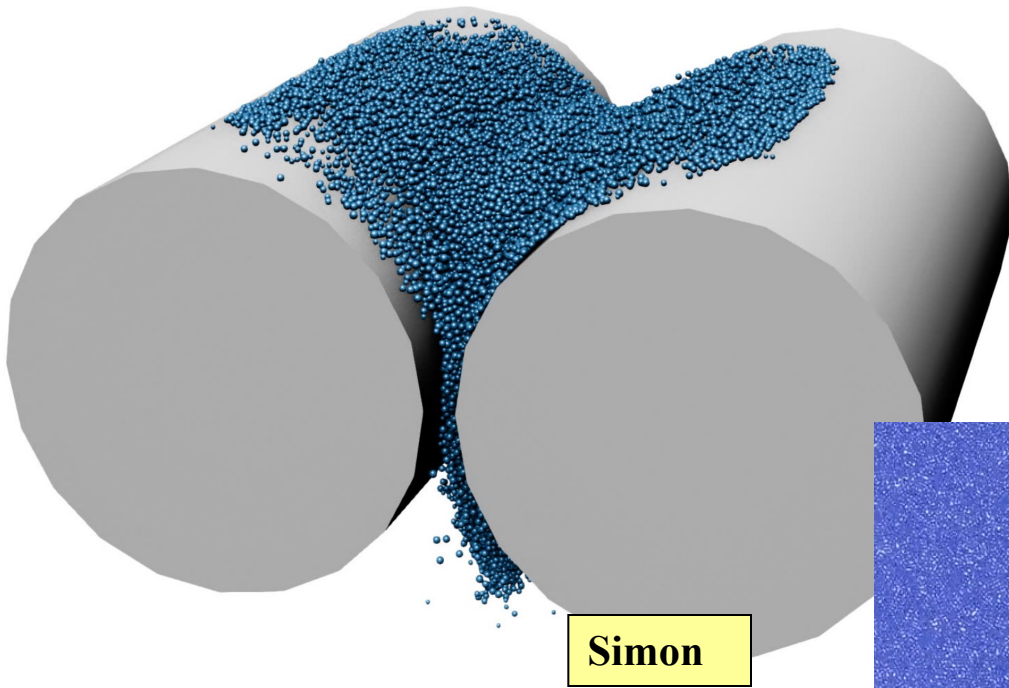


Miguel

Finite
Elements

Ange

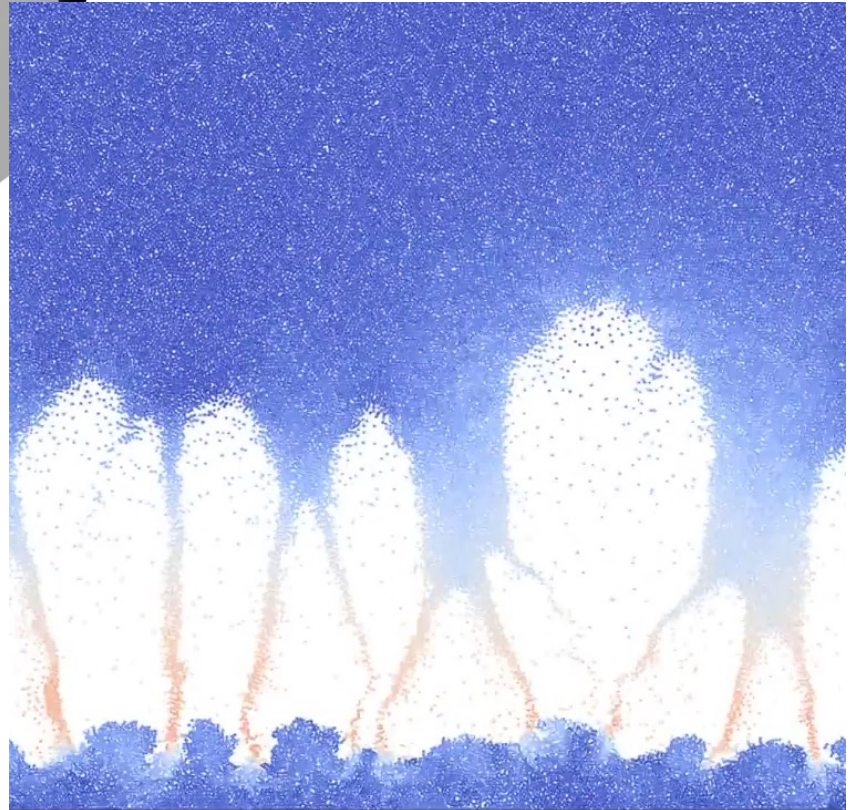


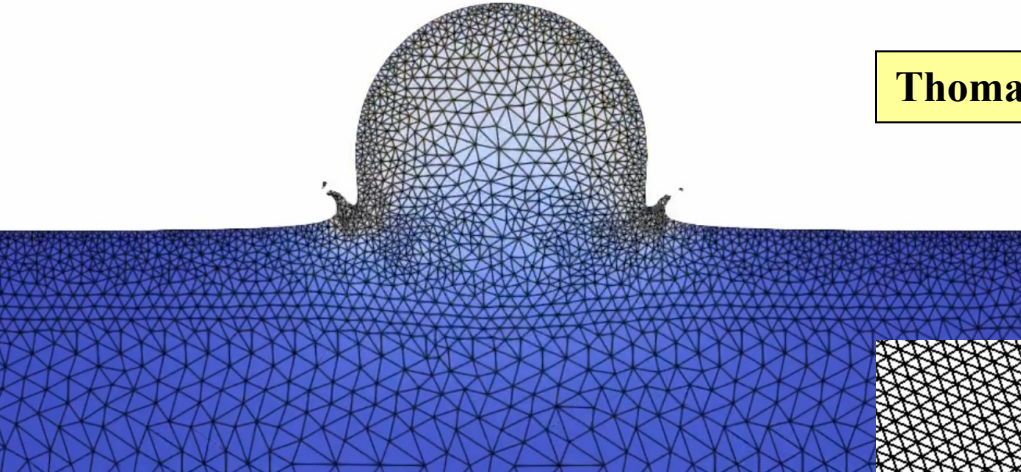


Michel

Simon

Finite
Elements

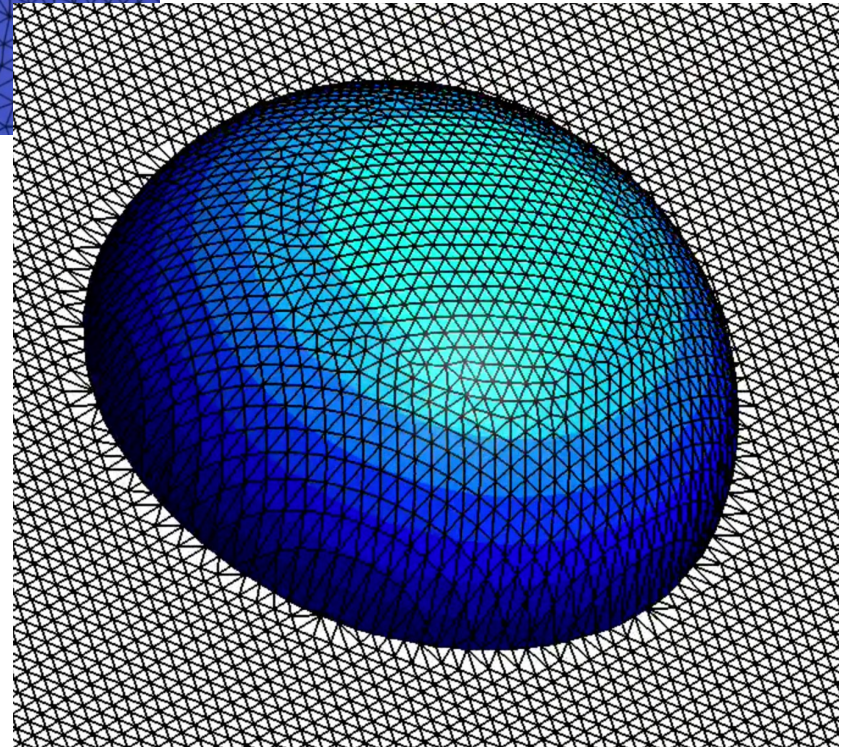


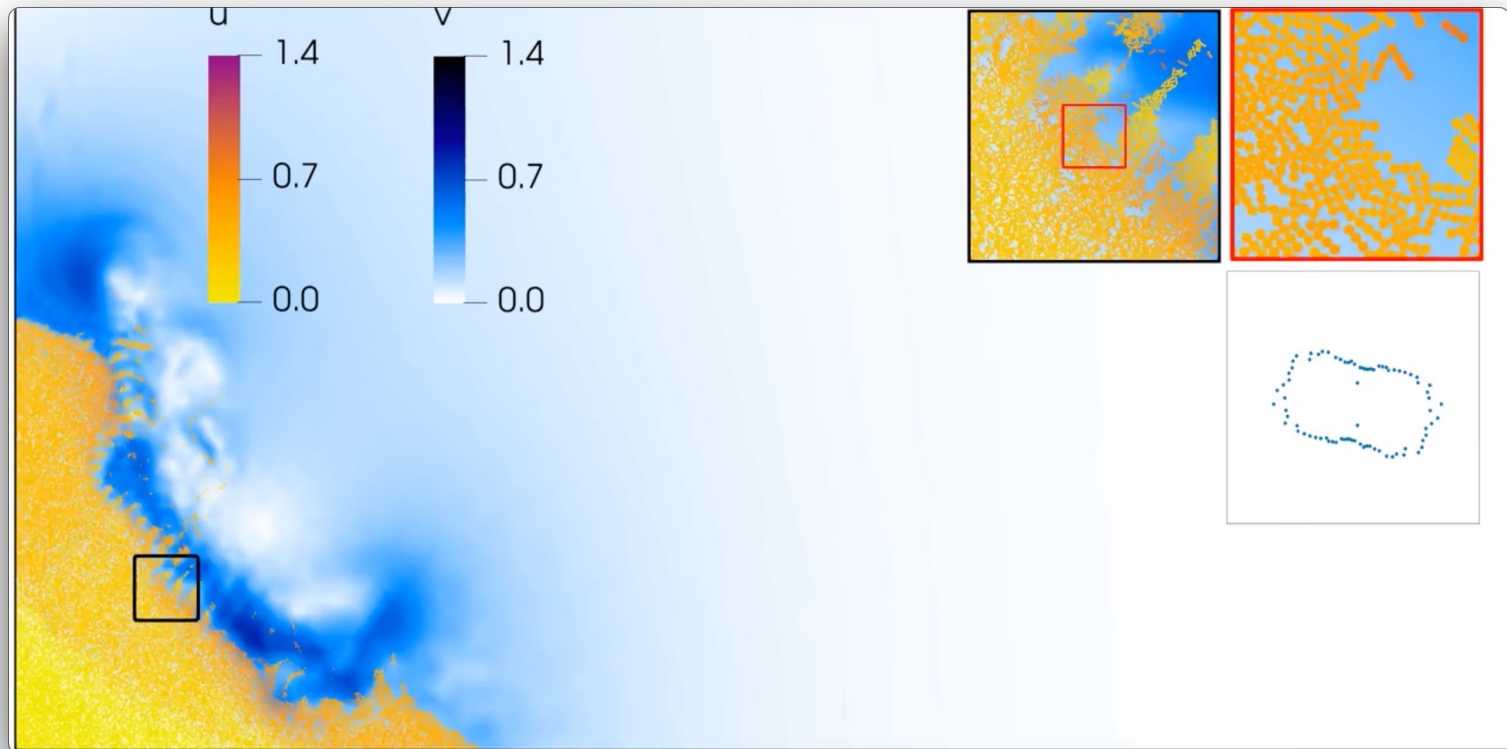


Thomas

Alexandre

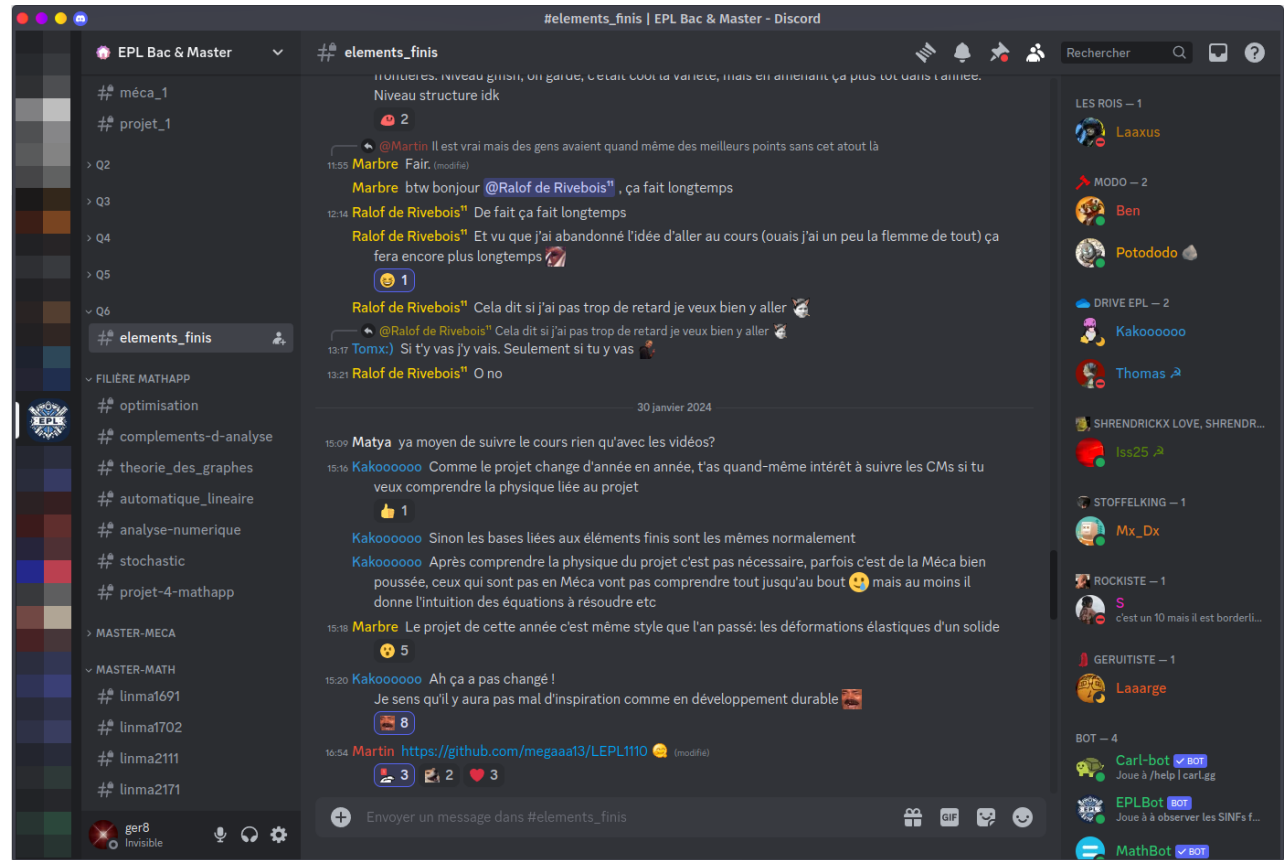
Finite
Elements





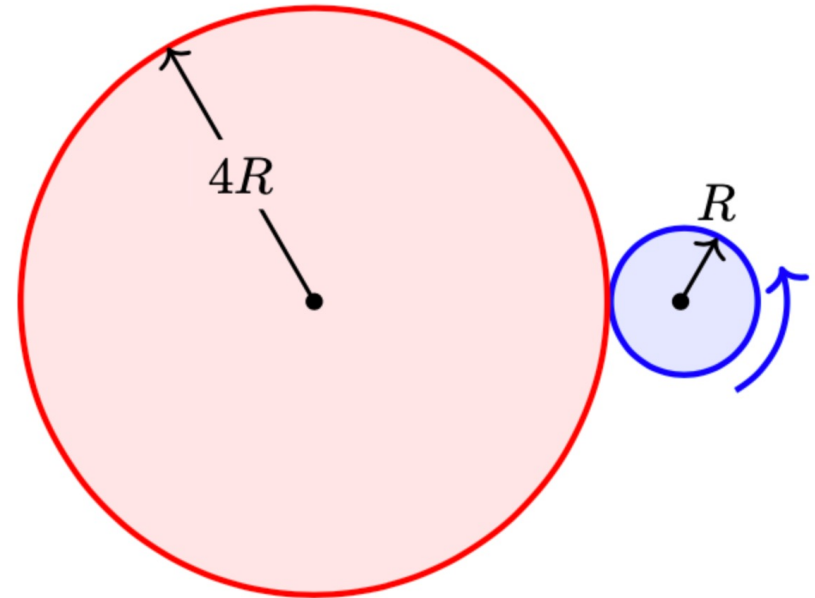
Nathan

Finite
Elements



Faut-il seulement
regarder les videos ?

Un petit test pour rire...



app.wooclap.com/events/LOYGXU/0

Sharepoints Informations Gazou Wikipedia Divers Php Actions Cours kine Finite Elements

Wooclap

← Quitter

Allez sur **wooclap.com** et utilisez le code **LOYGXU**

Un disque de rayon 1 roule sans glisser autour d'un cercle de ...

1 n=2 0% 0

2 n=3 0% 0

3 n=4 0% 0

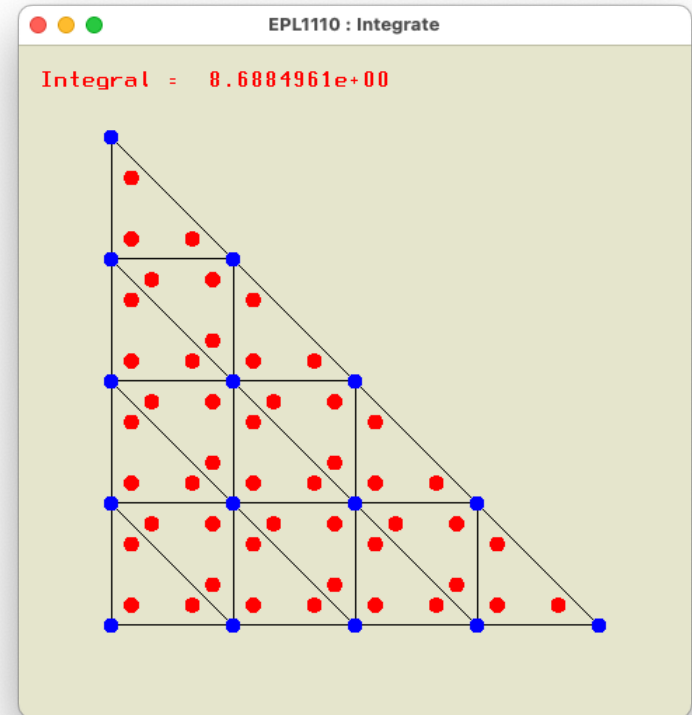
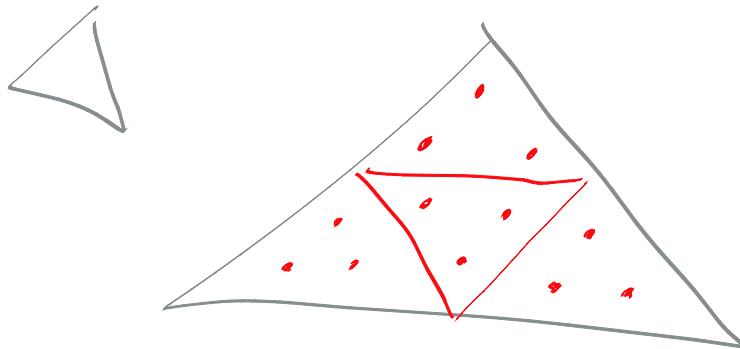
4 n=5 0% 0

wooclap

Votes 1 / 5 Messages 100 % 0 / 1

Homework 1

$$\underbrace{\int_{\hat{\Omega}} f(x, y) \, dx \, dy}_I \approx \underbrace{\sum_{k=1}^3 w_k f(x_k, y_k)}_{I_h}$$



Ecrire la règle de Hammer

hello.c

```
#include <stdio.h>

int main()
{
    printf("hello, world\n");
    return 0;
}
```

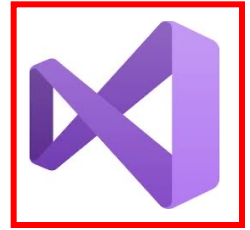
*Vous aimeriez apprendre à programmer,
mais vous ne savez pas par où commencer ?*

*(autrement dit : vous en avez marre des cours trop
compliqués que vous ne comprenez pas ? :-)*



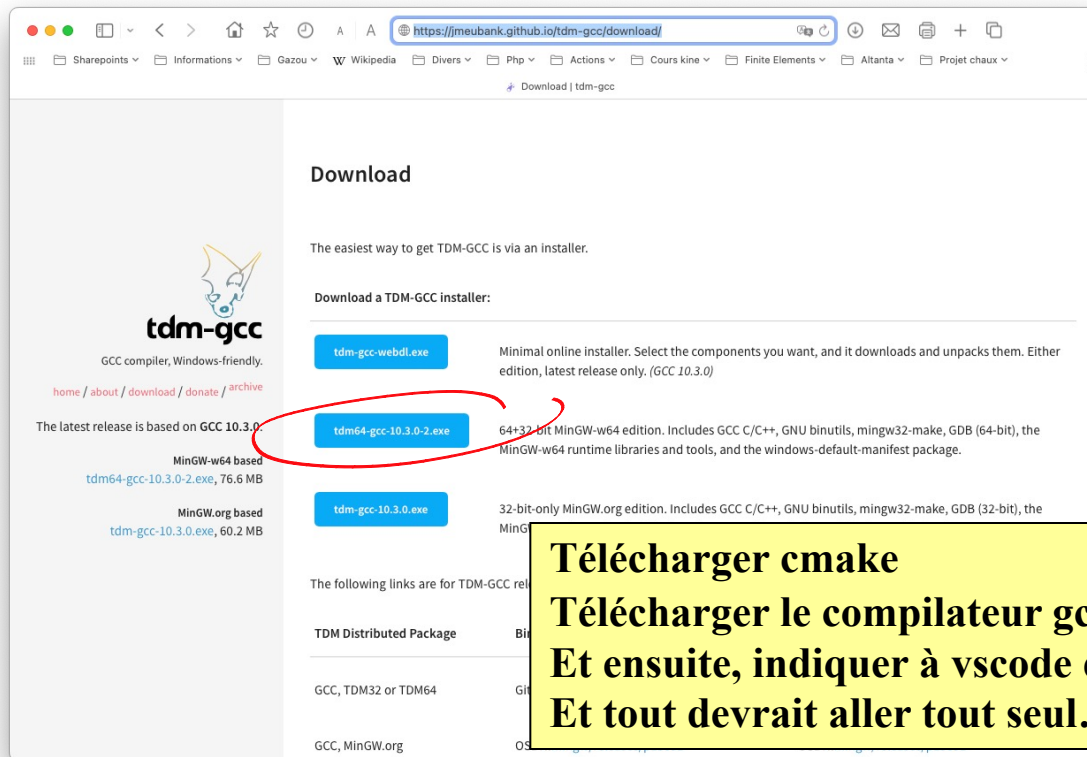
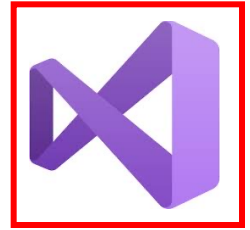
```
Cours1 — -bash — 71x16
Last login: Fri Feb 4 08:32:27 on ttys001
(base) mac-1U0-352:~ vl$ cd LaTeX/ep11110-2122/Slides/Cours1
(base) mac-1U0-352:Cours1 vl$ ls
a.out  hello.c
(base) mac-1U0-352:Cours1 vl$ touch hello.c
(base) mac-1U0-352:Cours1 vl$ cc hello.c
(base) mac-1U0-352:Cours1 vl$ ./a.out
hello, world
(base) mac-1U0-352:Cours1 vl$
```

Comment compiler hello.c sur votre ordinateur ?



```
Cours1 — -bash — 93x27
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$ touch hello.c
(base) mac-1U0-352:Cours1 vl$ cc -c hello.c
(base) mac-1U0-352:Cours1 vl$ ls
hello.c hello.o
(base) mac-1U0-352:Cours1 vl$ cc -o hello hello.c
(base) mac-1U0-352:Cours1 vl$ ./hello
hello, world
(base) mac-1U0-352:Cours1 vl$ cc -o yep hello.o
(base) mac-1U0-352:Cours1 vl$ ./yep
hello, world
(base) mac-1U0-352:Cours1 vl$
```

Avoir un compilateur sous Windows ?

A screenshot of a web browser showing the TDM-GCC download page. The browser's address bar displays 'https://meubank.github.io/tdm-gcc/download/'. The page has a sidebar on the left with the TDM-GCC logo and navigation links. The main content area is titled 'Download' and provides instructions on how to get TDM-GCC. It lists three download options: 'tdm-gcc-webdl.exe' (Minimal online installer), 'tdm64-gcc-10.3.0-2.exe' (64+32-bit MinGW-w64 edition, highlighted with a red circle), and 'tdm-gcc-10.3.0.exe' (32-bit-only MinGW.org edition). Below these, there are links for 'TDM Distributed Package', 'GCC, TDM32 or TDM64', and 'GCC, MinGW.org'.

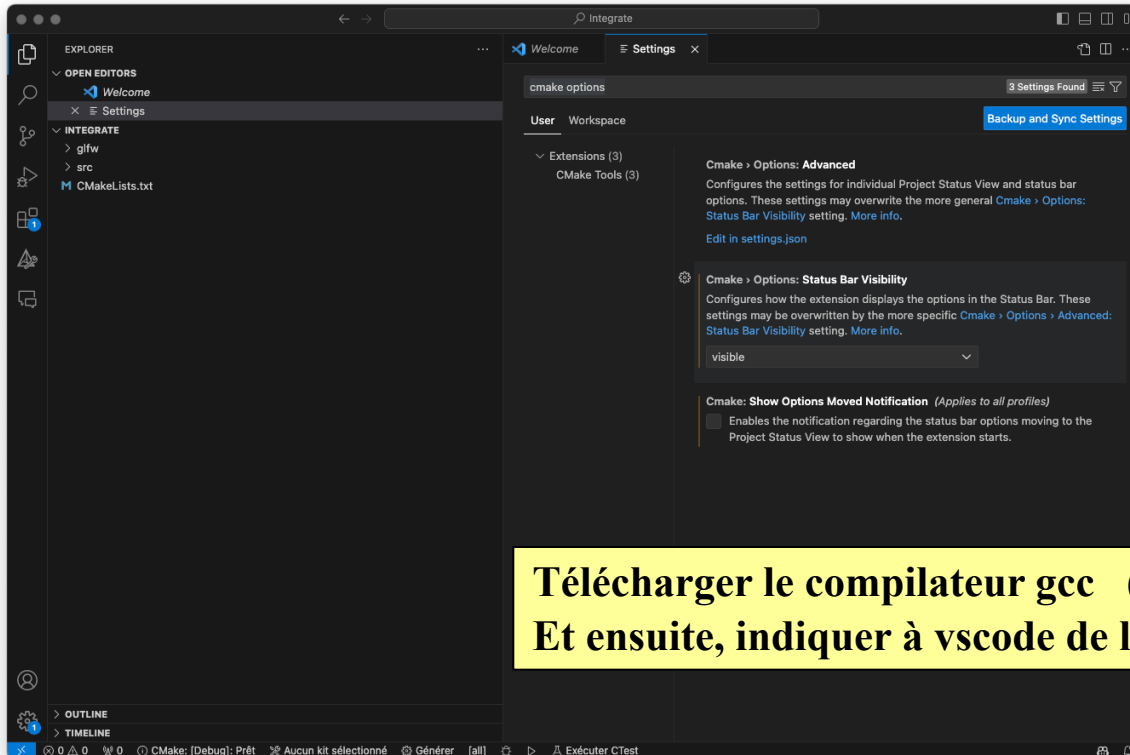
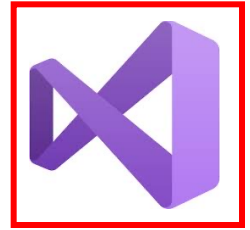
Télécharger cmake

Télécharger le compilateur gcc (pas celui de Microsoft !)

Et ensuite, indiquer à vscode de l'utiliser !

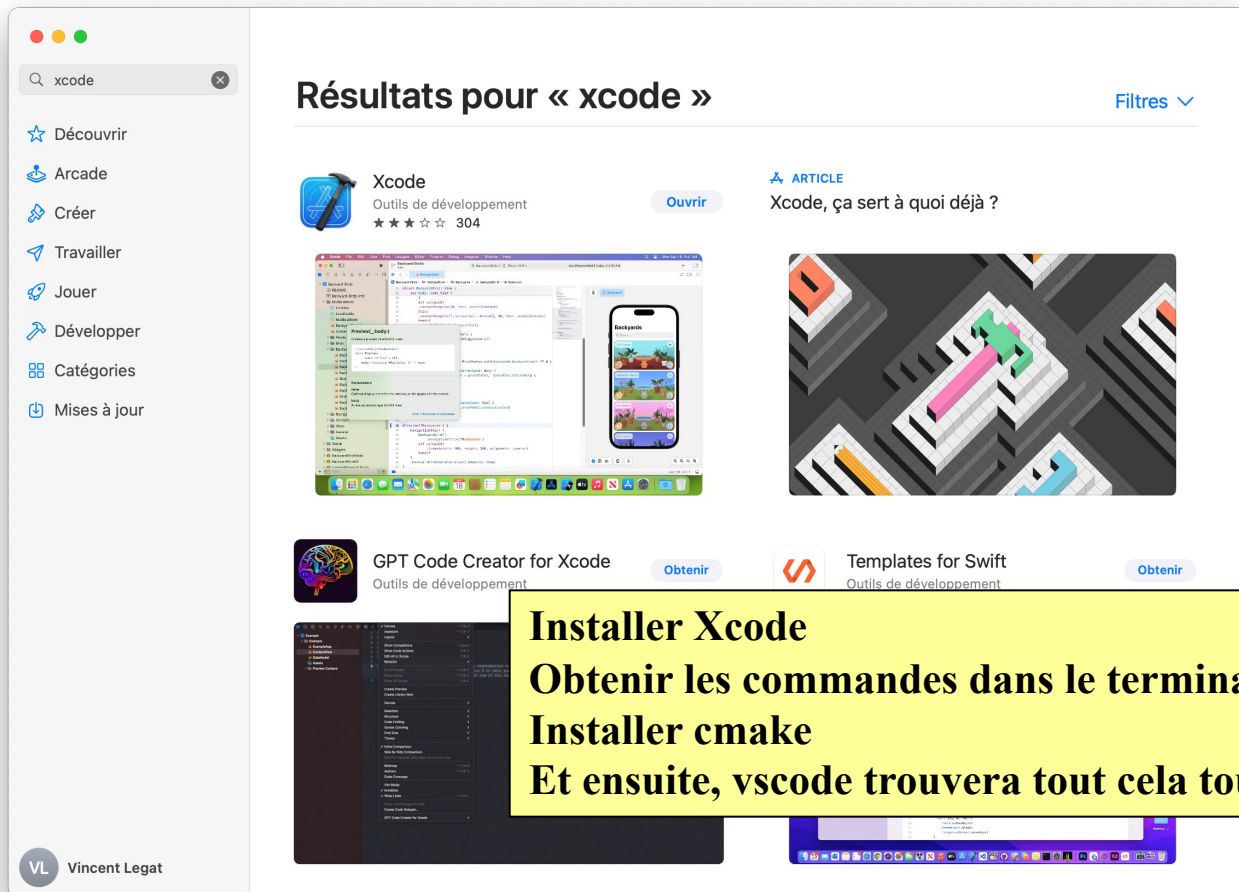
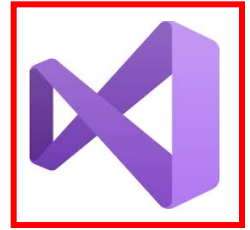
Et tout devrait aller tout seul.... (enfin, on espère !)

Avoir un compilateur sous Windows ?



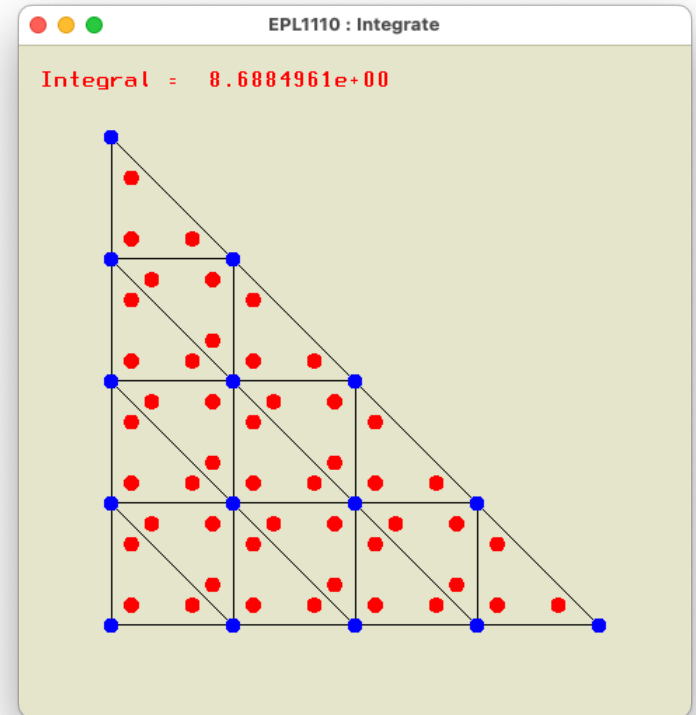
**Télécharger le compilateur gcc (pas celui de Microsoft !)
Et ensuite, indiquer à vscode de l'utiliser !**

Avoir un compilateur sous Mac ?



Homework 1

$$\underbrace{\int_{\hat{\Omega}} f(x, y) \, dx \, dy}_I \approx \underbrace{\sum_{k=1}^3 w_k f(x_k, y_k)}_{I_h}$$



Ecrire la règle de Hammer

mainBasic.c

```
#include <stdio.h>
#include <math.h>

double integrate(double x[3], double y[3], double(*f)(double,double));
double integrateRecursive(double x[3], double y[3], double(*f)(double,double), int n);

double fun(double x, double y) { return cos(x) + y * y; }
double stupid(double x, double y) { return 1.0; }

int main()
{
    double x[3] = {0, 1, 0};
    double y[3] = {0, 0, 1};
    int n;
    printf("Surface integration      : %14.7e \n", integrate(x,y,stupid));
    printf("More funny integration : %14.7e \n", integrate(x,y,fun));
    for (n=0; n <= 4; n++) {
        double I = integrateRecursive(x,y,fun,n);
        printf("Recursive integration (n = %2d) : %14.7e \n", n,I);
    }
    return 0;
}
```

homework.c et mainBasic.c

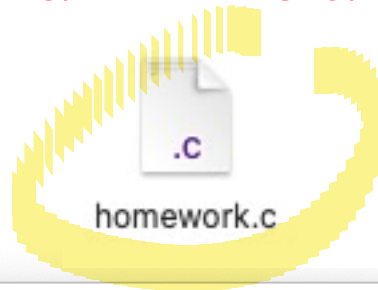
```
double integrate(double x[3], double y[3], double (*f) (double, double))
{
    double I = 3.14;
    return I;
}
```

```
double integrateRecursive(double x[3], double y[3], double (*f)(double,double), int n)
{
    double I = 0.0;
    return I;
}
```

```
double fun(double x, double y) { return cos(x) + y * y; }
double stupid(double x, double y) { return 1.0; }
```

```
int main()
{
    double x[3] = {0, 1, 0};
    double y[3] = {0, 0, 1};
    int n;
    printf("Surface integration      : %14.7e \n", integrate(x,y,stupid));
    printf("More funny integration : %14.7e \n", integrate(x,y,fun));
    for (n=0; n <= 1; n++) {
        double I = integrateRecursive(x,y,fun,n);
        printf("Recursive integration (n = %2d) : %14.7e \n", n,I); }
    return 0;
}
```

Comment compiler le devoir sur votre ordinateur ?



```
Cours1 — -bash — 79x29
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$
(base) mac-1U0-352:Cours1 vl$ cc -o myFem mainBasic.c homework.c
(base) mac-1U0-352:Cours1 vl$ ./myFem
Surface integration      : 3.1400000e+00
More funny integration  : 3.1400000e+00
Recursive integration (n = 0) : 0.0000000e+00
Recursive integration (n = 1) : 0.0000000e+00
Recursive integration (n = 2) : 0.0000000e+00
Recursive integration (n = 3) : 0.0000000e+00
Recursive integration (n = 4) : 0.0000000e+00
(base) mac-1U0-352:Cours1 vl$ cc -o myFem mainBasic.c homeworkSoluce.c
(base) mac-1U0-352:Cours1 vl$ ./myFem
Surface integration      : 5.0000000e-01
More funny integration  : 5.4302895e-01
Recursive integration (n = 0) : 5.4302895e-01
Recursive integration (n = 1) : 2.1721229e+00
Recursive integration (n = 2) : 8.6884961e+00
Recursive integration (n = 3) : 3.4753986e+01
Recursive integration (n = 4) : 1.3901594e+02
(base) mac-1U0-352:Cours1 vl$
```


Bon : c'est aussi simple ?

perso.uclouvain.be/vincent.l

Yahoo Apple Twitter Gazou Wikipedia Informations Divers Php Actions Cours kiné Altanta

linux logo - Recherche Google

EPL1110 News Horaire Documents

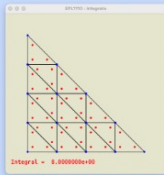
 Introduction aux éléments finis (LEPL1110)
Vincent Legat
Jean-François Remacle
Louvain School of Engineering
Université catholique de Louvain

Il faut d'abord t'identifier :-)

News Documents Videos & podcasts !

Comment obtenir un exécutable sur votre ordinateur ? Programmes Python Devoirs en C

Devoirs en C...
Les énoncés seront disponibles progressivement...

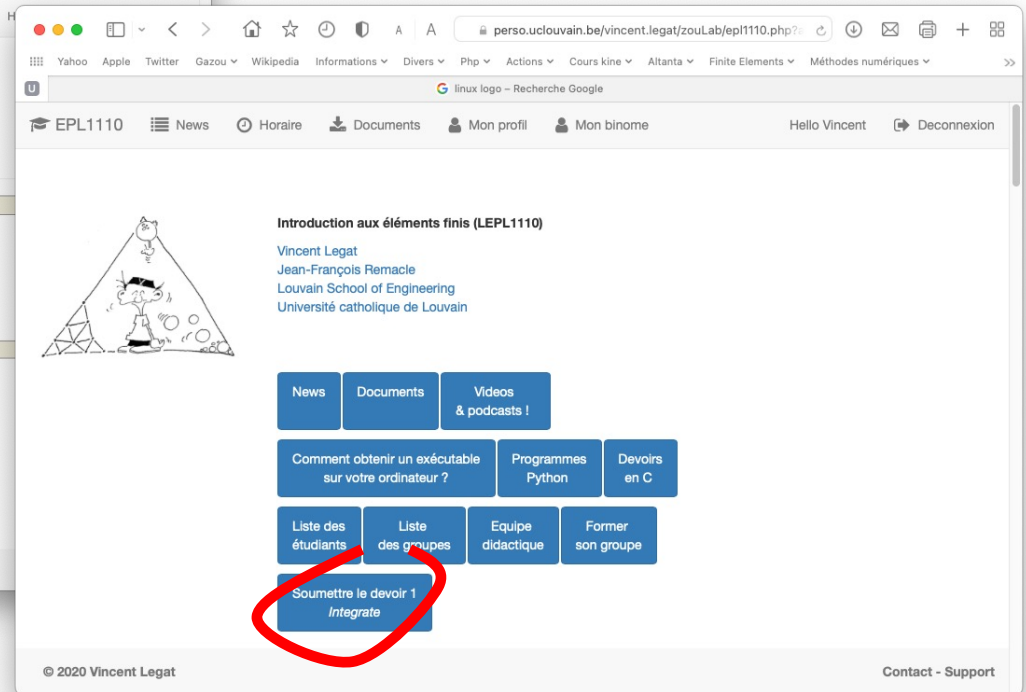
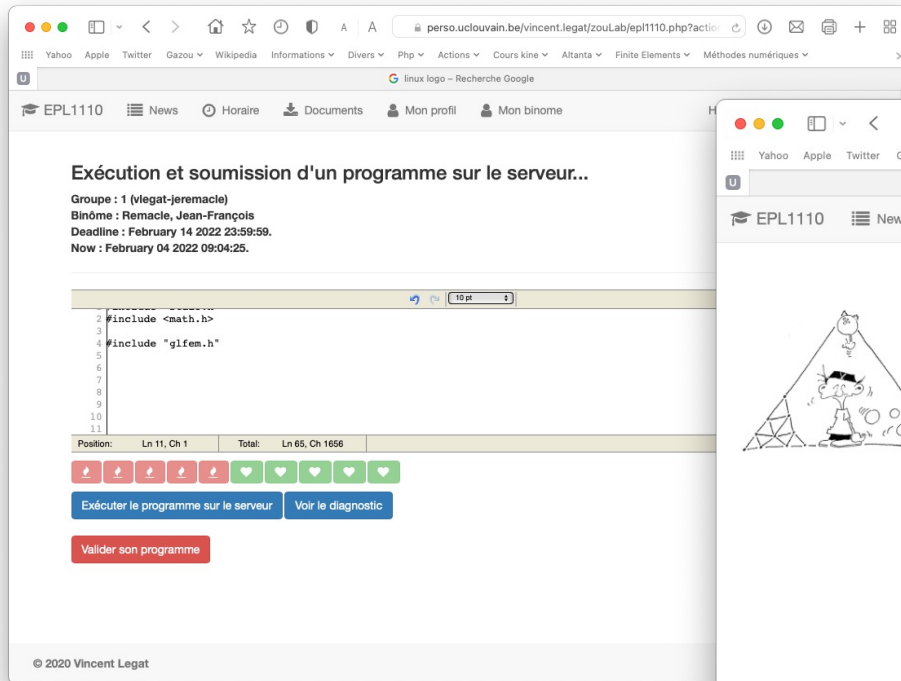
 **Devoir 1 : Integrate** (02-02-2022)
Projet à télécharger : Integrate.zip
Enoncé du devoir : Integrate.pdf
Deadline : **Lundi 14 février 2022 à 23h59**

© 2020 Vincent Legat Contact - Support

```
src — -bash — 90x27
IntegrateWithoutBov    homework.c    myFem
hello                  homeworkSolve.c  yep
hello.c                mainBasic.c

(base) mac-1U0-352:Cours1 v1$ cd Integrate
(base) mac-1U0-352:Integrate v1$ ls
CMakeLists.txt  glfw      src
(base) mac-1U0-352:Integrate v1$ cd src
(base) mac-1U0-352:src v1$ ls
fem.c            glfem.c      homework.c
fem.h            glfem.h      main.c
(base) mac-1U0-352:src v1$ gcc -o myFem *.c
In file included from glfem.c:10:
./glfem.h:21:10: fatal error: 'GLFW/glfw3.h' file not found
#include <GLFW/glfw3.h>
^
1 error generated.
In file included from homework.c:3:
./glfem.h:21:10: fatal error: 'GLFW/glfw3.h' file not found
#include <GLFW/glfw3.h>
^
1 error generated.
In file included from main.c:5:
./glfem.h:21:10: fatal error: 'GLFW/glfw3.h' file not found
#include <GLFW/glfw3.h>
^
1 error generated.
(base) mac-1U0-352:src v1$
```

Eh non !



Soumettre votre devoir !

**Tous les devoirs seront corrigés de manière automatique !
Bien veiller à ce que la version soumise soit bien compilée !
Aucune soumission tardive ne sera admise !
Les devoirs sont réalisés individuellement !**

perso.uclouvain.be/vincent.legat/zouLab/epi

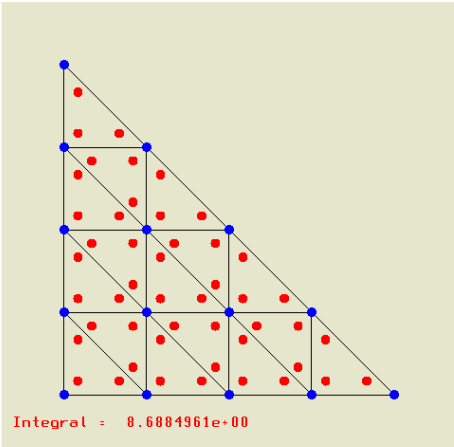
Yahoo Apple Twitter Gazou Wikipedia Informations Divers Php Actions Cours kine Altanta Finite Elements

linux logo - Recherche Google

EPL1110 News Horaire Documents Mon profil Mon binome

Exécution et soumission d'un programme sur le serveur...

Groupe : 1 (vlegat-jeremacle)
Binôme : Remacle, Jean-François
Deadline : February 14 2022 23:59:59.
Now : February 04 2022 09:06:11.



Integral = 8.6884961e+00

© 2020 Vincent Legat

perso.uclouvain.be/vincent.legat/zouLab/epi

Yahoo Apple Twitter Gazou Wikipedia Informations Divers Php Actions Cours kine Altanta Finite Elements

linux logo - Recherche Google

EPL1110 News Horaire Documents Mon profil Mon binome Hello Vincent Deconnexion

```
int i,j;
const int nodes[4][3] = {{0,3,5},{3,1,4},{5,4,2},{3,4,5}};
const double xsi[6] = {0.0,1.0,0.0,0.5,0.5,0.0};
const double eta[6] = {0.0,0.0,1.0,0.0,0.5,0.5};
double xLoc[3];
double yLoc[3];

if (n <= 0) return integrate(x,y,f);

double I = 0.0;
for (i=0; i<4; i++) {
    for (j=0; j<3; j++) {
        double xsiLoc = xsi[nodes[i][j]];
        double etaLoc = eta[nodes[i][j]];
        xLoc[j] = interpolate(x,xsiLoc,etaLoc);
        yLoc[j] = interpolate(y,xsiLoc,etaLoc);
        I += integrateRecursive(xLoc,yLoc,f,n-1);
    }
}

return I;
}
```

👍 👍 👍 👍 👍 ❤️ ❤️ ❤️ ❤️

Faire une nouvelle soumission Voir le diagnostic

Valider son programme

© 2020 Vincent Legat Contact - Support

Valider
et verifier son devoir !

homework.c

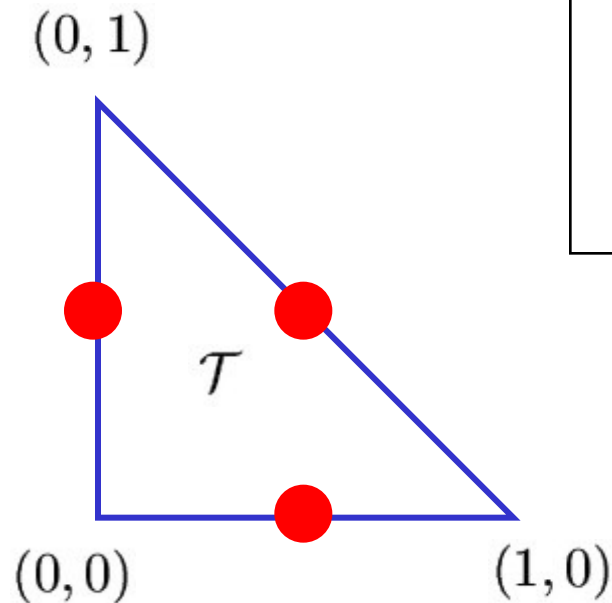
La solution doit se trouver dans le fichier homework.c uniquement.... On ne regarde jamais main.c !

```
#include <stdio.h>
#include <math.h>

double integrate(double x[3], double y[3], double (*f) (double, double))
{
    double I = 3.14;
    int I;
    for (i=0; i<3; i++)
        printf("    === node %d : %14.7e %14.7e \n",i+1,x[i],y[i]);
    return I;
}

double integrateRecursive(double x[3], double y[3], double (*f) (double,double), int n)
{
    double I = 0.0;
    return I;
}
```

Intégration sur un triangle : Règle de Hammer à 3 points



$$\underbrace{\int_{\mathcal{T}} f(x,y) \, dx \, dy}_I \approx \underbrace{\sum_{k=1}^3 w_k f(X_k, Y_k)}_{I^h}$$

	X_k	Y_k	w_k
1	0.5	0.0	1/6
2	0.5	0.5	1/6
3	0.0	0.5	1/6

Démontrer que la formule de Hammer à trois points permet d'intégrer exactement n'importe quel polynôme à deux variables de degré deux : $a + bx + cy + dx^2 + ey^2 + fxy$

Question

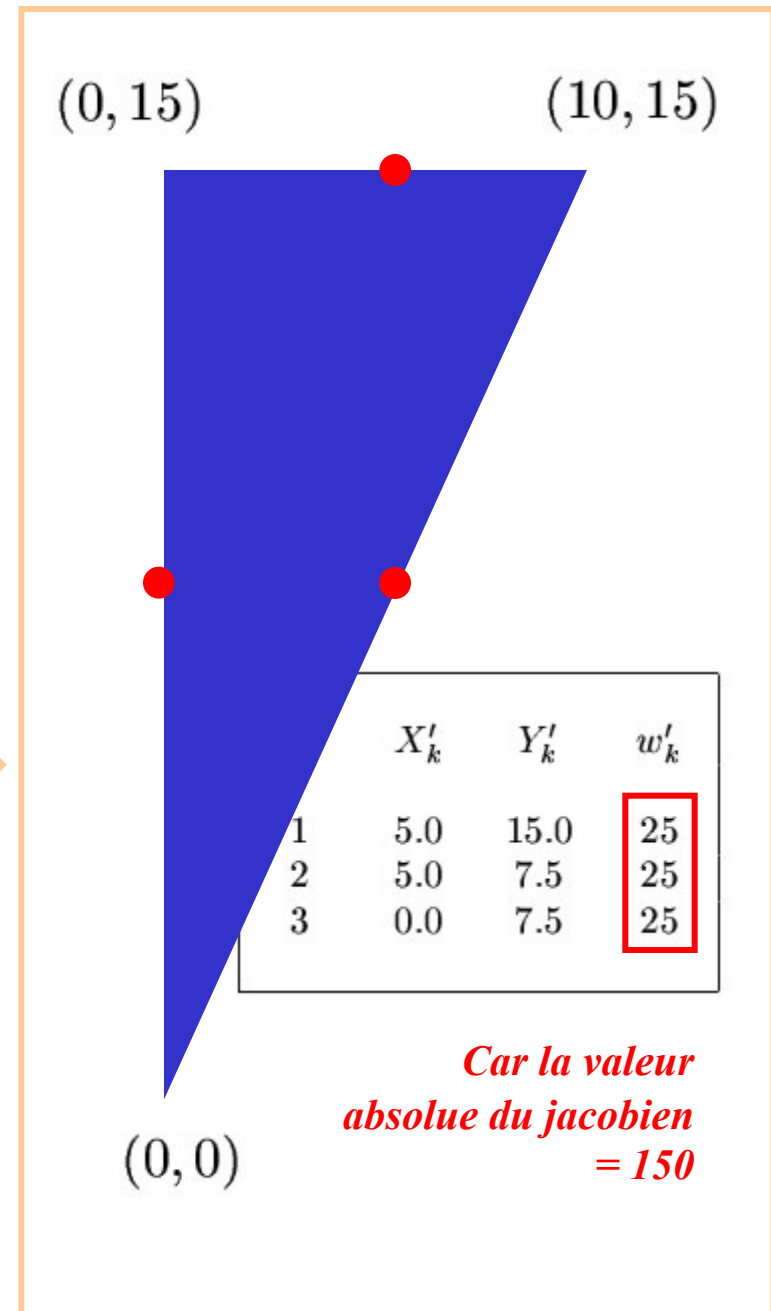
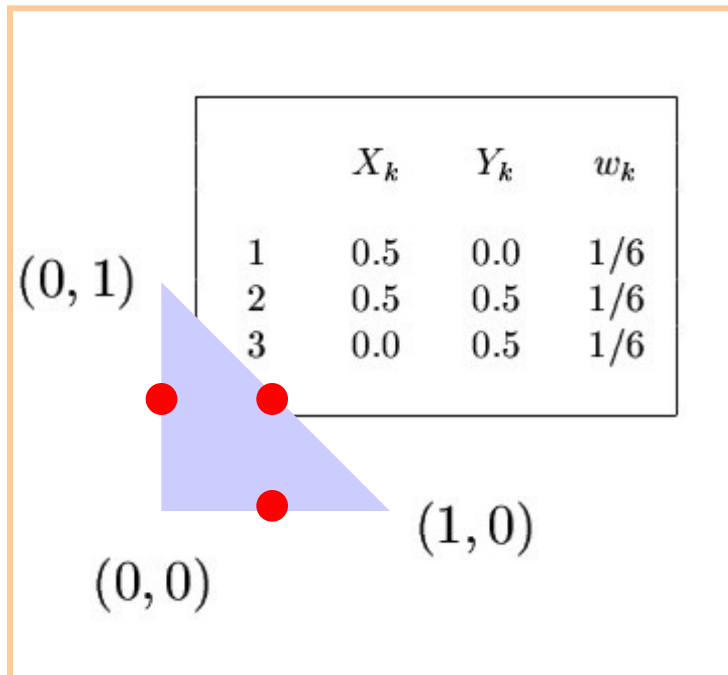
$$\begin{aligned}
 I &= \frac{a}{2} + b \int_0^1 x \int_0^{1-x} dy \, dx + c \int_0^1 y \int_0^{1-y} dx \, dy \\
 &\quad + d \int_0^1 x^2 \int_0^{1-x} dy \, dx + e \int_0^1 y^2 \int_0^{1-y} dx \, dy + f \int_0^1 x \int_0^{1-x} y \, dy \, dx \\
 &= \frac{a}{2} + b \left[\frac{x^2}{2} - \frac{x^3}{3} \right]_0^1 + c \left[\frac{y^2}{2} - \frac{y^3}{3} \right]_0^1 \\
 &\quad + d \left[\frac{x^3}{3} - \frac{x^4}{4} \right]_0^1 + e \left[\frac{y^3}{3} - \frac{y^4}{4} \right]_0^1 + f \left[\frac{x^2}{4} - \frac{x^3}{3} + \frac{x^4}{8} \right]_0^1 \\
 &= \frac{a}{2} + \frac{b}{6} + \frac{c}{6} + \frac{d}{12} + \frac{e}{12} + \frac{f}{24} \\
 &= I^h
 \end{aligned}$$



**Degré de
précision**

Et un autre triangle ?

$$\begin{aligned}x' &= 10x \\ y' &= 15 - 15y\end{aligned}$$



Effectuons cela ensemble !

The screenshot shows a web browser window displaying the zouLab 1.0.0 website for EPL1110. The browser's address bar shows the URL `perso.uclouvain.be/vincent.legat/zouLab/epl1110`. The website's navigation bar includes links for EPL1110, News, Horaire, Documents, and a login button labeled 'S'identifier'. Below the navigation bar, there are several blue buttons: 'News', 'Documents', 'Videos & podcasts !', 'Installer vscode et copilot sur votre ordinateur !', 'Programmes Python', and 'Devoirs en C'. The 'Devoirs en C' button is highlighted in green. Below these buttons, the section 'Devoirs en C...' is displayed, with the text 'Les énoncés seront disponibles progressivement...'. A blue box contains a thumbnail image of a document titled 'EPL1110 - Integrate' showing a triangular grid of red dots. To the right of the thumbnail, the following information is provided: 'Devoir 1 : Integrate (25/01/2024)', 'Projet à télécharger : [Integrate.zip](#)', 'Enoncé du devoir : [Integrate.pdf](#)', and 'Deadline : **Lundi 19 février 2024 à 23h59**'. The footer of the website shows '© 2020 Vincent Legat' on the left and 'Contact - Support' on the right.

perso.uclouvain.be/vincent.legat/zouLab/epl1110

zouLab 1.0.0 : epl1110

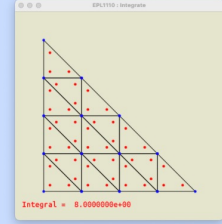
EPL1110 News Horaire Documents S'identifier

News Documents Videos & podcasts !

Installer vscode et copilot sur votre ordinateur ! Programmes Python Devoirs en C

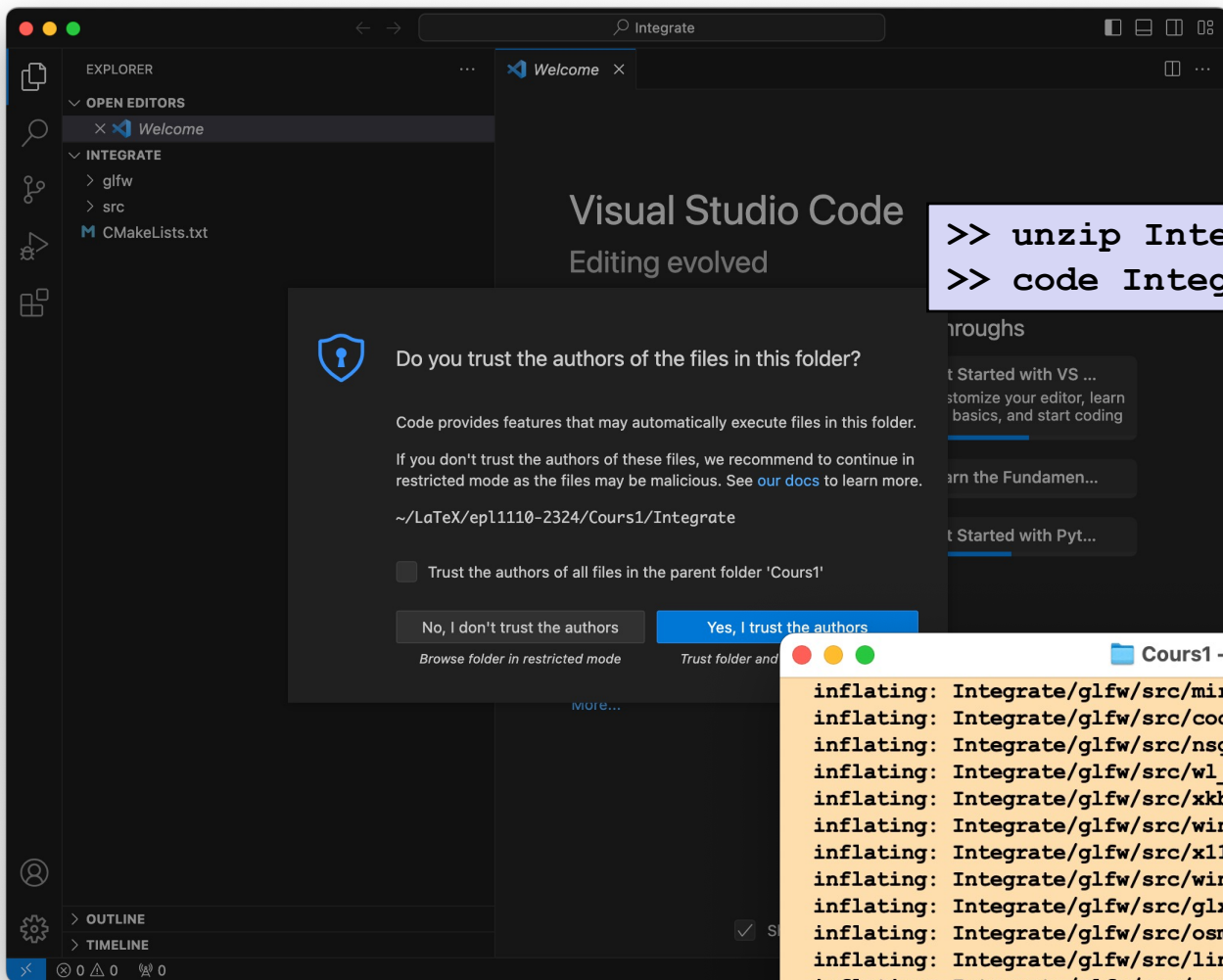
Devoirs en C...

Les énoncés seront disponibles progressivement...

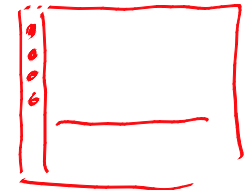


Devoir 1 : Integrate (25/01/2024)
Projet à télécharger : [Integrate.zip](#)
Enoncé du devoir : [Integrate.pdf](#)
Deadline : **Lundi 19 février 2024 à 23h59**

© 2020 Vincent Legat Contact - Support

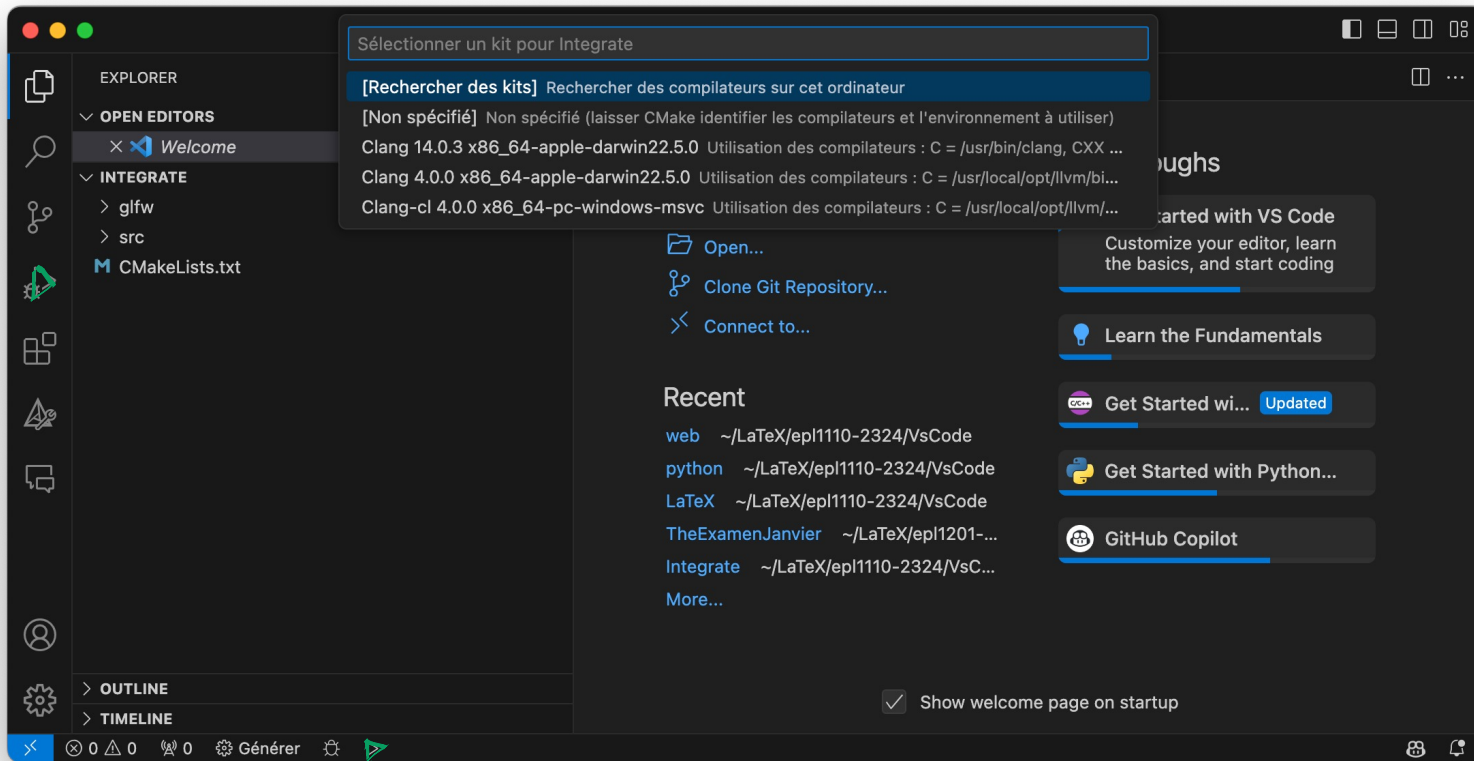


```
>> unzip Integrate.zip  
>> code Integrate
```



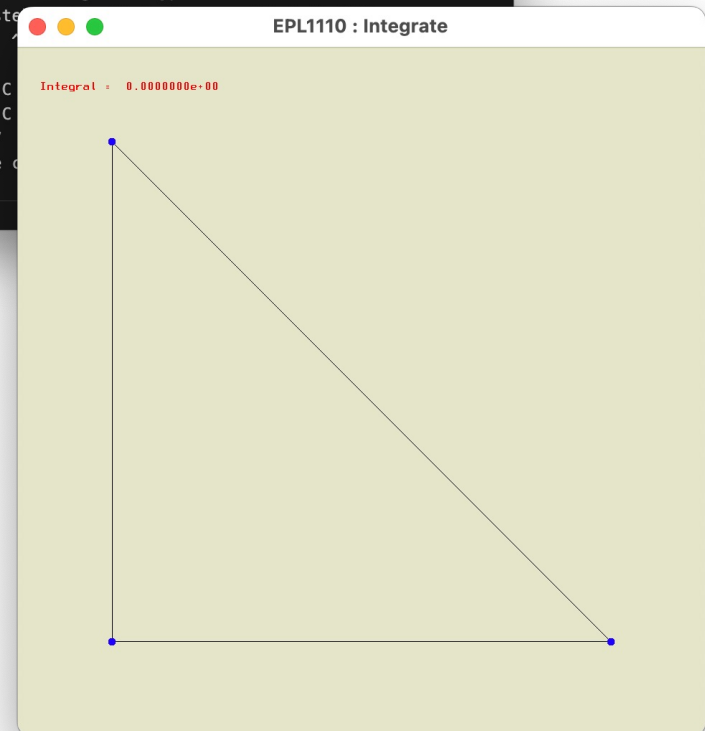
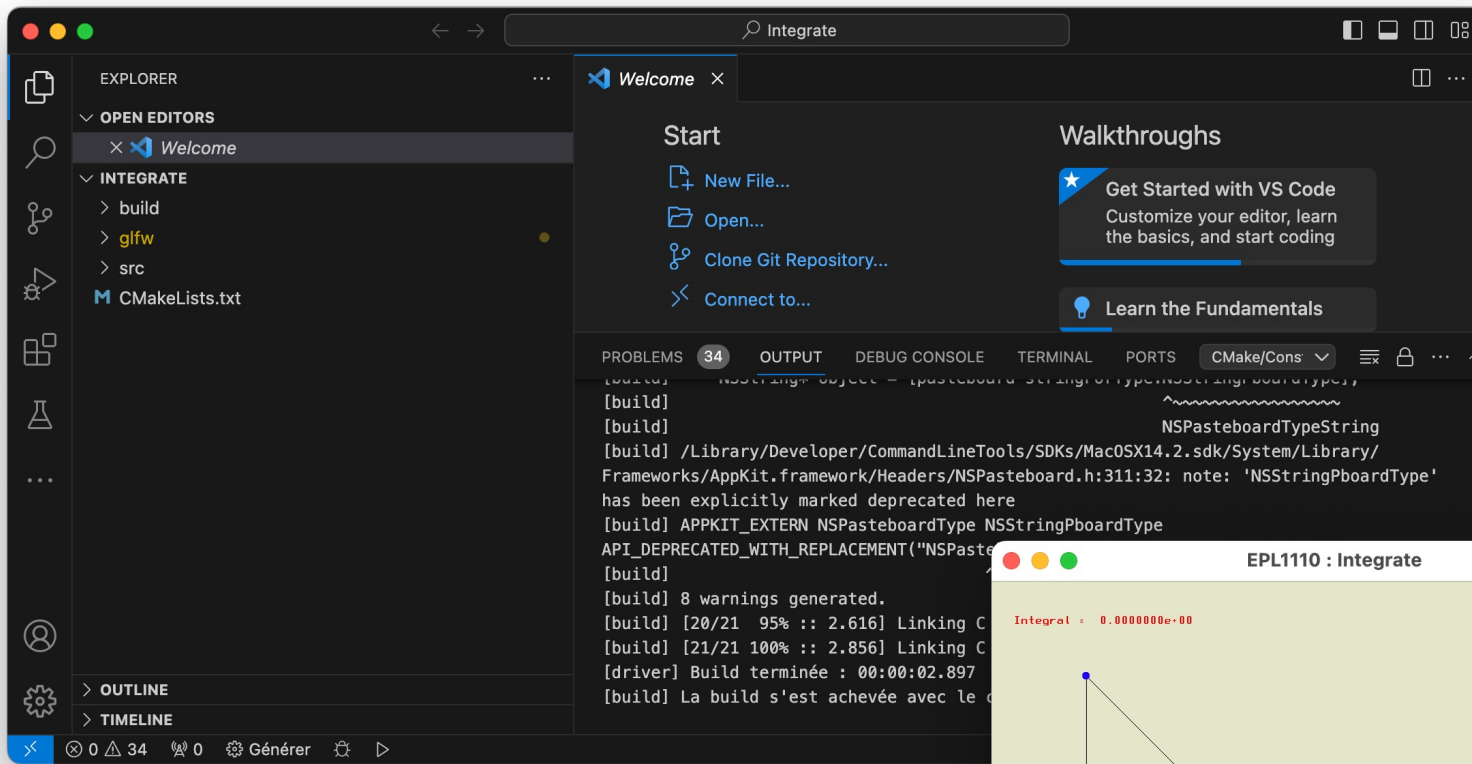
code integrate

```
Cours1 — -bash — 72x22  
inflating: Integrate/glfw/src/mir_platform.h  
inflating: Integrate/glfw/src/cocoa_time.c  
inflating: Integrate/glfw/src/nsgl_context.m  
inflating: Integrate/glfw/src/wl_platform.h  
inflating: Integrate/glfw/src/xkb_unicode.c  
inflating: Integrate/glfw/src/win32_window.c  
inflating: Integrate/glfw/src/x11_monitor.c  
inflating: Integrate/glfw/src/win32_platform.h  
inflating: Integrate/glfw/src/glx_context.c  
inflating: Integrate/glfw/src/osmesa_context.h  
inflating: Integrate/glfw/src/linux_joystick.c  
inflating: Integrate/glfw/src/cocoa_monitor.m  
inflating: Integrate/glfw/src/window.c  
inflating: Integrate/src/fem.h  
inflating: Integrate/src/.DS_Store  
inflating: Integrate/src/glfem.h  
inflating: Integrate/src/fem.c  
inflating: Integrate/src/main.c  
inflating: Integrate/src/homework.c  
inflating: Integrate/src/glfem.c  
(base) mac-1U0-352:Cours1 vl$ code Integrate  
(base) mac-1U0-352:Cours1 vl$
```

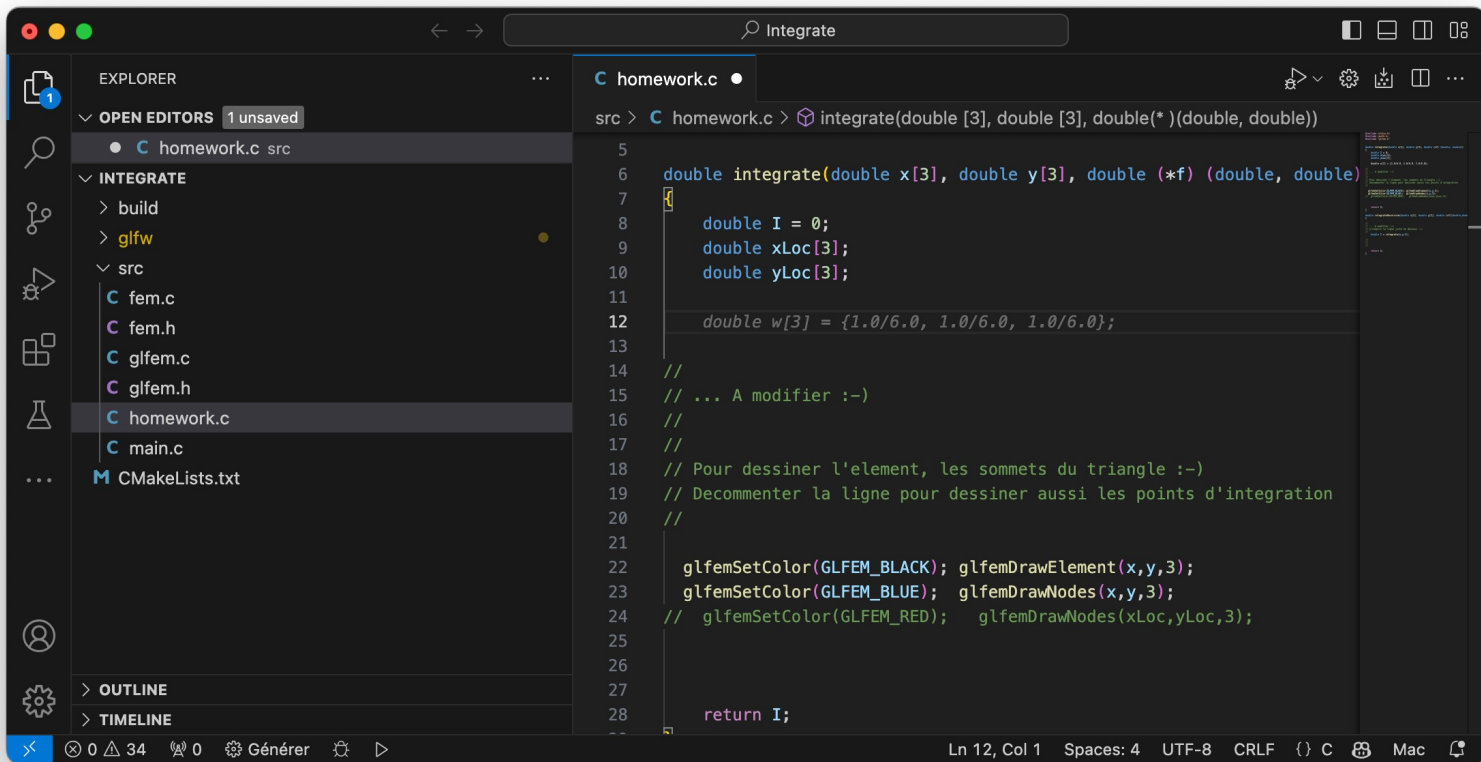


CLIQUER
ici :-)

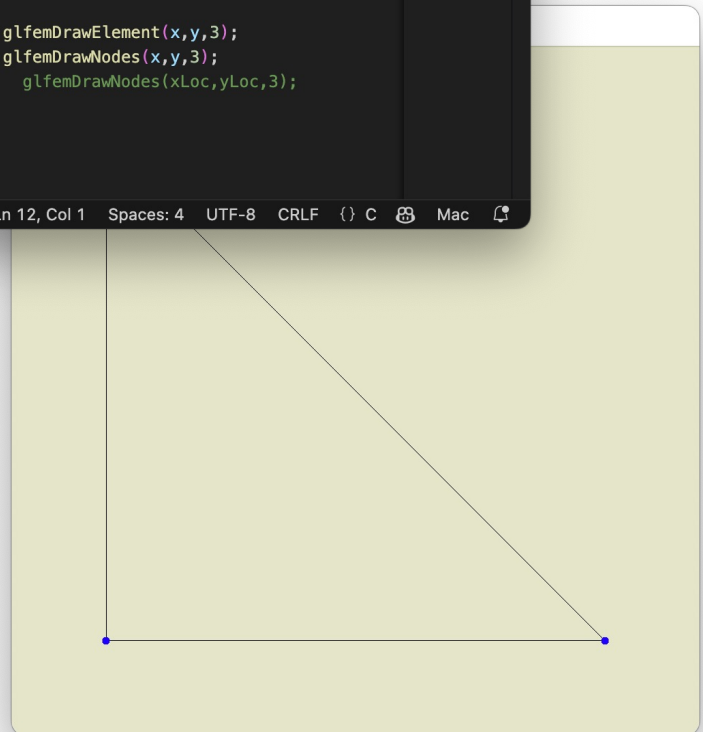
Sélectionner le compilateur !
Lancer la compilation et l'exécution !

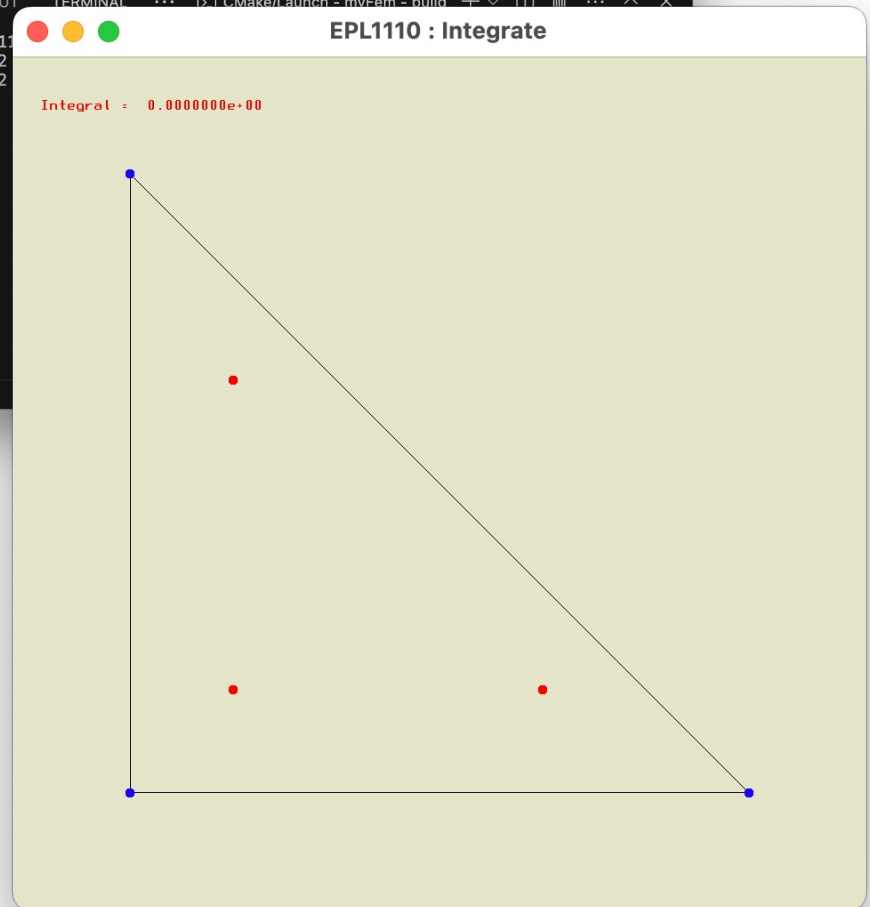
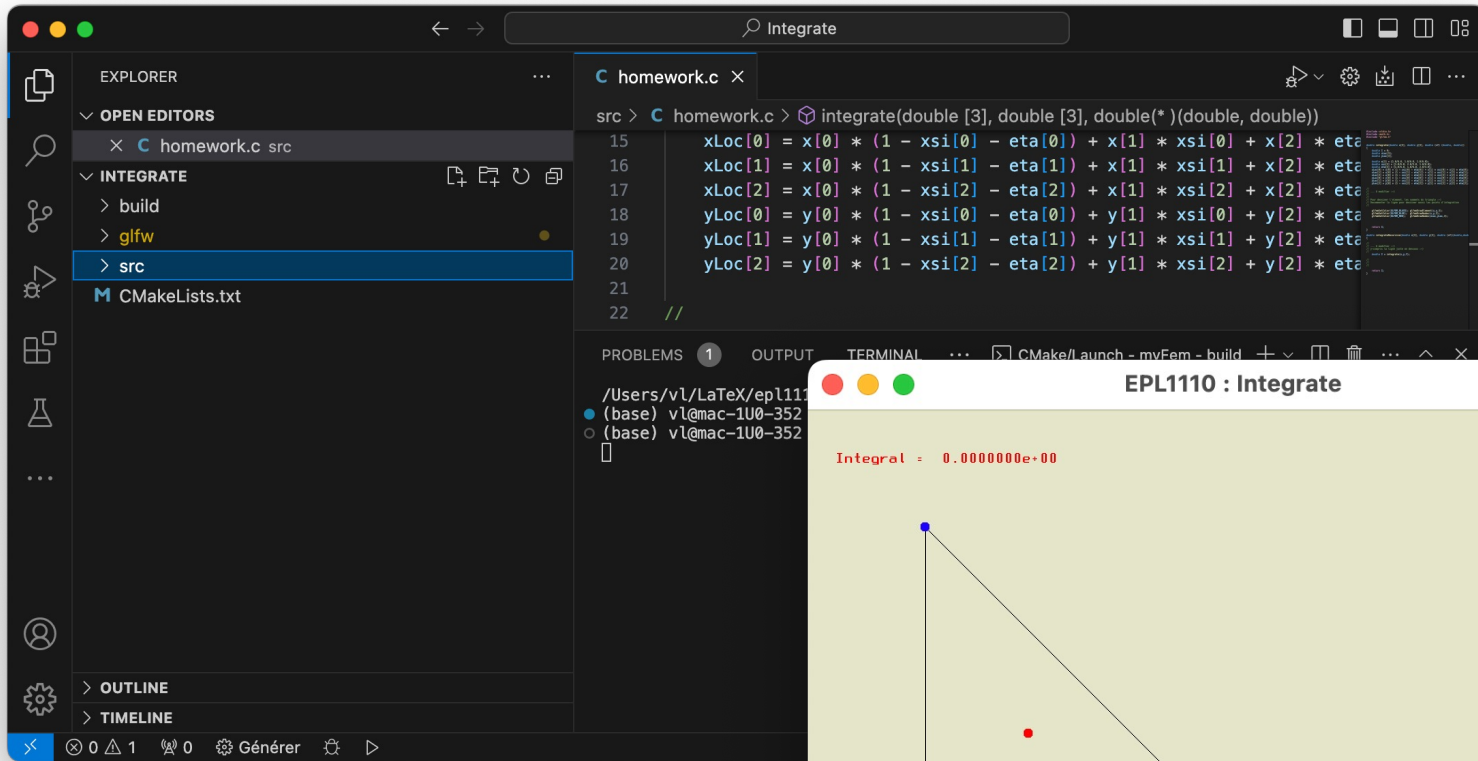


Et zou !

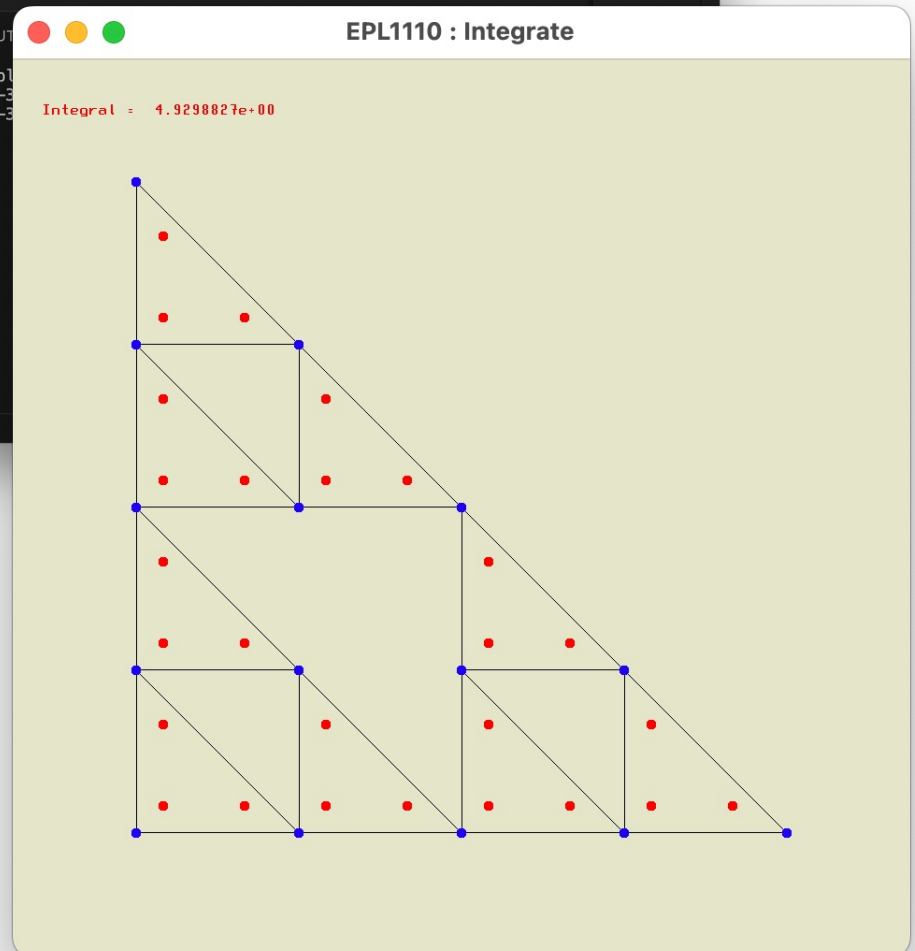
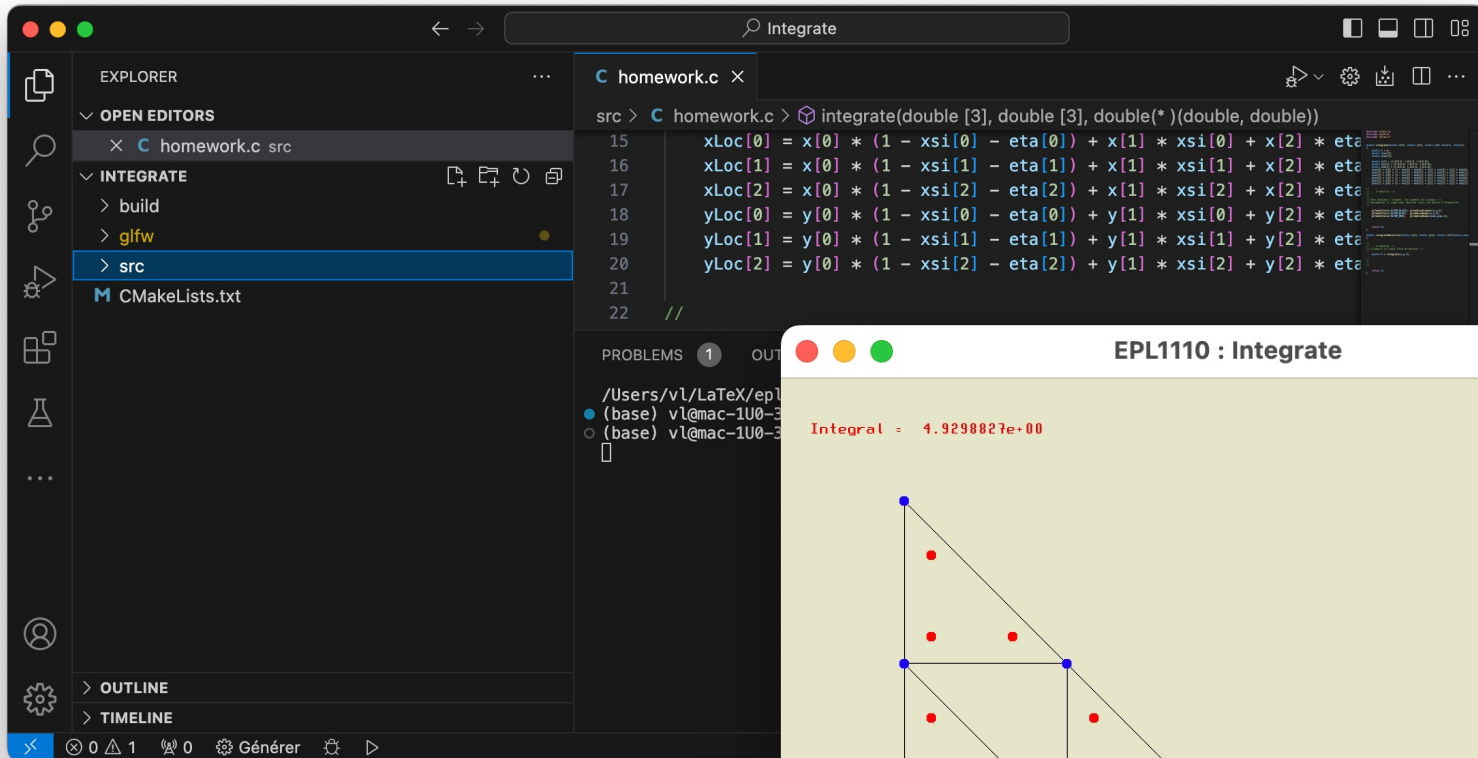


Et l'intelligence artificielle
va écrire le devoir...

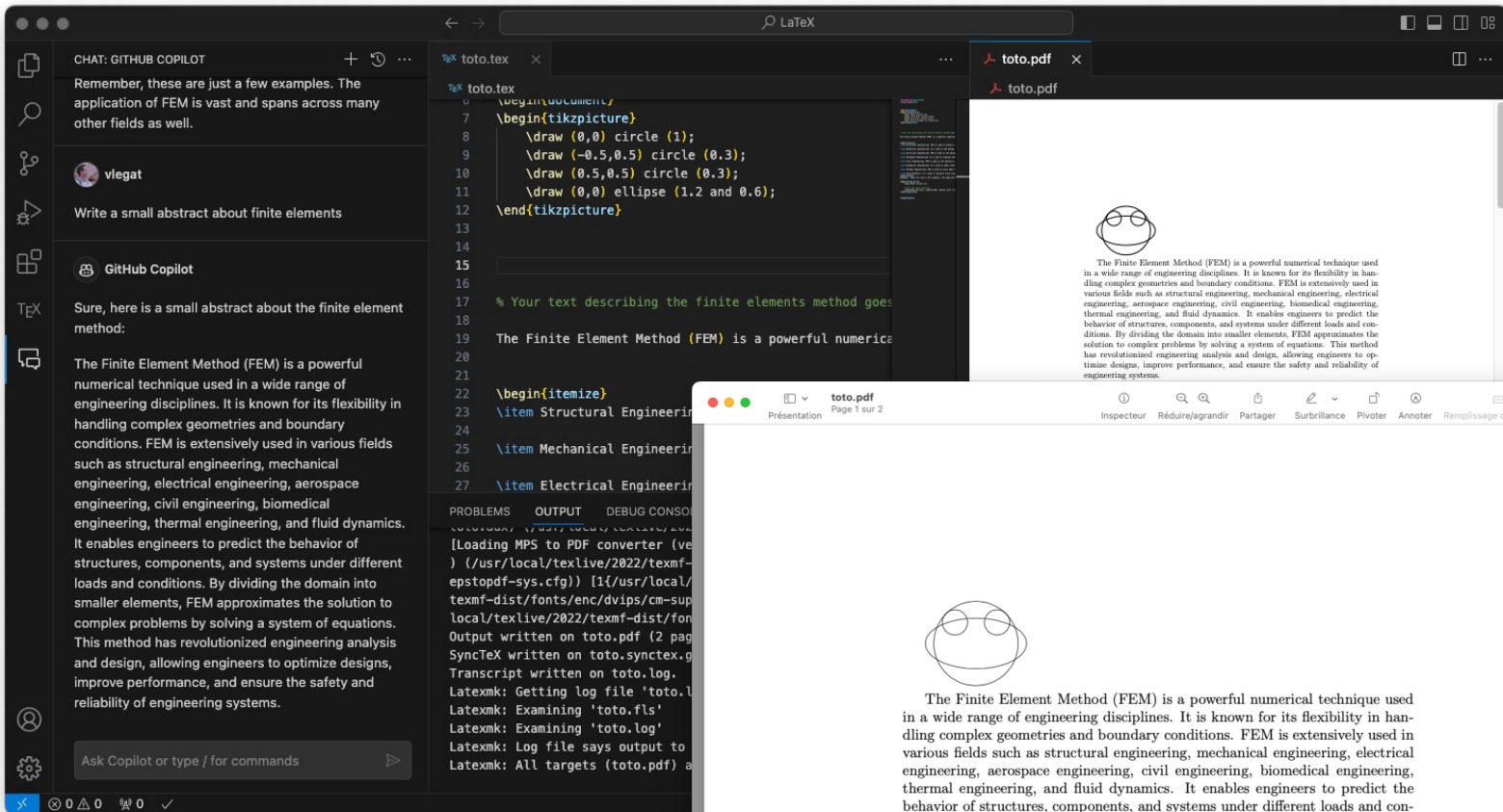




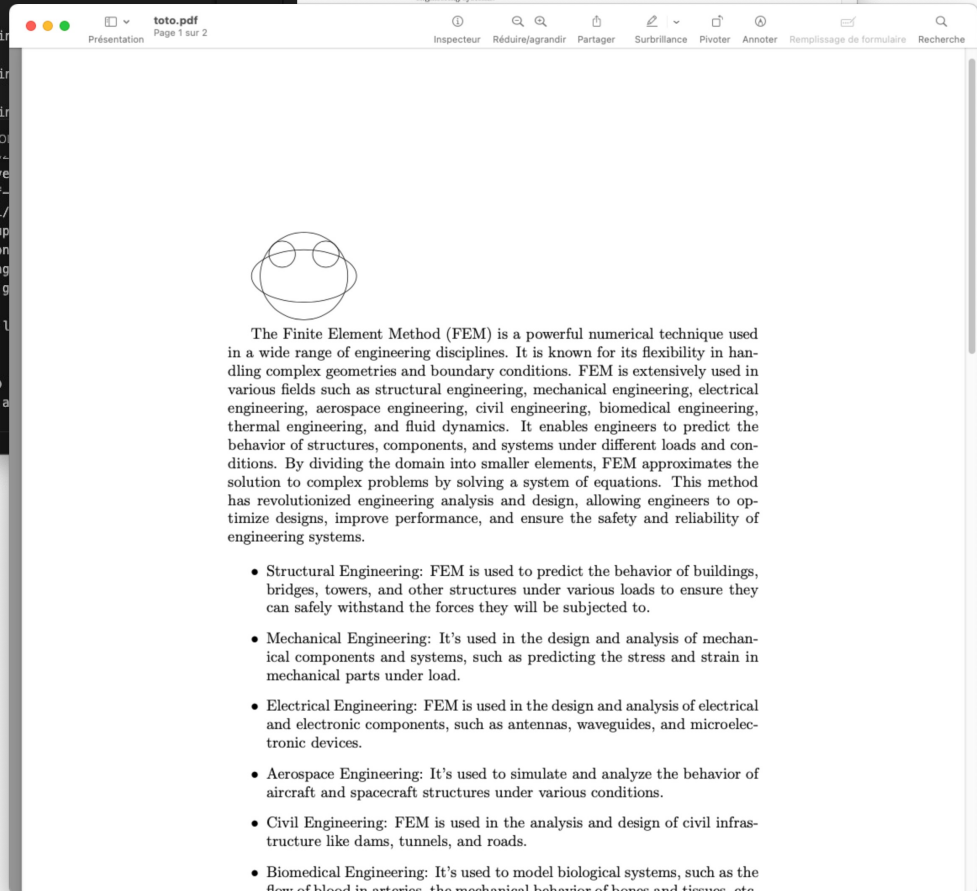
Et copilot
a fait le job !



Enfin
plus ou moins....

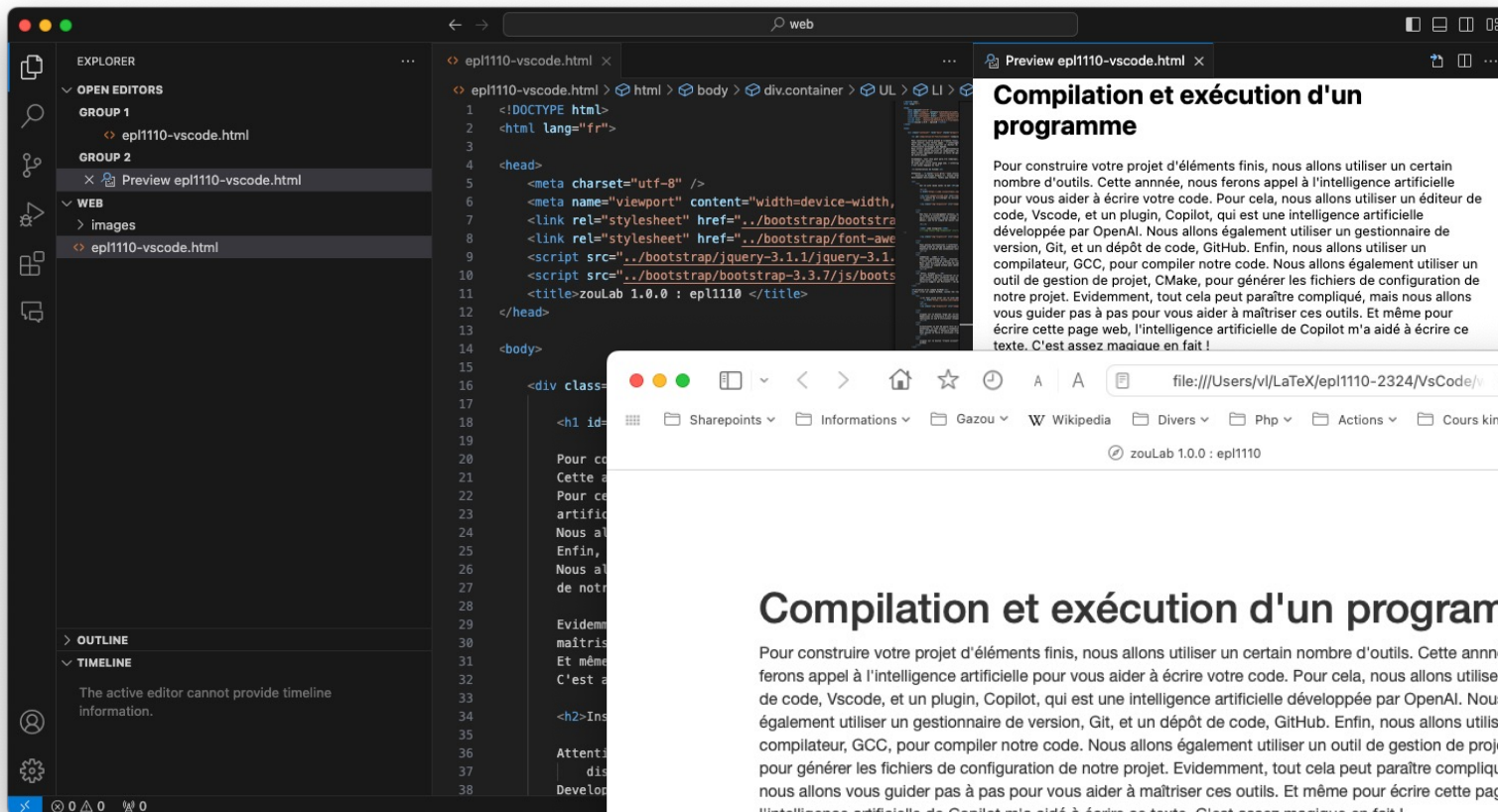


Et même
le rapport en LaTeX !

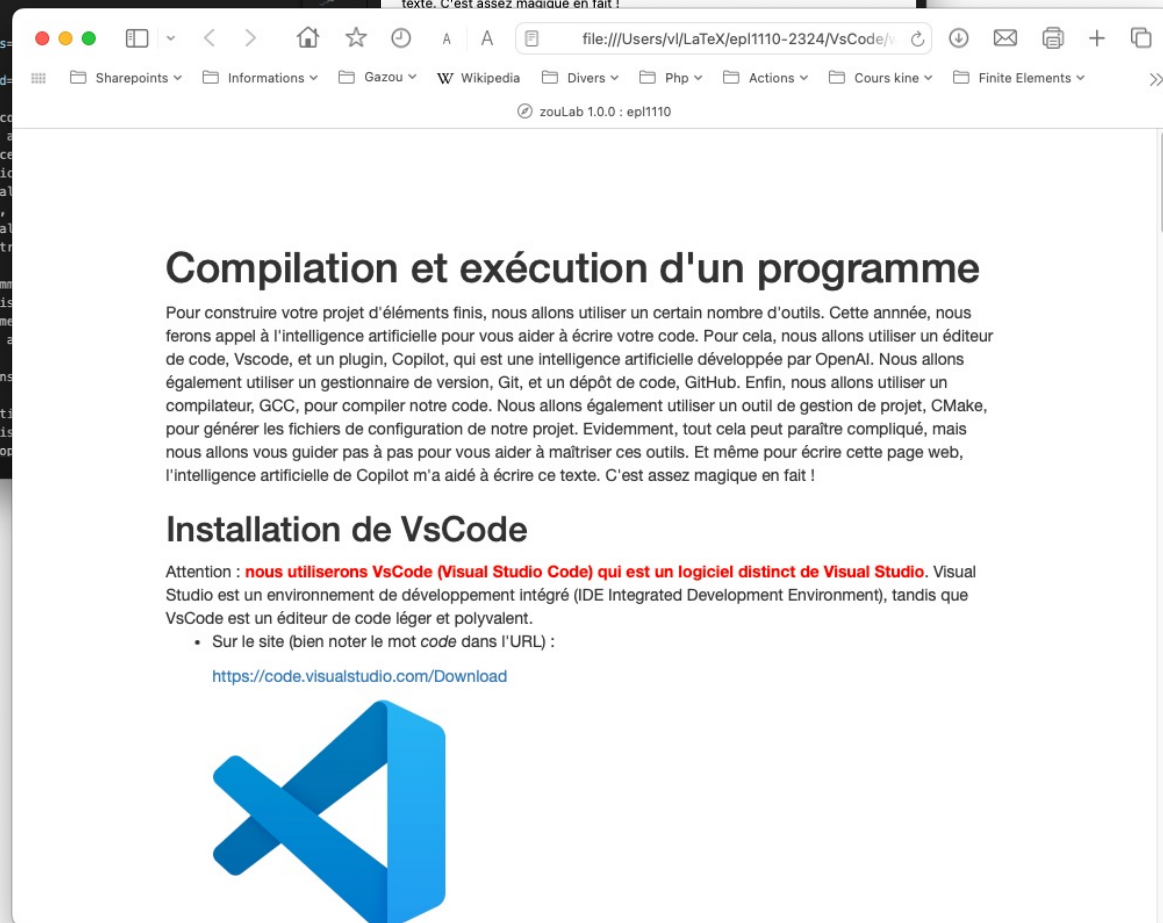


The Finite Element Method (FEM) is a powerful numerical technique used in a wide range of engineering disciplines. It is known for its flexibility in handling complex geometries and boundary conditions. FEM is extensively used in various fields such as structural engineering, mechanical engineering, electrical engineering, aerospace engineering, civil engineering, biomedical engineering, thermal engineering, and fluid dynamics. It enables engineers to predict the behavior of structures, components, and systems under different loads and conditions. By dividing the domain into smaller elements, FEM approximates the solution to complex problems by solving a system of equations. This method has revolutionized engineering analysis and design, allowing engineers to optimize designs, improve performance, and ensure the safety and reliability of engineering systems.

- Structural Engineering: FEM is used to predict the behavior of buildings, bridges, towers, and other structures under various loads to ensure they can safely withstand the forces they will be subjected to.
- Mechanical Engineering: It's used in the design and analysis of mechanical components and systems, such as predicting the stress and strain in mechanical parts under load.
- Electrical Engineering: FEM is used in the design and analysis of electrical and electronic components, such as antennas, waveguides, and microelectronic devices.
- Aerospace Engineering: It's used to simulate and analyze the behavior of aircraft and spacecraft structures under various conditions.
- Civil Engineering: FEM is used in the analysis and design of civil infrastructure like dams, tunnels, and roads.
- Biomedical Engineering: It's used to model biological systems, such as the flow of blood in arteries, the mechanical behavior of bones and tissues, etc.



Et même
ZouLab !



A l'issue de ce cours, vous serez capables de...

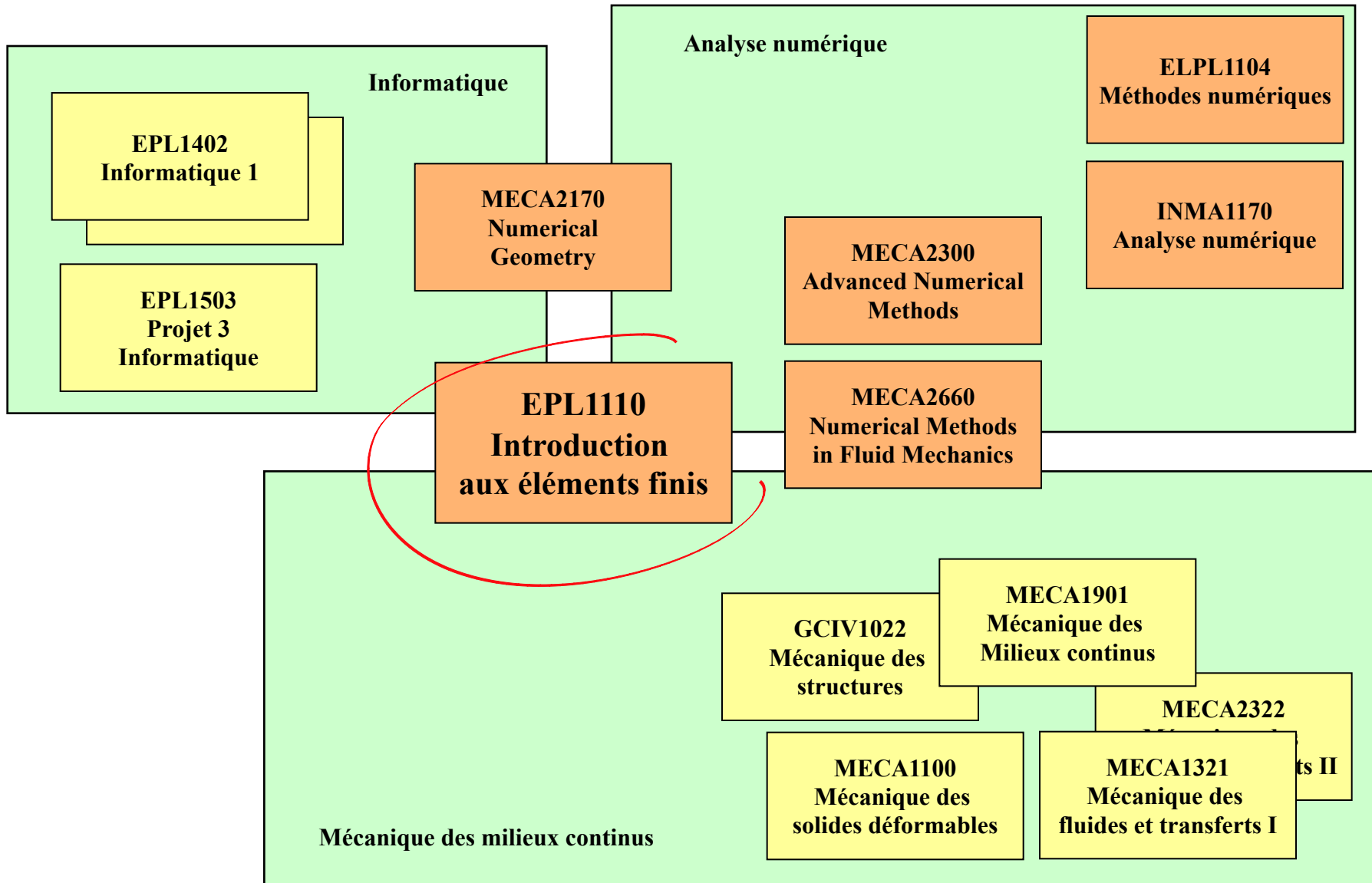
- Comprendre la méthode des éléments finis
- Réaliser un petit programme en C
- Certifier et valider une simulation
- Choisir la voie numérique la plus efficace
- Estimer la précision d'un résultat
- Découvrir les joies et les aléas du numérique
- Découvrir les joies de l'intelligence artificielle



Non, non : ceci on ne fera pas !

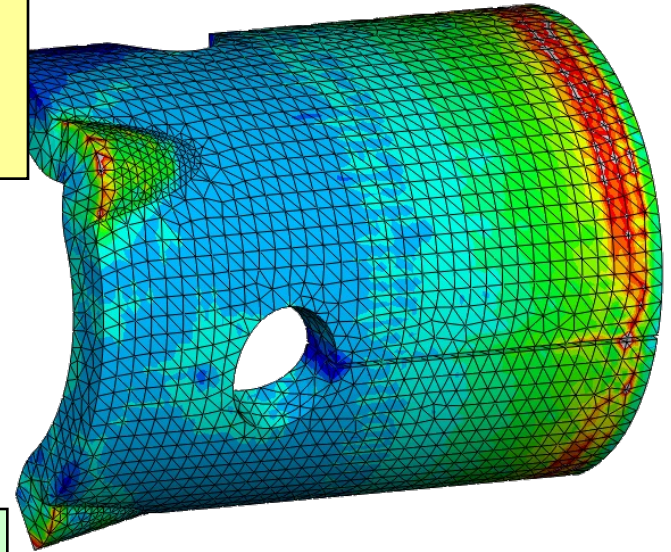
- Apprendre le génie logiciel de l'orienté-objet
- Utiliser des logiciels commerciaux
- Faire de l'analyse numérique théorique
- Faire du calcul parallèle
- Résoudre les équations de Navier-Stokes
- Créer automatiquement des maillages

Et les autres cours....



Plan du cours et évaluation

Comment intégrer
numériquement
une fonction
sur un carré ?



Evaluation continue

S1-S8 : 8 cours et 7 petits problèmes ***A***

Comment résoudre
l'élasticité linéaire ?

Evaluation certificative

S7-S9 : mini-projet ***B***

S9-S12 : 4 cours théorie éléments finis

En session : examen écrit ***C***

En juin, note finale $= (A+B+C)/3$ *si* $C > 10$
 $= C$ *sinon*

En septembre, examen écrit + pas de projet de rattrapage !

Evaluation

Elasticité linéaire pourquoi ?

Centrale nucléaire de Tihange 1

Rapport de synthèse
Long Term Operation 2015-2025
Version finale



Et la transition ? A vous de choisir quel problème simuler...

GREENPEACE

Bonjour Vincent,

Incroyable mais vrai : TotalEnergies ne figure plus parmi les sponsors des 20 km de Bruxelles.

Cela fait 2 ans que Greenpeace, Totalement Down, Youth for Climate, Amnesty International et Grand-Parents pour le climat organisent des actions et des pétitions pour demander à l'organisation des 20 km de Bruxelles de ne plus accepter le sponsoring de TotalEnergies.



Objectifs du projet

Réaliser	Créer une application pour prédire un tsunami.
Certifier	Tester et valider le travail de votre groupe.
Expliquer	Expliquer de manière efficace et rapide à l'enseignant et aux autres étudiants ce que vous avez réalisé.
Comprendre	Comprendre ce que vous avez réalisé. Comprendre ce que d'autres groupes ont réalisé.

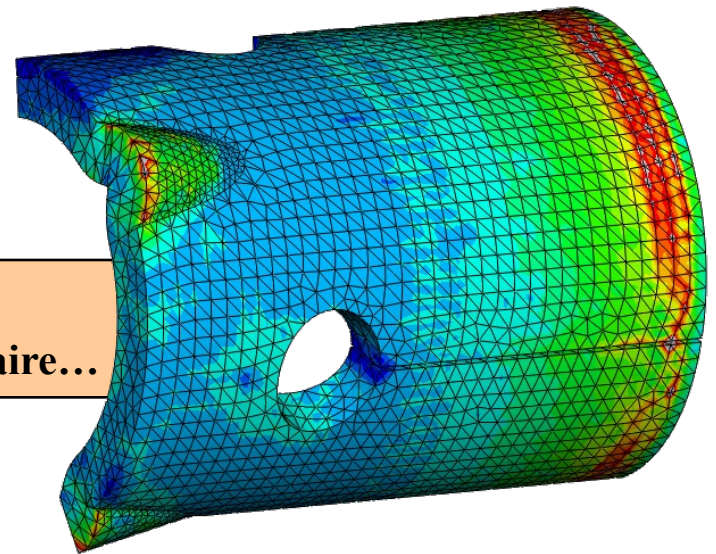
Exercices : 8 petits problèmes

Quelques petits problèmes
élémentaires pour apprivoiser le C

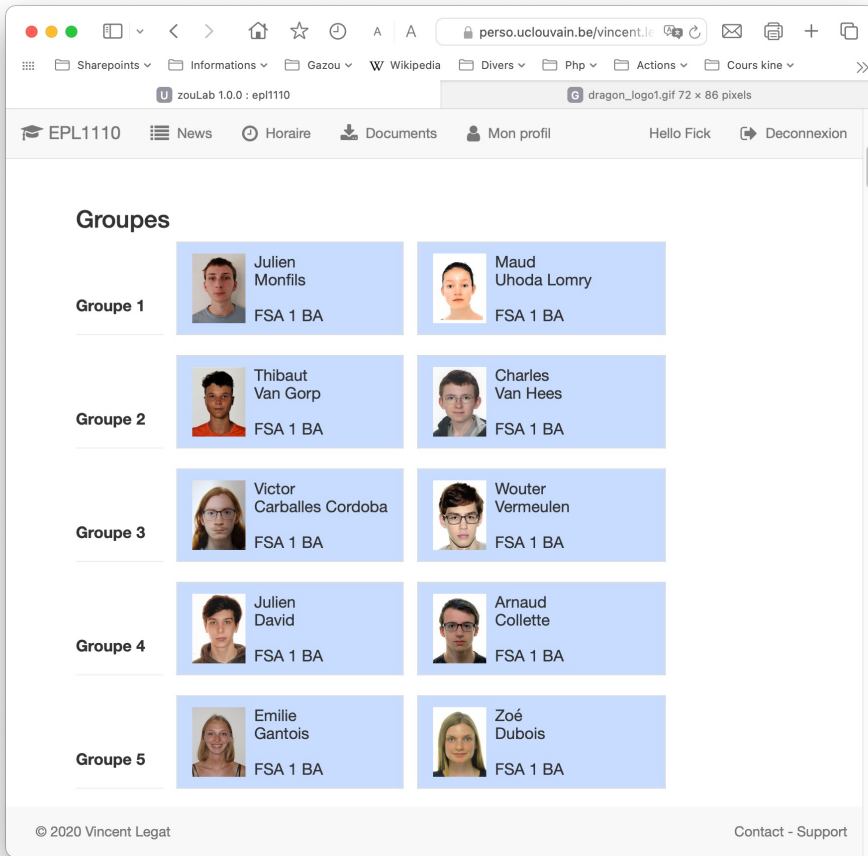
Projet en C :

Une petite application efficace pour l'élasticité linéaire...

The Practice of Programming :
Simplicity, Clarity, Generality. (B.K Kernighan & R. Pike 99)



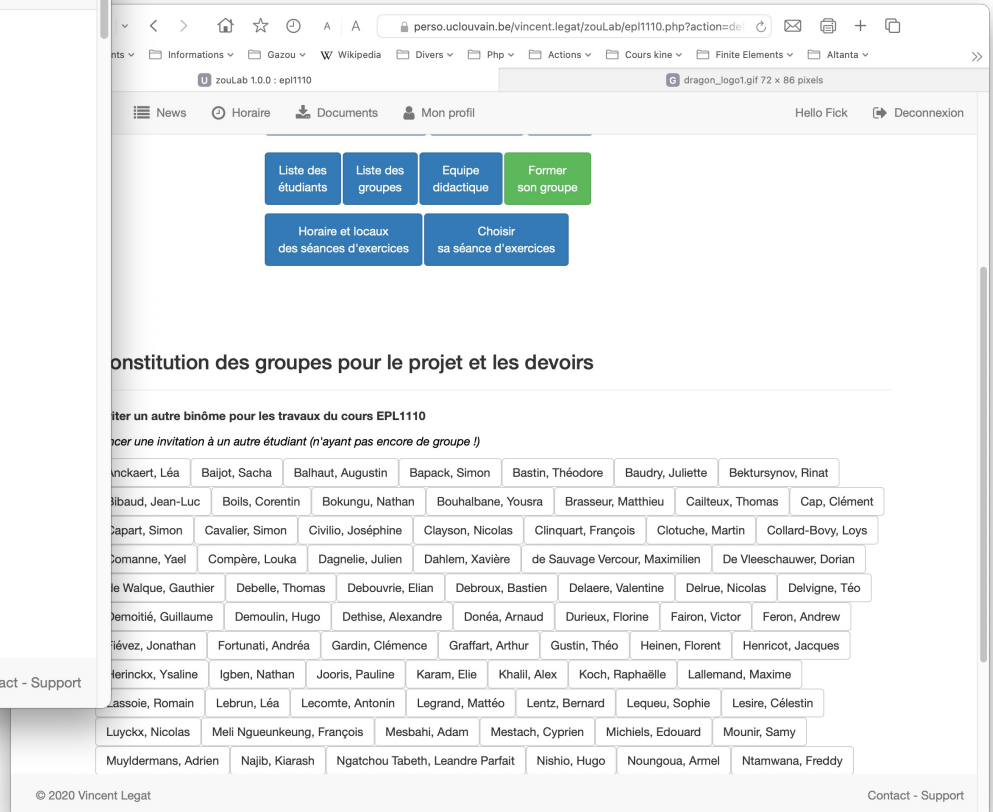
Former un groupe de deux étudiants (pas trois) !



The screenshot shows the 'Groupe 1' page on the EPL1110 website. The page displays a list of student profiles for the first group. The profiles are arranged in two columns. The first column contains the names and photos of Julien Monfils, Thibaut Van Gorp, Victor Carballes Cordoba, Julien David, and Emilie Gantois. The second column contains the names and photos of Maud Uhoda Lomry, Charles Van Hees, Wouter Vermeulen, Arnaud Collette, and Zoé Dubois. All students are listed as 'FSA 1 BA'.

© 2020 Vincent Legat

Contact - Support



The screenshot shows the 'Former son groupe' page on the EPL1110 website. The page displays a list of student profiles for the first group. The profiles are arranged in two columns. The first column contains the names and photos of Julien Monfils, Thibaut Van Gorp, Victor Carballes Cordoba, Julien David, and Emilie Gantois. The second column contains the names and photos of Maud Uhoda Lomry, Charles Van Hees, Wouter Vermeulen, Arnaud Collette, and Zoé Dubois. All students are listed as 'FSA 1 BA'.

© 2020 Vincent Legat

Contact - Support

Avoir une séance !

Avoir un tuteur et un assistant !



perso.uclouvain.be/vincent.legat/zi

zouLab 1.0.0 : epl1110

dragon_logo1.gif 72 x 86 pixels

EPL1110 News Horaire Documents Mon profil Hello Fick Deconnexion

Choix d'une des quatre séances d'exercices

Inscrit à la séance du mardi 14h00 : BARB03 - assistants : Nathan, Ange et Miguel

Vous pouvez choisir la séance de votre choix : le nombre d'étudiants à chaque séance de tutorat est strictement limité aux 100 premiers inscrits par séance (50 pour le BARB03).

Important : il faut choisir la même séance que votre binôme, car cela fixera votre tuteur et votre assistant de groupe...

Pour le moment, toutes les séances se font exclusivement en présentiel.

Séance du mardi 14h00 : BARB91 avec **Nathan, Ange et Miguel** : 99 étudiant(e)s déjà inscrit(e)s
Séance du mardi 14h00 : BARB03 avec **Nathan, Ange et Miguel** : 19 étudiant(e)s déjà inscrit(e)s
Séance du mercredi 10h45 : BARB92 avec **Alexandre, Michel et Thomas** : 78 étudiant(e)s déjà inscrit(e)s
Séance du jeudi 10h45 : BARB91 avec **Antoine, Simon et Christophe** : 49 étudiant(e)s déjà inscrit(e)s

Sélection d'un séance de tutorat :

Sélectionner un séance ci-dessus et cliquer sur le gros bouton ci-dessous ensuite :-)

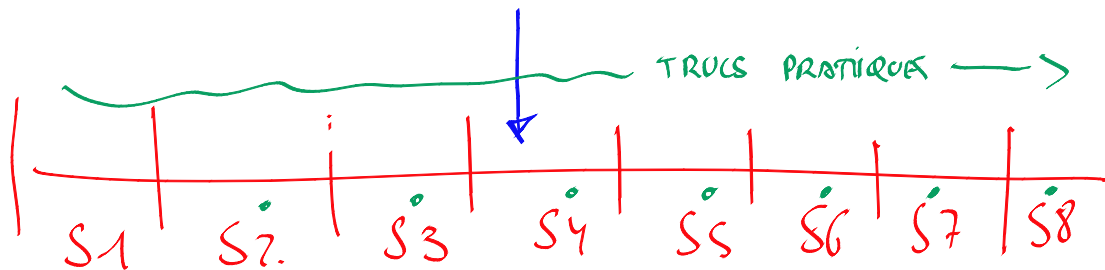
Vincent Legat

Contact - Support

Planning

MINI
PROPOSAL
↳ ASSISTANT

DEFINIR
LE PROBLEME
A RESOUDRE !



INTEGRATION

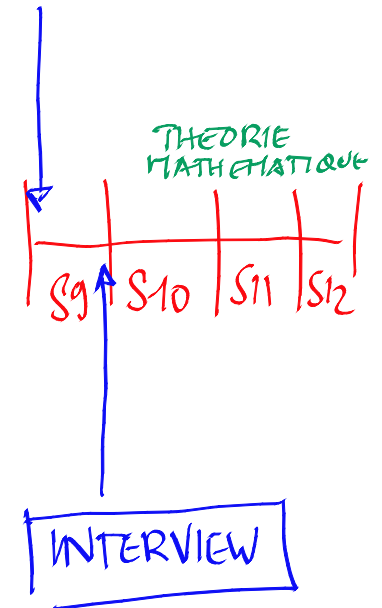
GMSH

TURNER

ENONCE
DU PROJET

REMISE
DU PROJET

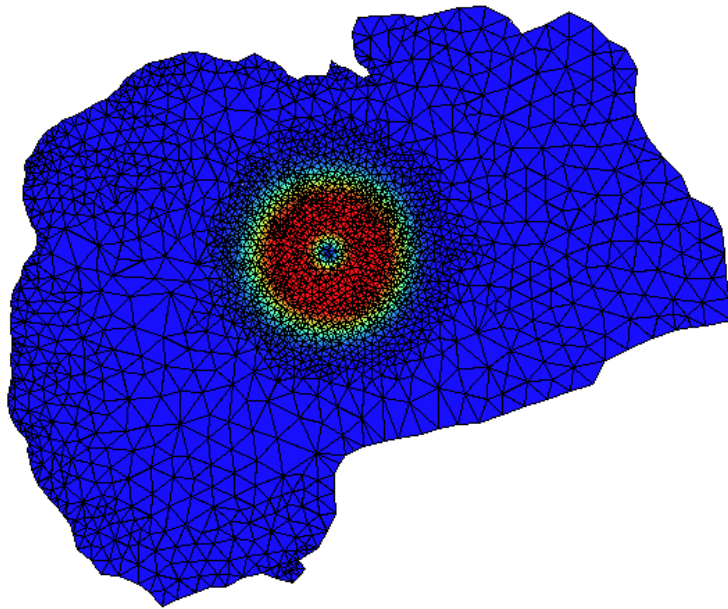
- CODE
- RAPPORT
D'UNE PAGE



THEORIE
MATHEMATIQUE

INTERVIEW

The Finite Element Method



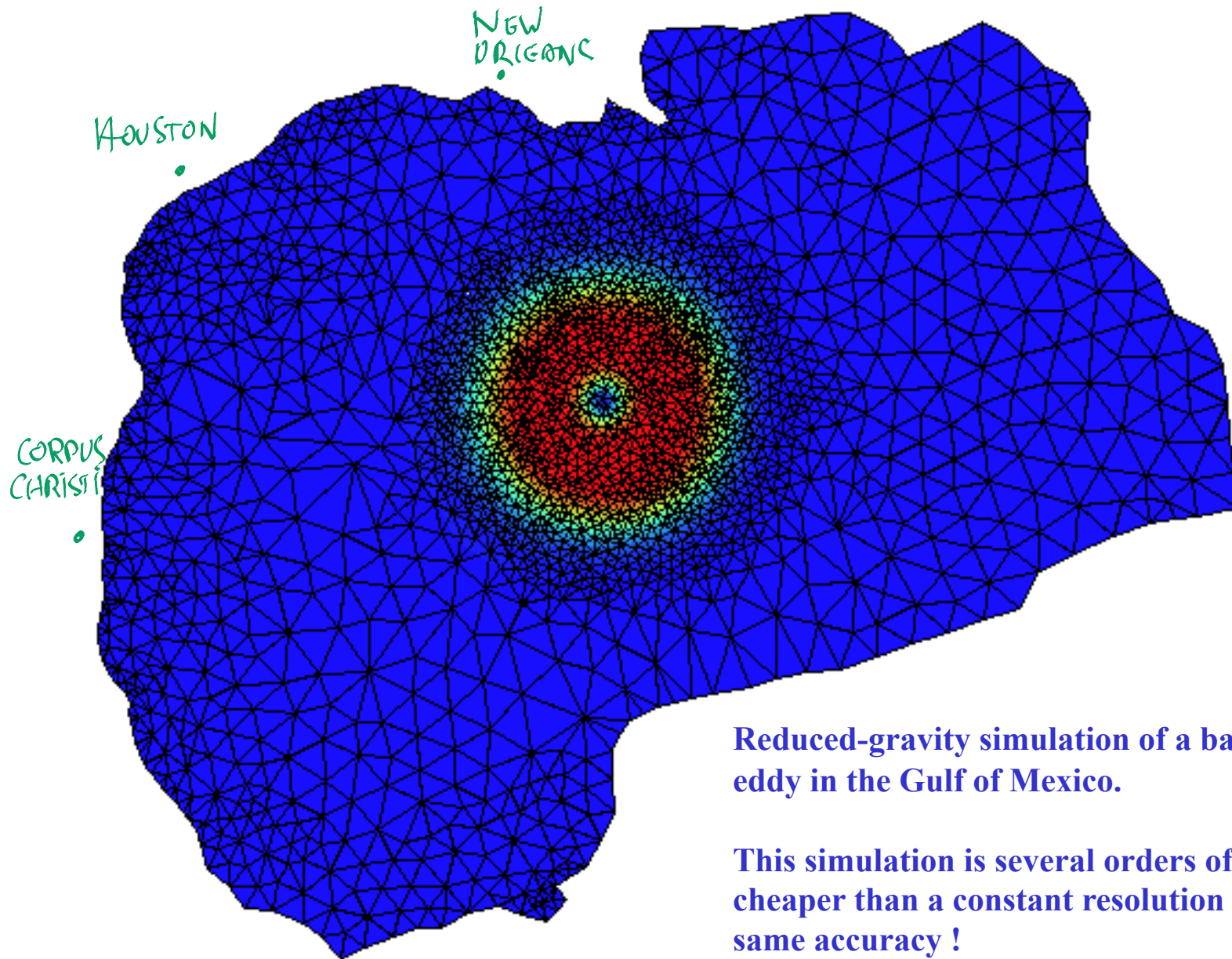
Typical applications

Deformable solids mechanics
Fluid dynamics (CFD)
Electromagnetism
Transport phenomena
Climatology

is a way of computing approximate solutions to a mathematical model describing a physical process.

What is a mathematical model ?
A boundary value problem.

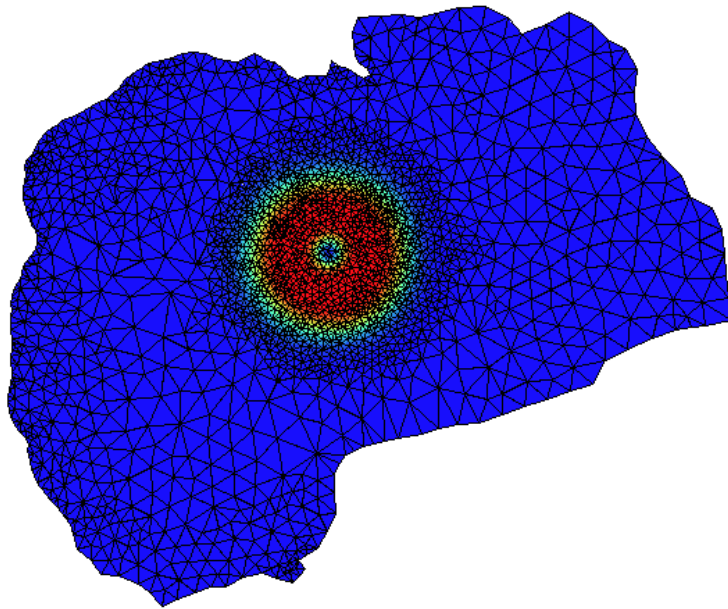
What is a boundary value problem ?
A set of partial differential equations with boundary and initial conditions.



Reduced-gravity simulation of a baroclinic eddy in the Gulf of Mexico.

This simulation is several orders of magnitude cheaper than a constant resolution one of the same accuracy !

The Finite Element Method



Typical applications

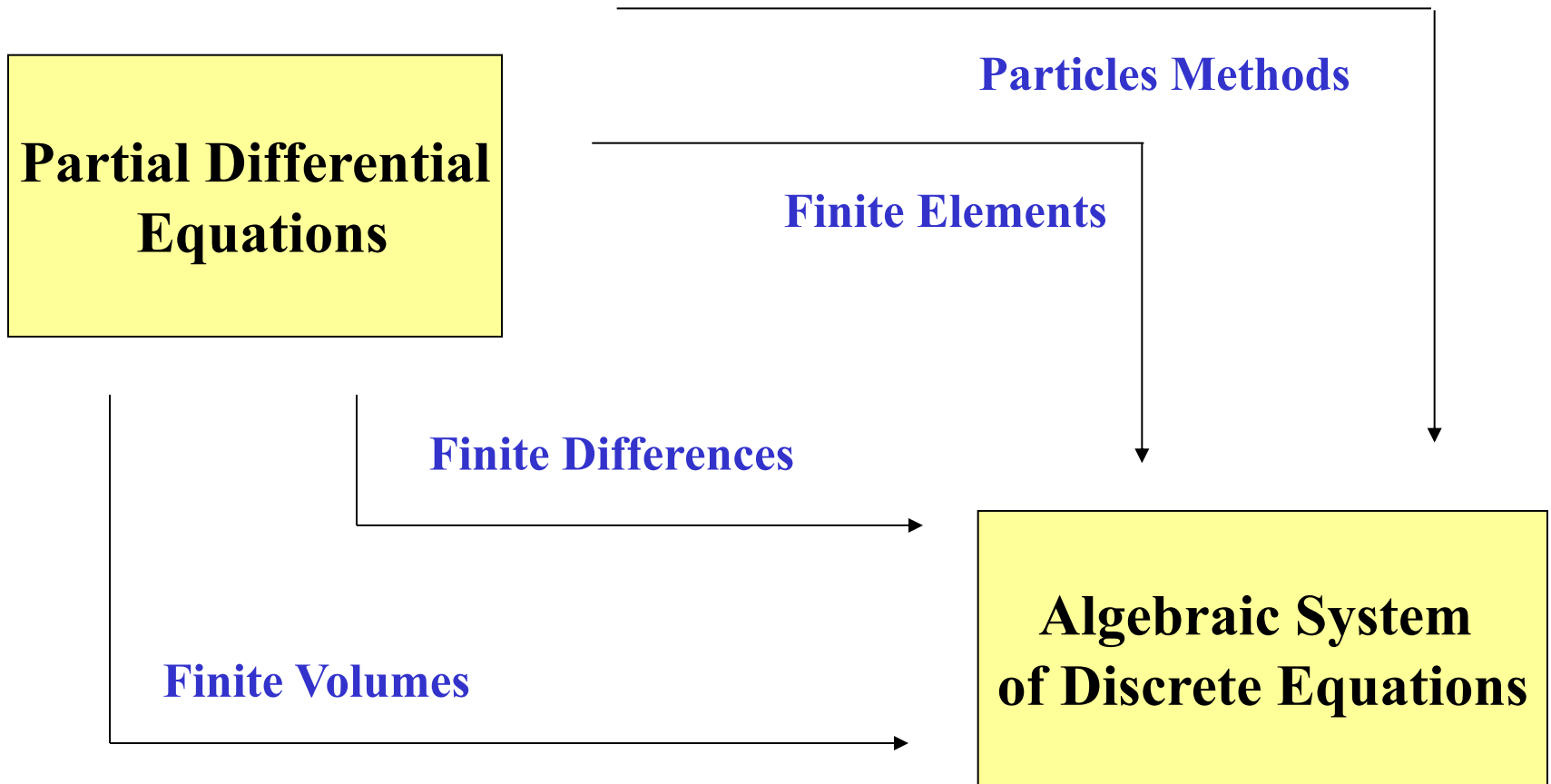
Deformable solids mechanics
Fluid dynamics (CFD)
Electromagnetism
Transport phenomena
Climatology

is a way of computing approximate solutions to a mathematical model describing a physical process.

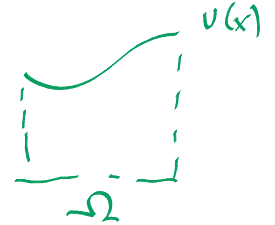
What is a mathematical model ?
A boundary value problem.

What is a boundary value problem ?
A set of partial differential equations with boundary and initial conditions.

Finite Elements, Finite Differences, Finite Volumes etc.



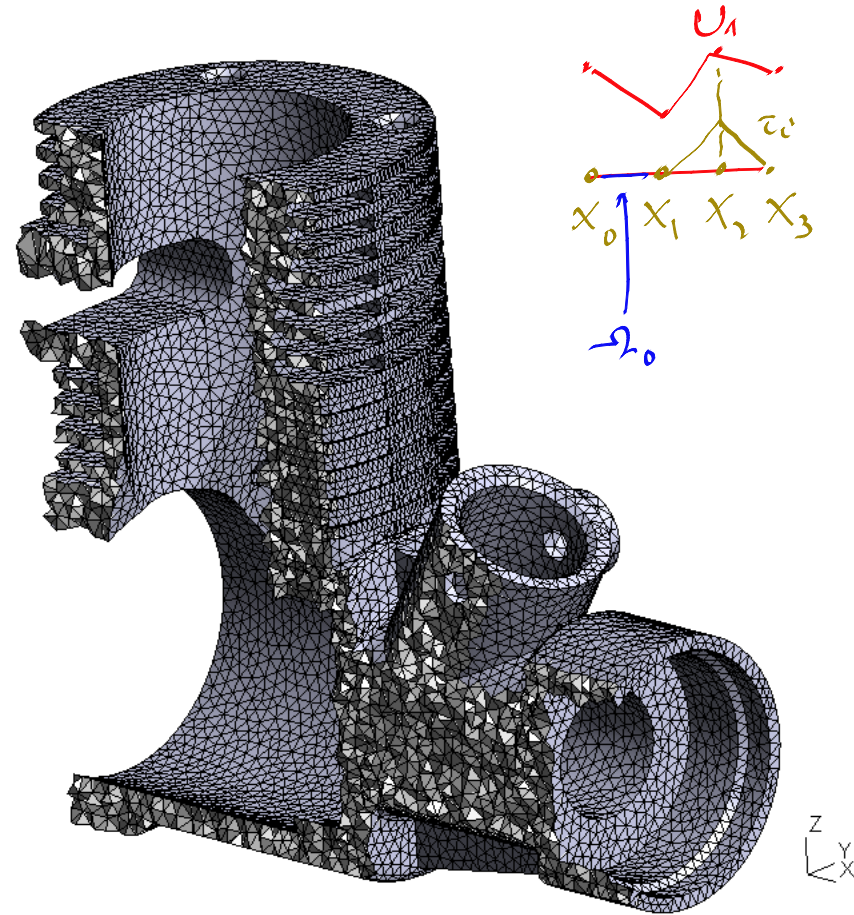
The Finite Elements Method is a discretization method



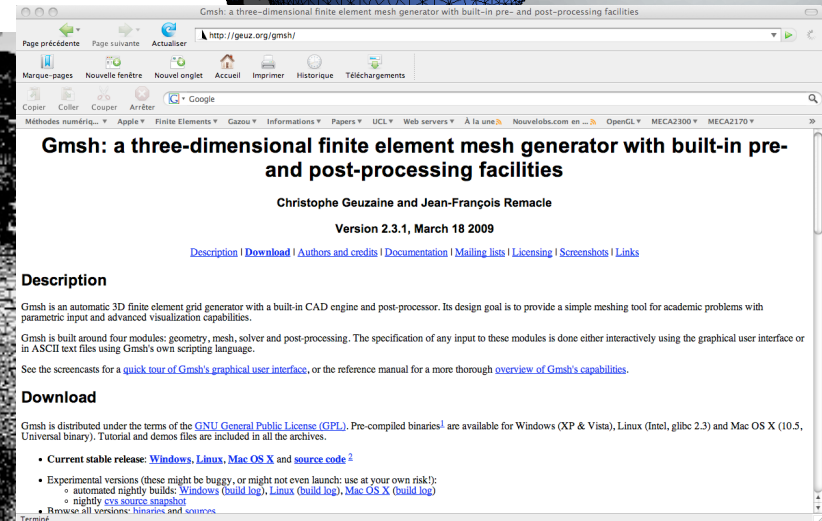
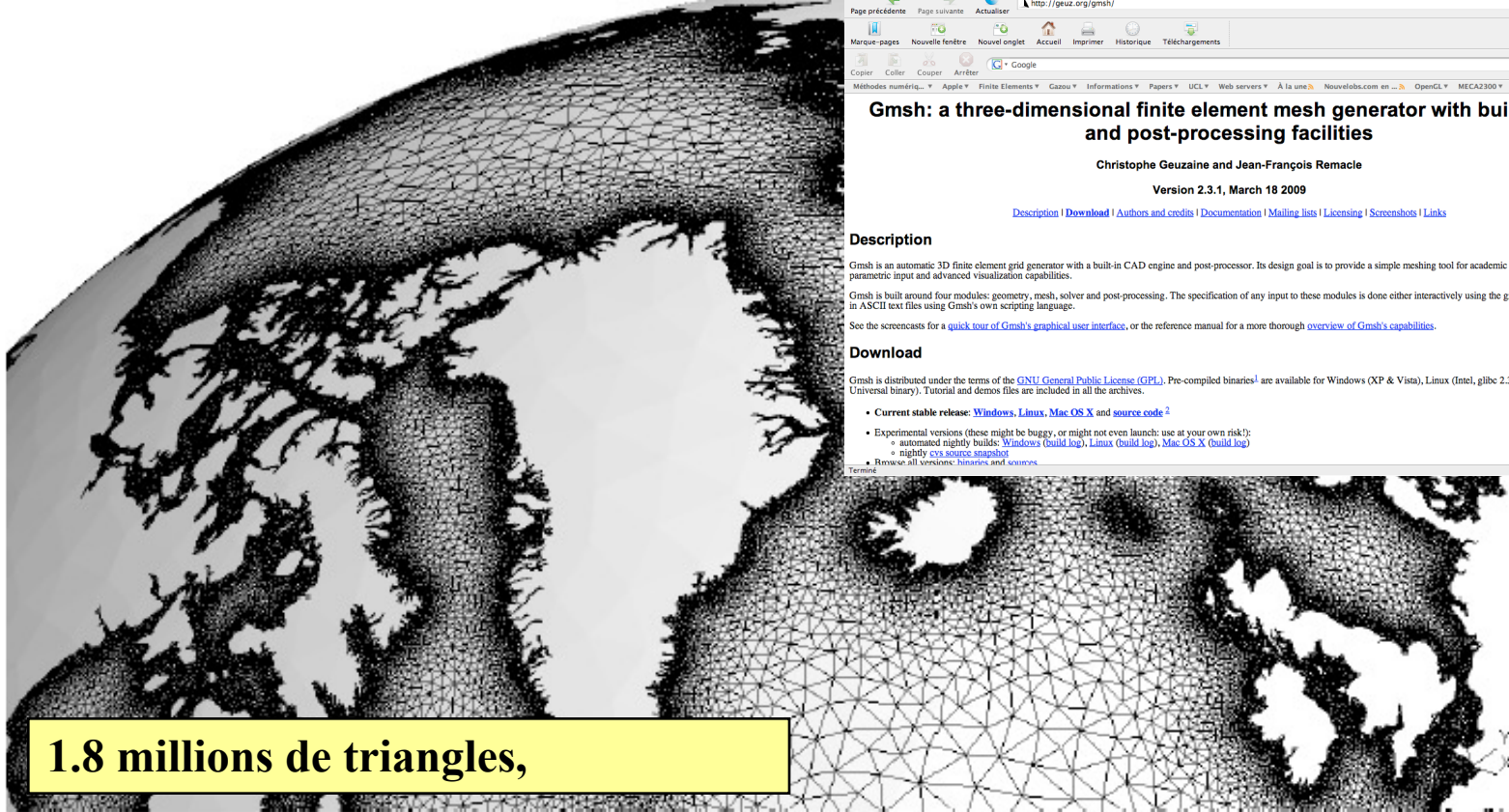
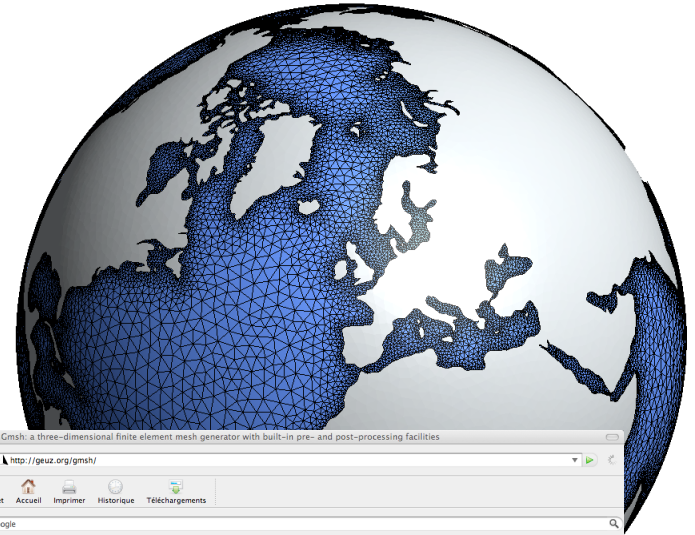
The problem geometry is divided in small finite elements.

On each element, the solution is approximated by means of unknown nodal values and given polynomials

$$u(\mathbf{x}) \approx u^h(\mathbf{x}) = \sum_{j=1}^n U_j \tau_j(\mathbf{x})$$

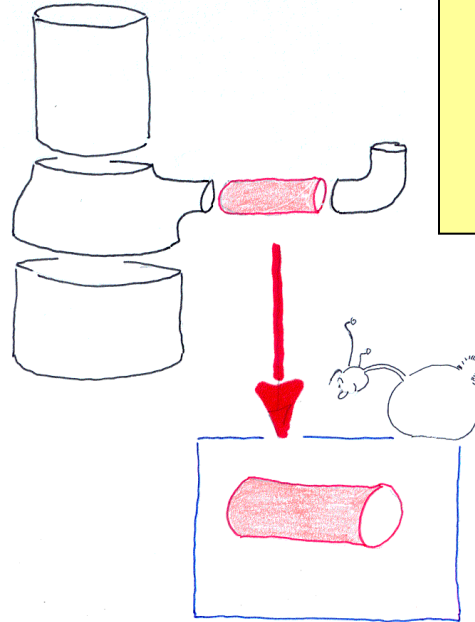


Second-generation Louvain-la-Neuve Ice-ocean Model



1.8 millions de triangles,

Classical Engineering Analysis



**Exact solution
to approximate problems**

**Analysis through simple geometries
and a limited combination of
approximate models :**

- Lubrication theory**
- Bars**
- Beams**
- Plates and shells**

Low computer's cost
Good physical understanding

Simplicity of models

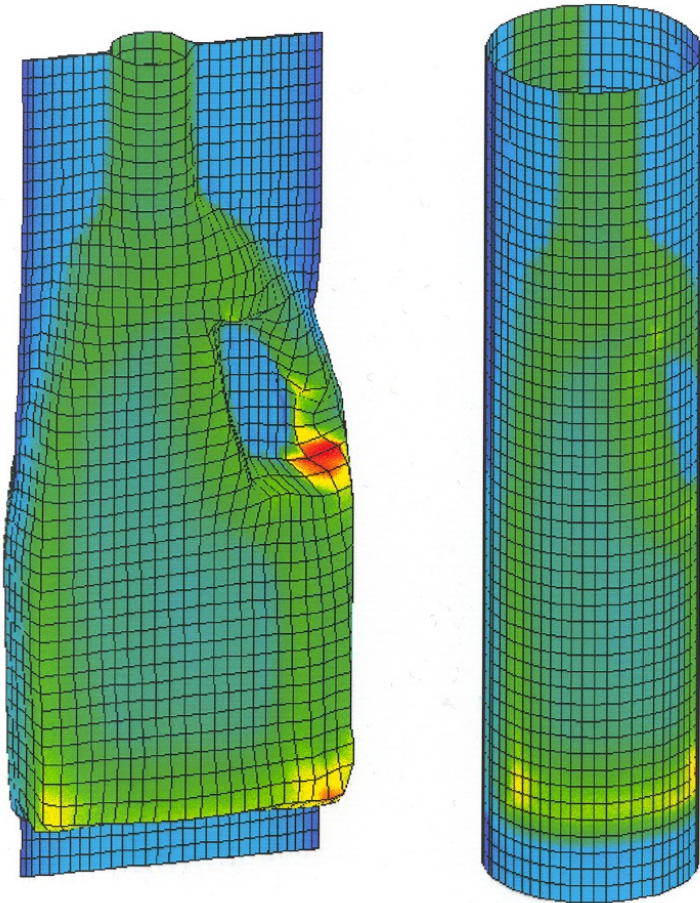
Complex geometries and loads cannot be handled

Complex materials cannot be analyzed

Computer Aided Engineering Analysis

The Finite Element Method provides approximate solutions to more realistic problems

FEM developed in the sixties for linear elasticity and generalized to many other applications...



Powerful and flexible

High (cheap) computer's cost

Low (expensive) engineer's cost

Complex processes can be analyzed

Complex material laws can be included

However...

Garbage in



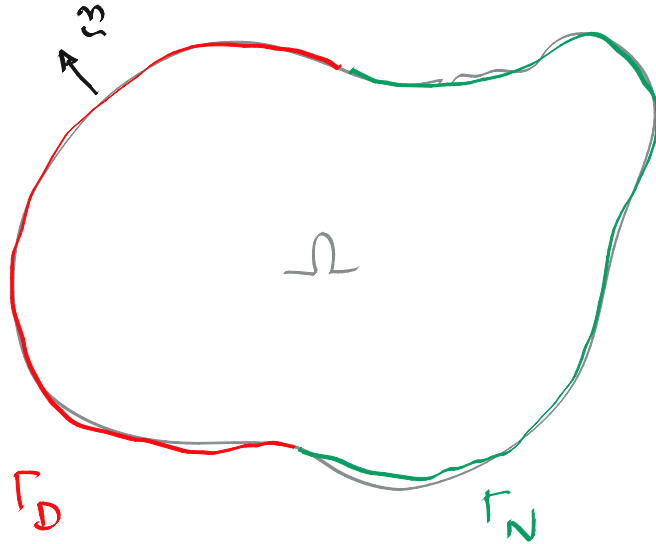
Illusion of non-qualified users to be able to analyze everything

New modeling issues requires higher qualifications...

Garbage out

How to define complex problems in an accurate and efficient way for the computer software ?

Typical elliptic boundary value problem



$$\Delta \cdot \nabla u + f = 0 \quad \text{sur } \Omega$$

$$\begin{aligned} \Delta \cdot \nabla u &= g & \text{sur } \Gamma_N \\ u &= t & \text{sur } \Gamma_D \end{aligned}$$

$$\partial\Omega = \Gamma_D \cup \Gamma_N$$

CONDITIONS
ESSENTIELLES

DIRICHLET

CONDITIONS
NATURELLE

NEUMANN

Some nice spaces and notations...

$$\mathcal{U} = \{ v \dots \text{tel que } v = t \text{ sur } \Gamma_0 \}$$

$$\hat{\mathcal{U}} = \{ \hat{v} \dots \text{tel que } v = 0 \text{ sur } \Gamma_0 \}$$

ESPACE DES CANDIDATS SOLUTION

ESPACE DES FONCTIONS TEST

$$\langle f g \rangle = \int_{\Omega} f g \, d\Omega$$

$$\ll f g \gg = \int_{\partial\Omega} f g \, d\sigma$$

$$\ll f g \gg_N = \int_{\Gamma_N} f g \, d\sigma$$

... to do calculus !

Three tips !

$$\langle \hat{v} (\nabla_x \cdot \nabla_x v + f) \rangle = 0 \quad \forall \hat{v} \in \hat{U}$$

[TIP 1]

$$(fg)' = fg' + f'g$$

$$\langle \hat{v} \nabla_x \cdot \nabla_x v \rangle = \langle \nabla_x \cdot (\hat{v} \nabla_x v) \rangle - \langle (\nabla_x \hat{v}) \cdot (\nabla_x v) \rangle$$

$\hat{v} \in \hat{U} \quad v \in U$

[TIP 2] $\langle \nabla_x \cdot a \rangle = \langle \eta \cdot a \rangle$

$$\langle \langle \eta \cdot \nabla_x v \rangle \hat{v} \rangle$$

[TIP 3] $\hat{v} = 0 \text{ sur } \Gamma_D$

$$\langle \underbrace{(\eta \cdot \nabla_x v)}_g \hat{v} \rangle_\Gamma$$

And it is all !

Strong formulation



? u tel que

$$\nabla_x \cdot \nabla_x u + f = 0 \quad \text{sur } \Omega$$

$$m \cdot \nabla_x u = g \quad \text{sur } \Gamma_N$$

$$u = t \quad \text{sur } \Gamma_D$$

FORMULATION
FORTE

FORMULATION
FAIBLE

? $u \in U$

$$\underbrace{\langle \nabla_x \hat{u}, \nabla_x u \rangle}_{a(\hat{u}, u)} = \underbrace{\langle\langle g, \hat{u} \rangle\rangle_N}_{b(\hat{u})} + \underbrace{\langle f, \hat{u} \rangle}_{c(\hat{u})} \quad \forall \hat{u} \in \hat{U}$$



$a(\hat{u}, u)$

$b(\hat{u})$

$$? u \in U \quad J(u) = \min_{v \in U} \left(\underbrace{\frac{1}{2} a(v, v)}_{J(v)} - \underbrace{b(v)}_{J(v)} \right)$$

PROBLEME DE MINIMUM



Weak formulation