Uma introdução Teórica e Numérica em alguns problemas clássicos de EDP's (PARTE 1)

Prof.: Pitágoras Pinheiro de Carvalho - **(UESPI)**

(Escola de Verão 2021 - UFPB)

22 de fevereiro de 2021

Programa de Pós Graduação em
MATEMÁTICA

Sumário

- Genericamente falando, uma equação diferencial parcial é uma equação que envolve uma função desconhecida $u(x_1, ..., x_n)$ e suas derivadas parciais de até uma certa ordem.

- A ordem de uma equação diferencial parcial é a ordem da derivada mais alta que aparece na mesma. Por exemplo, uma equação diferencial de primeira ordem nas variáveis (x, y) é da forma

$$\boxed{\mathbf{F(x,\ y,\ u, u_x, u_y) = 0,}}$$

enquanto uma equação diferencial parcial de segunda ordem nas variáveis (x, y) é da forma

$$\boxed{\mathbf{F(x,\ y,\ u, u_x, u_y, u_{xy}, u_{xx}, u_{yy}) = 0.}}$$

- Como surgem as EDP's? As equações diferenciais parciais vem de problemas de modelagem em ciências (Física, Biologia, etc). Alguns exemplos:
- Equação de Burger com viscosidade:

$$\boxed{u_t + uu_x = \nu u_{xx},}$$

que aparece em várias formas de modelagens matemáticas, dentre elas: Mecânica de fluidos, acústica não linear, dinâmica de gases, entre outras...

- Equação de Laplace

$$\boxed{\Delta u = 0,}$$

onde Δ é o operador $\Delta u = \sum_{i=1}^{n} \frac{\partial^2 u}{\partial x_i^2}$.

Aparecem frequentemente na astronomia, no eletromagnetismo, na mecânica dos fluidos entre outras.

- Equação do calor:

$$\boxed{u_t + \Delta u = f,}$$

alguns exemplos: estudo de processos de **difusão** química, na estatística o estudo do movimento browniano através da equação de Fokker–Planck, dentre outras.

  - A equação de difusão, é uma versão mais geral da equação do calor.

- Equação da onda:

$$\boxed{u_{tt} + \Delta u = f,}$$

alguns exemplos: ondas sonoras, luminosas, eletromagnética ou aquáticas. Surgem em áreas como a óptica, acústica, eletromagnetismo, e dinâmica dos fluidos.

Solução Clássica × Solução Fraca

Clássica: Utiliza ferramentas fundamentais do Cálculo Diferencial e proporciona, quando possível, funções que são soluções em um sentido habitual. Em outras palavras, funções com derivadas parcias de ordem adequada que satisfazem o problema, "pontualmente", en todos pontos de uma região apropriada.

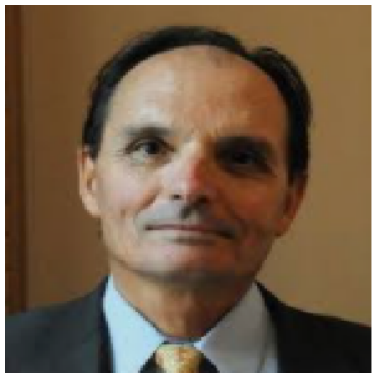
Fraca: Generaliza o conceito de função, aparecendo as distribuições; em seguida, aceitamos que uma solução é uma distribuição que verifica identidades adequadas, donde conseguimos (entre outras vantagens) baixar a ordem das derivadas. É comum dizer que estamos tratando de soluções fracas. Se este processo é realizado de maneira correta, consegue-se resolver um número muito maior de problemas ligados a EDPs.

A programação e simulação de EDP's

- Apresentar alguma visualização em problemas abstratos;
- Simulação de problemas reais em diversos campos: Física, Química, Engenharia, Medicina, Aviação, dentre vários outros;
- Problemas que não apresentam uma solução exata, podem ser resolvidos numericamente por aproximação;
- As simulações nos permitem buscar melhores caminhos antes de uma tomada de decisão.

Freefem++

- **Free**Fem++ é um "software usado para resolver numericamente Equações Diferenciais Parciais em $\mathbb{R}^2$, $\mathbb{R}^3$ ou em uma superfície, usando o **M**étodo dos **E**lementos **F**initos (**FEM**)."
- **Lembrete**: O MEF subdivide o domínio de um problema em partes menores, denominadas elementos finitos.
- A primeira versão foi criada em 1987 (por Olivier Pironneau) e chamava-se MacFem (funcionava apenas no Macintosh) escrita em Pascal. A versão do **Freefem++** usada hoje foi escrita em 1998 (F. Hecht, O. Pironneau, K.Ohtsuka.)




O software:

- Versão Freefem++: sobre essa versão, ver https://freefem.org/
  O programa é executado em sistemas Linux , Solaris , macOS e MS Windows . **FreeFem++** é um software gratuito .
- Versão Freefem++-cs: sobre essa versão, ver https://www.ljll.math.upmc.fr/lehyaric/ffcs/index.htm
- A linguagem FreeFEM é uma linguagem **C++**, com algo que é mais parecido com LaTeX.

Em resumo :

- FREE = LIVRE
- FEM = Método dos Elementos Finitos
- ++ = C++ como linguagem de programação.

O software (Novidades:)

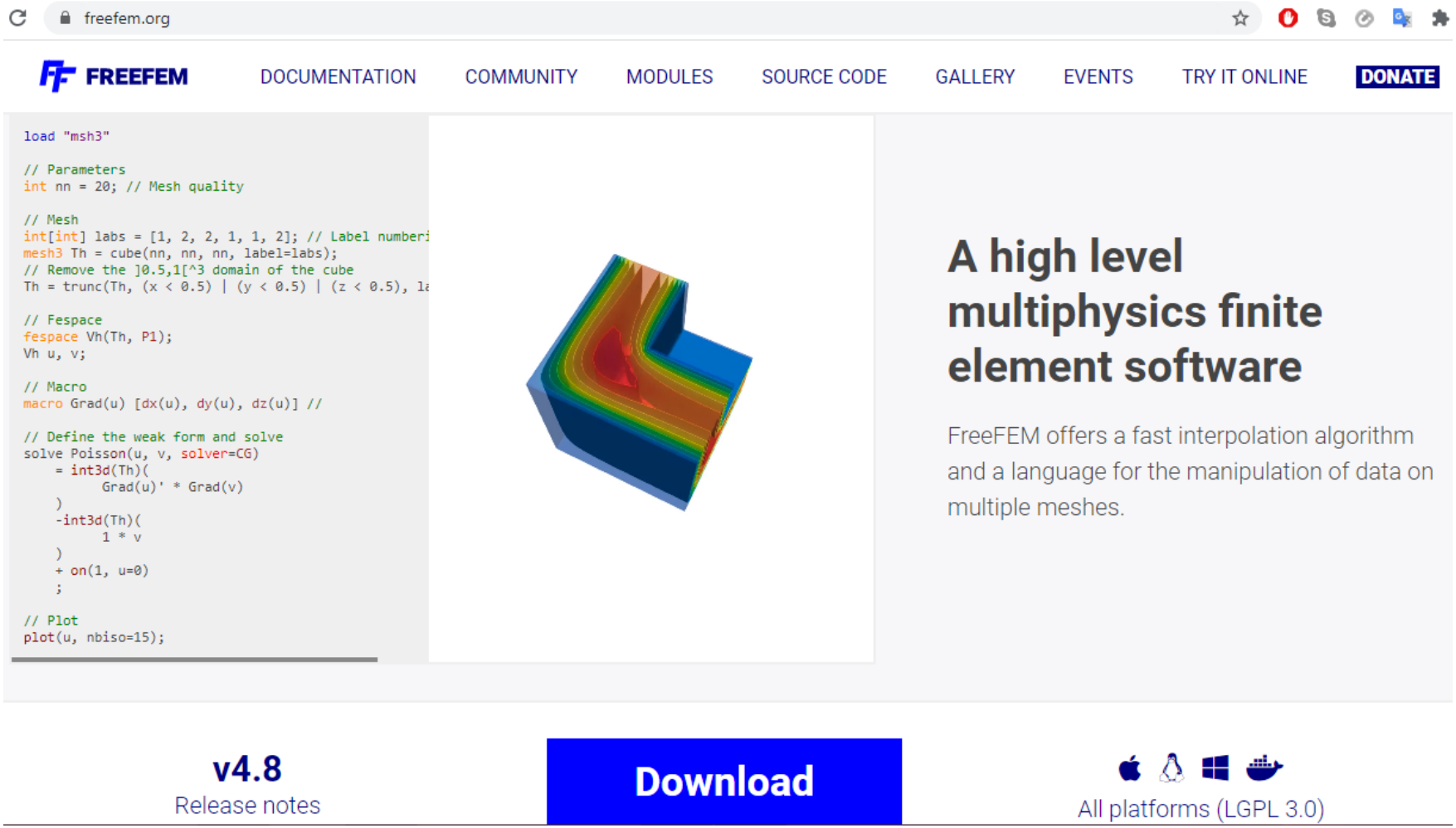
- Versão Javascript: (**Freefem++-js**) sobre essa versão, ver https://www.ljll.math.upmc.fr/lehyaric/ffjs/
  FreeFem++-js é um software gratuito .

- A versão Javascript do **FreeFem++** funciona diretamente a partir de uma página HTML.

- Qual é a configuração necessária do computador ?
  - **FreeFem++-js** funciona em qualquer computador ou smartphone.
  - Ele roda no navegador da Internet.
  - Nenhum download ou instalação é necessário.
- Como executar scripts **FreeFem++-js** quando desconectado da internet?
  - Salvar uma página HTML contendo **FreeFem++-js** em um disco local é o suficiente para torná-la disponível offline.

Instalação do FF++

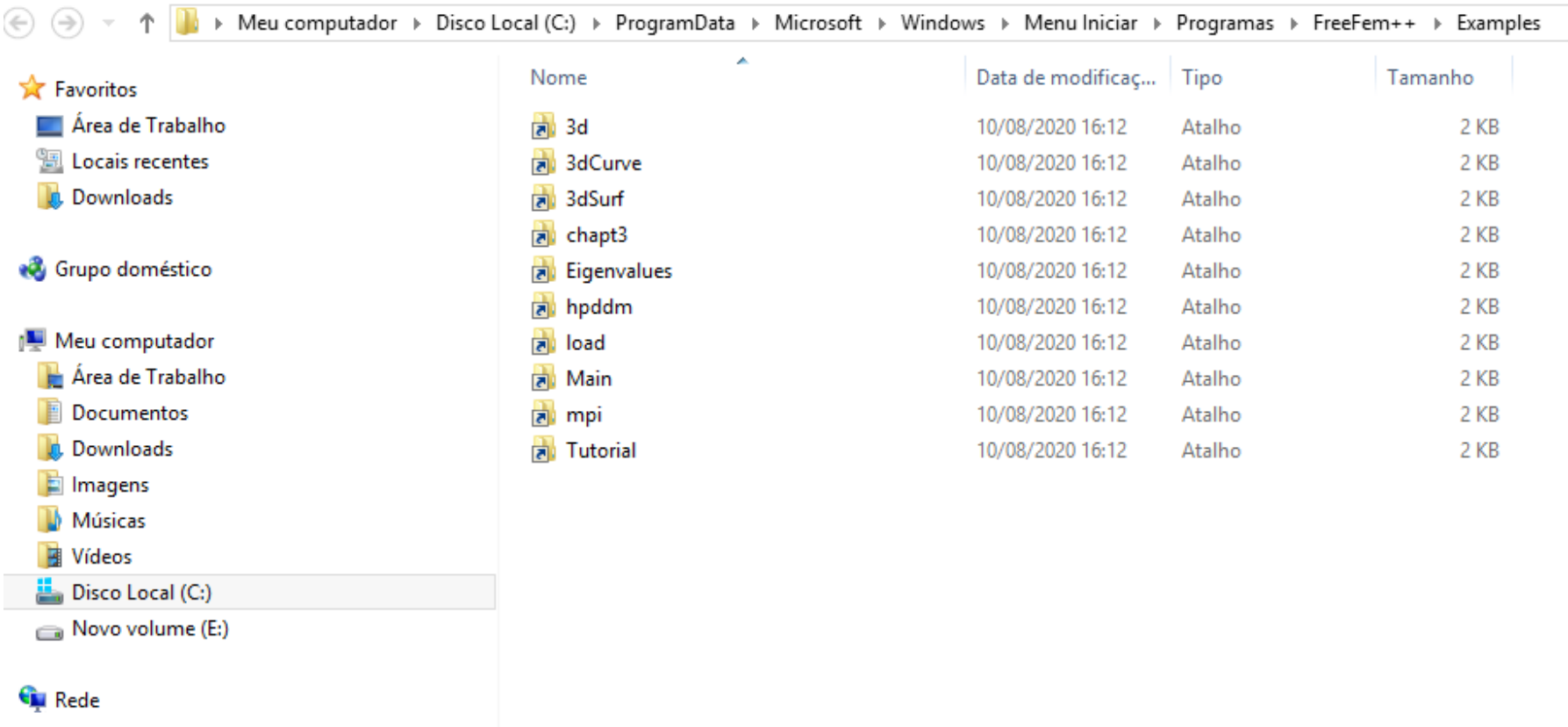
- Freefem++: Clique em: https://freefem.org/



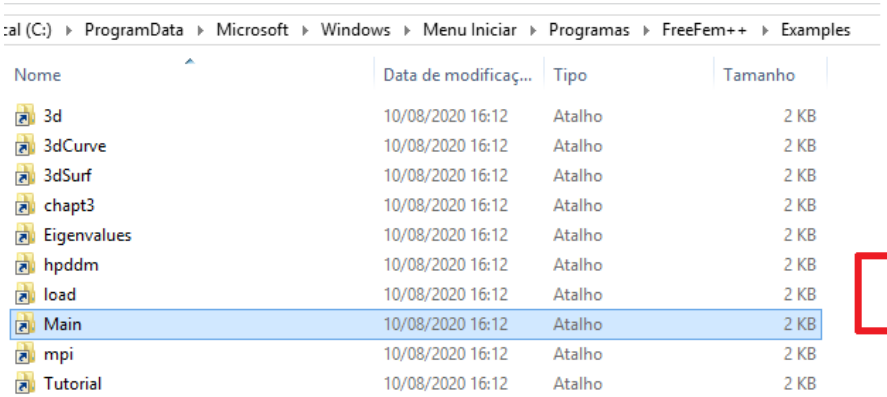
Identificando o Freefem++ :

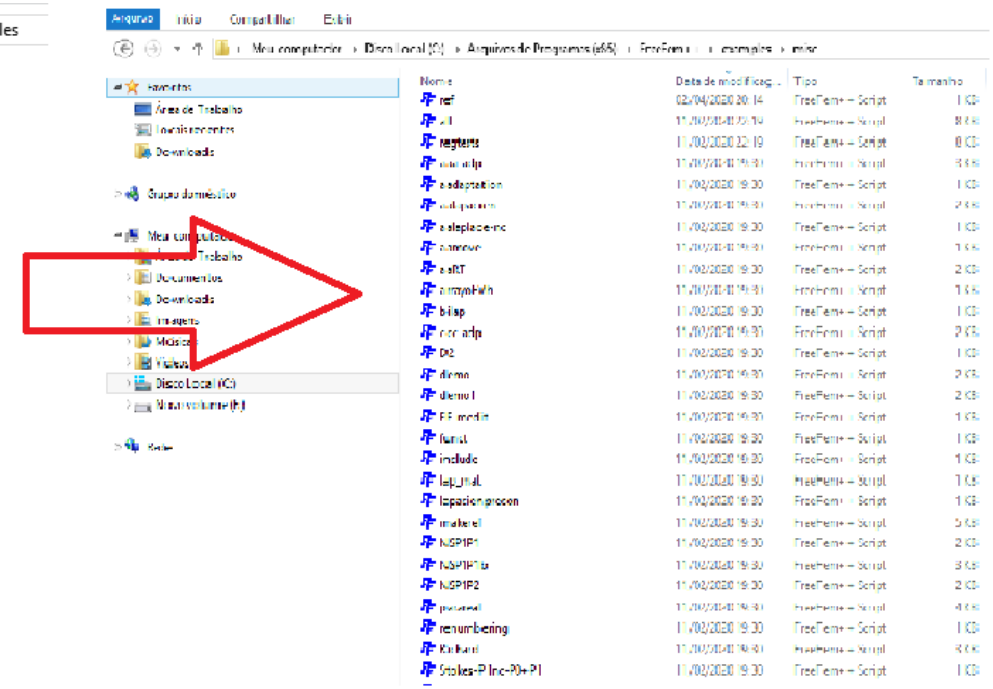
- Passo 1:

- Passo 2:

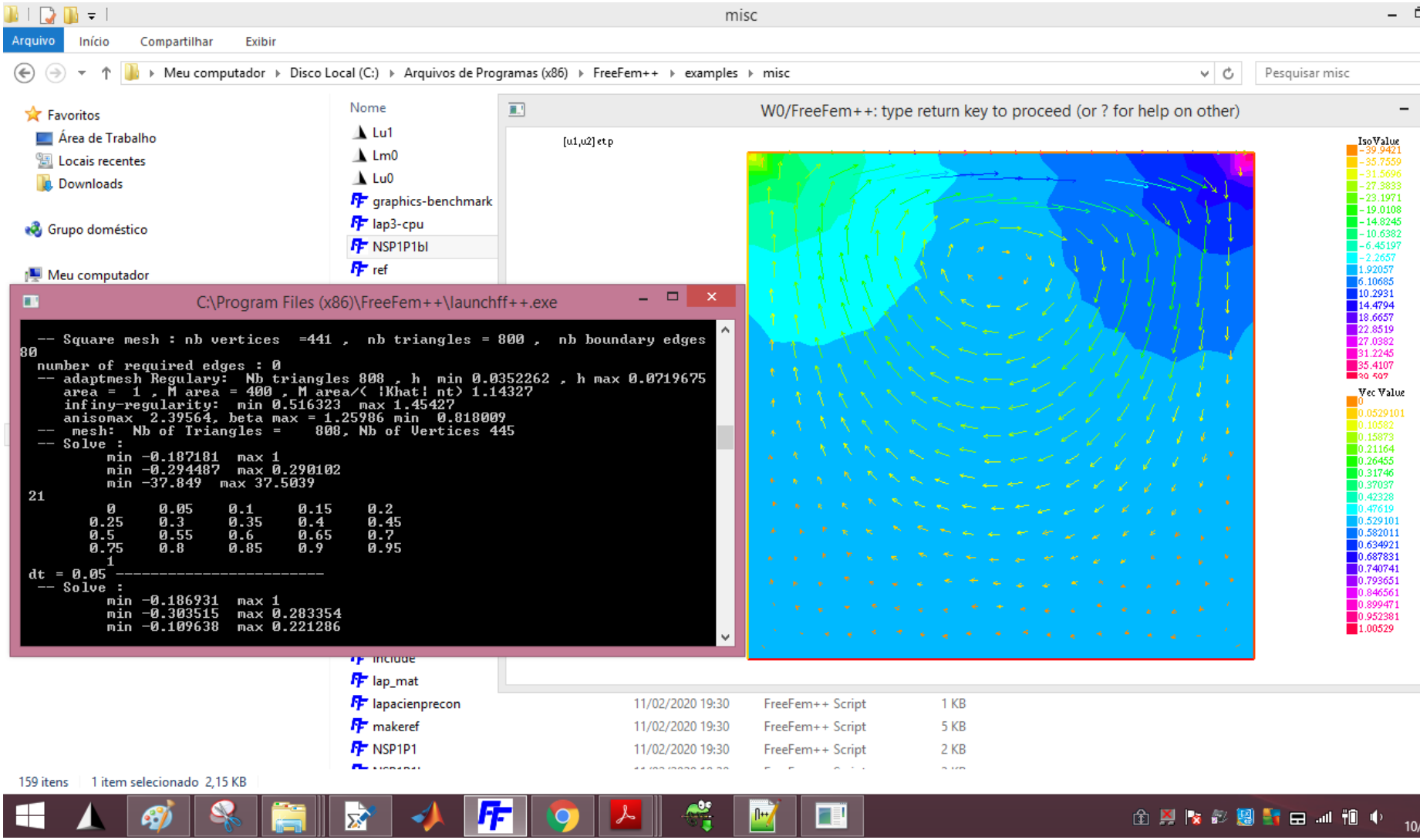

Meu computador ▸ Disco Local (C:) ▸ ProgramData ▸ Microsoft ▸ Windows ▸ Menu Iniciar ▸ Programas ▸ FreeFem++ ▸ Examples
Favoritos
Área de Trabalho
Locais recentes
Downloads
Grupo doméstico
Meu computador
Área de Trabalho
Documentos
Downloads
Imagens
Músicas
Vídeos
Disco Local (C:)
Novo volume (E:)
Rede
Nome
Data de modificaç...
Tipo
Tamanho
3d 10/08/2020 16:12 Atalho 2 KB
3dCurve 10/08/2020 16:12 Atalho 2 KB
3dSurf 10/08/2020 16:12 Atalho 2 KB
chapt3 10/08/2020 16:12 Atalho 2 KB
Eigenvalues 10/08/2020 16:12 Atalho 2 KB
hpddm 10/08/2020 16:12 Atalho 2 KB
load 10/08/2020 16:12 Atalho 2 KB
Main 10/08/2020 16:12 Atalho 2 KB
mpi 10/08/2020 16:12 Atalho 2 KB
Tutorial 10/08/2020 16:12 Atalho 2 KB

- Passo 3:


cal (C:) ▸ ProgramData ▸ Microsoft ▸ Windows ▸ Menu Iniciar ▸ Programas ▸ FreeFem++ ▸ Examples
Nome
Data de modificaç...
Tipo
Tamanho
3d 10/08/2020 16:12 Atalho 2 KB
3dCurve 10/08/2020 16:12 Atalho 2 KB
3dSurf 10/08/2020 16:12 Atalho 2 KB
chapt3 10/08/2020 16:12 Atalho 2 KB
Eigenvalues 10/08/2020 16:12 Atalho 2 KB
hpddm 10/08/2020 16:12 Atalho 2 KB
load 10/08/2020 16:12 Atalho 2 KB
Main 10/08/2020 16:12 Atalho 2 KB
mpi 10/08/2020 16:12 Atalho 2 KB
Tutorial 10/08/2020 16:12 Atalho 2 KB

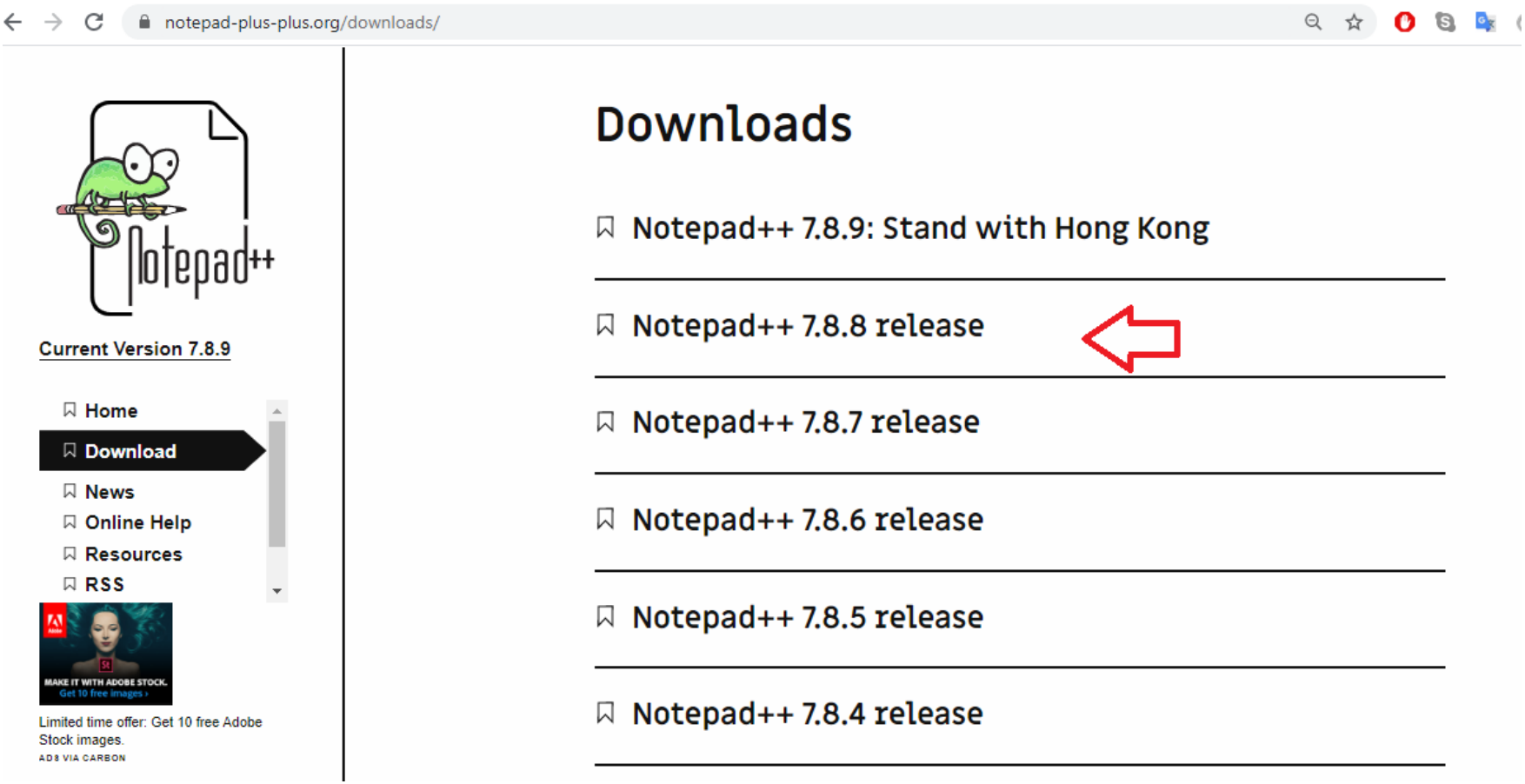


- Passo 4:

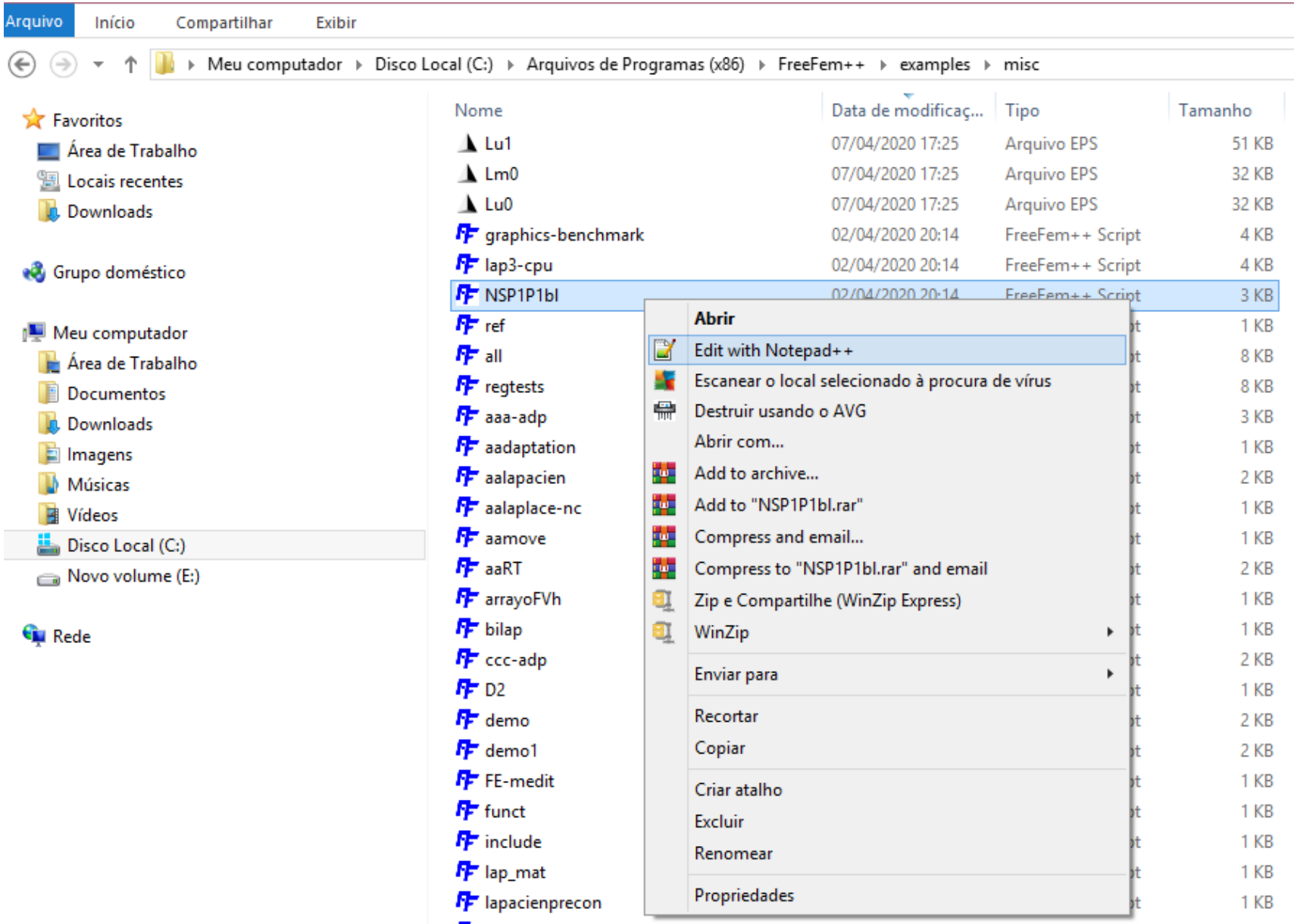

misc
Arquivo
Início
Compartilhar
Exibir
Meu computador ▸ Disco Local (C:) ▸ Arquivos de Programas (x86) ▸ FreeFem++ ▸ examples ▸ misc
Pesquisar misc
Favoritos
Área de Trabalho
Locais recentes
Downloads
Grupo doméstico
Meu computador
Nome
Lu1
Lm0
Lu0
graphics-benchmark
lap3-cpu
NSP1P1bl
ref
W0/FreeFem++: type return key to proceed (or ? for help on other)
[u1,u2] etp
IsoValue
Vec Value
C:\Program Files (x86)\FreeFem++\launchff++.exe
-- Square mesh : nb vertices =441 , nb triangles = 800 , nb boundary edges 80
number of required edges : 0
-- adaptmesh Regulary: Nb triangles 808 , h min 0.0352262 , h max 0.0719675
area = 1 , M area = 400 , M area/(!Khat! nt) 1.14327
infiny-regularity: min 0.516323 max 1.45427
anisomax 2.39564, beta max = 1.25986 min 0.818009
-- mesh: Nb of Triangles = 808, Nb of Vertices 445
-- Solve :
min -0.187181 max 1
min -0.294487 max 0.290102
min -37.849 max 37.5039
21
0 0.05 0.1 0.15 0.2
0.25 0.3 0.35 0.4 0.45
0.5 0.55 0.6 0.65 0.7
0.75 0.8 0.85 0.9 0.95
1
dt = 0.05 ----------
-- Solve :
min -0.186931 max 1
min -0.303515 max 0.283354
min -0.109638 max 0.221286
lap_mat
lapacienprecon 11/02/2020 19:30 FreeFem++ Script 1 KB
makeref 11/02/2020 19:30 FreeFem++ Script 5 KB
NSP1P1 11/02/2020 19:30 FreeFem++ Script 2 KB
159 itens
1 item selecionado 2,15 KB

Instalando um editor C++ :

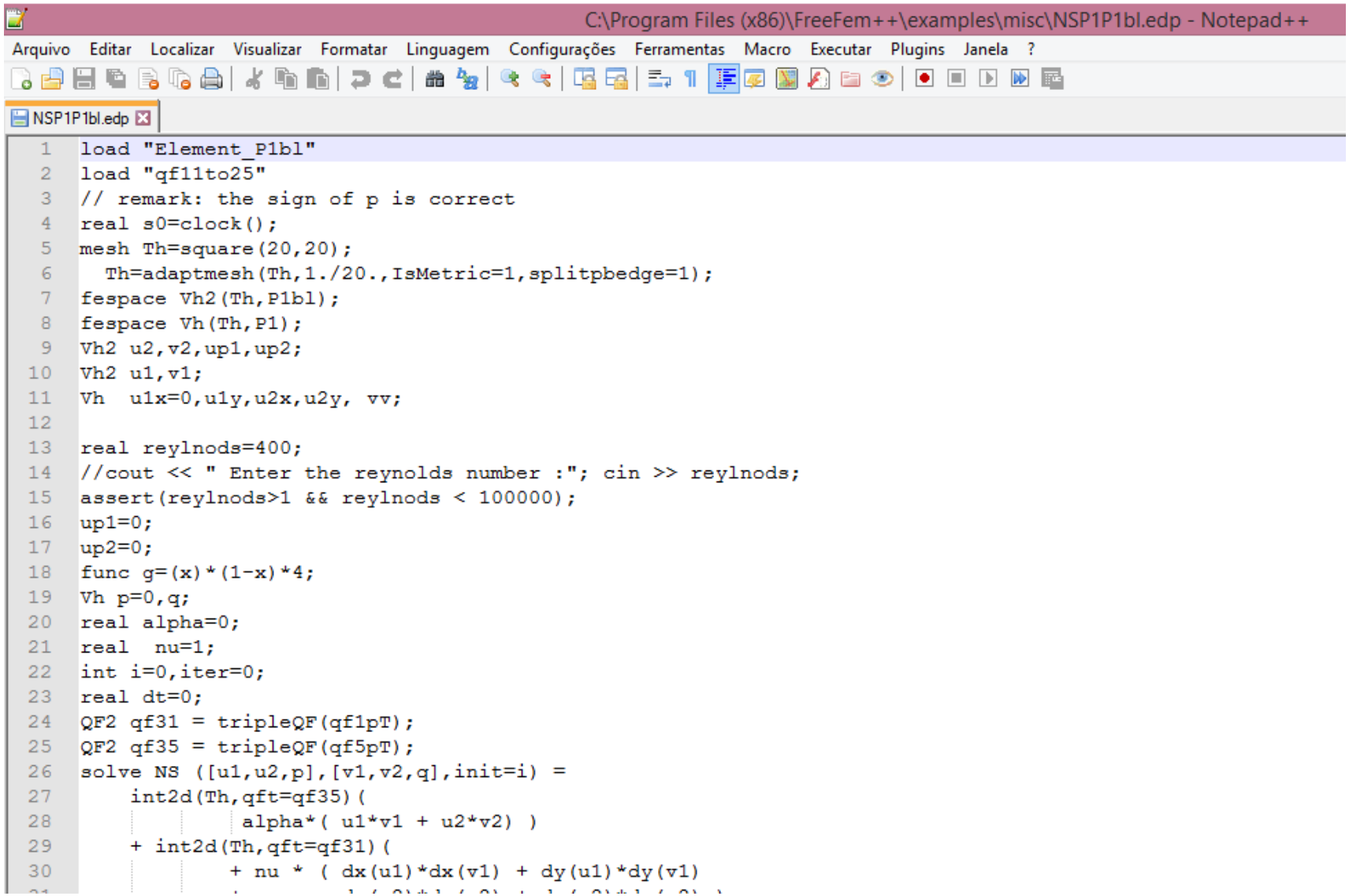
- Download: https://notepad-plus-plus.org/downloads/



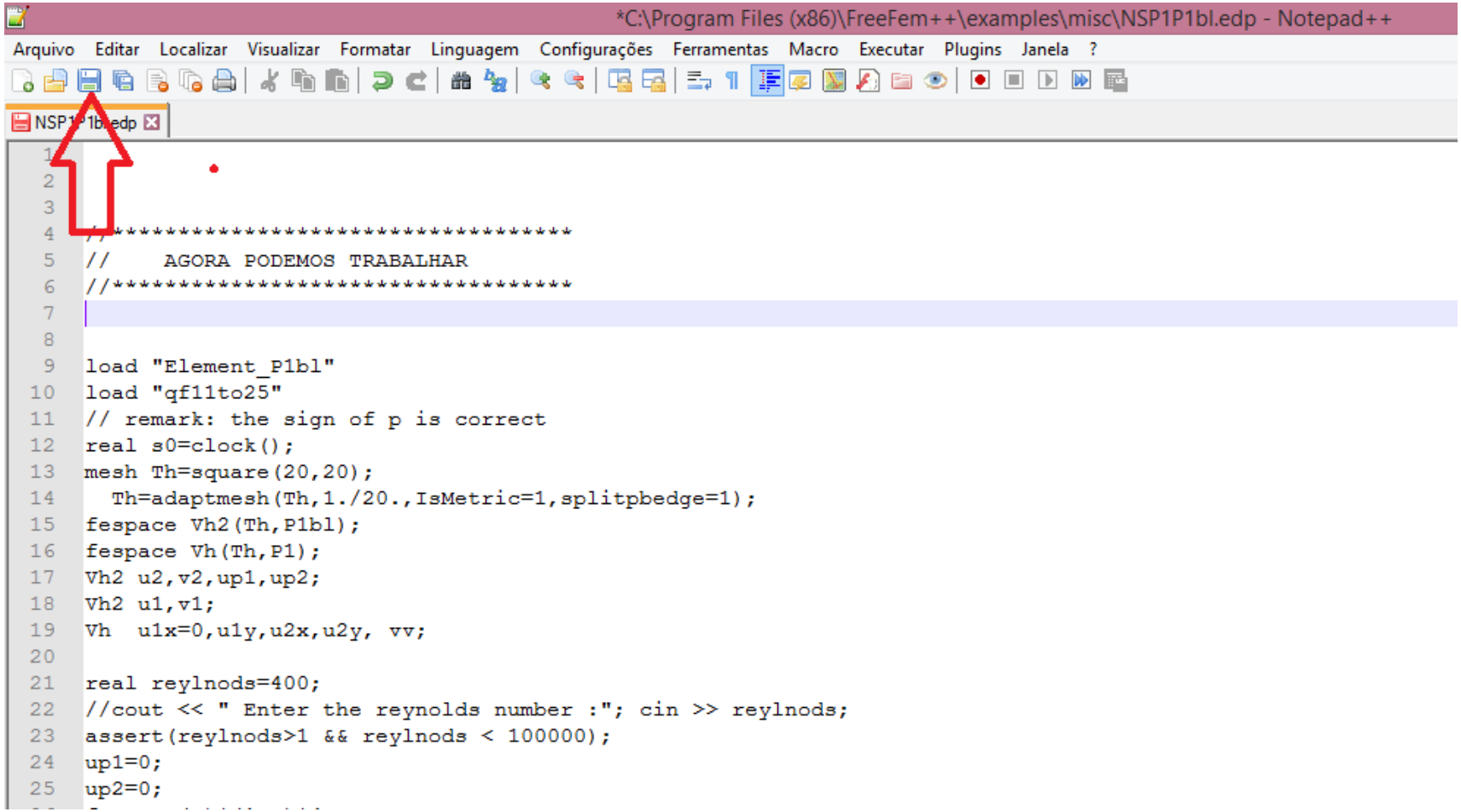
Abrindo um arquivo:



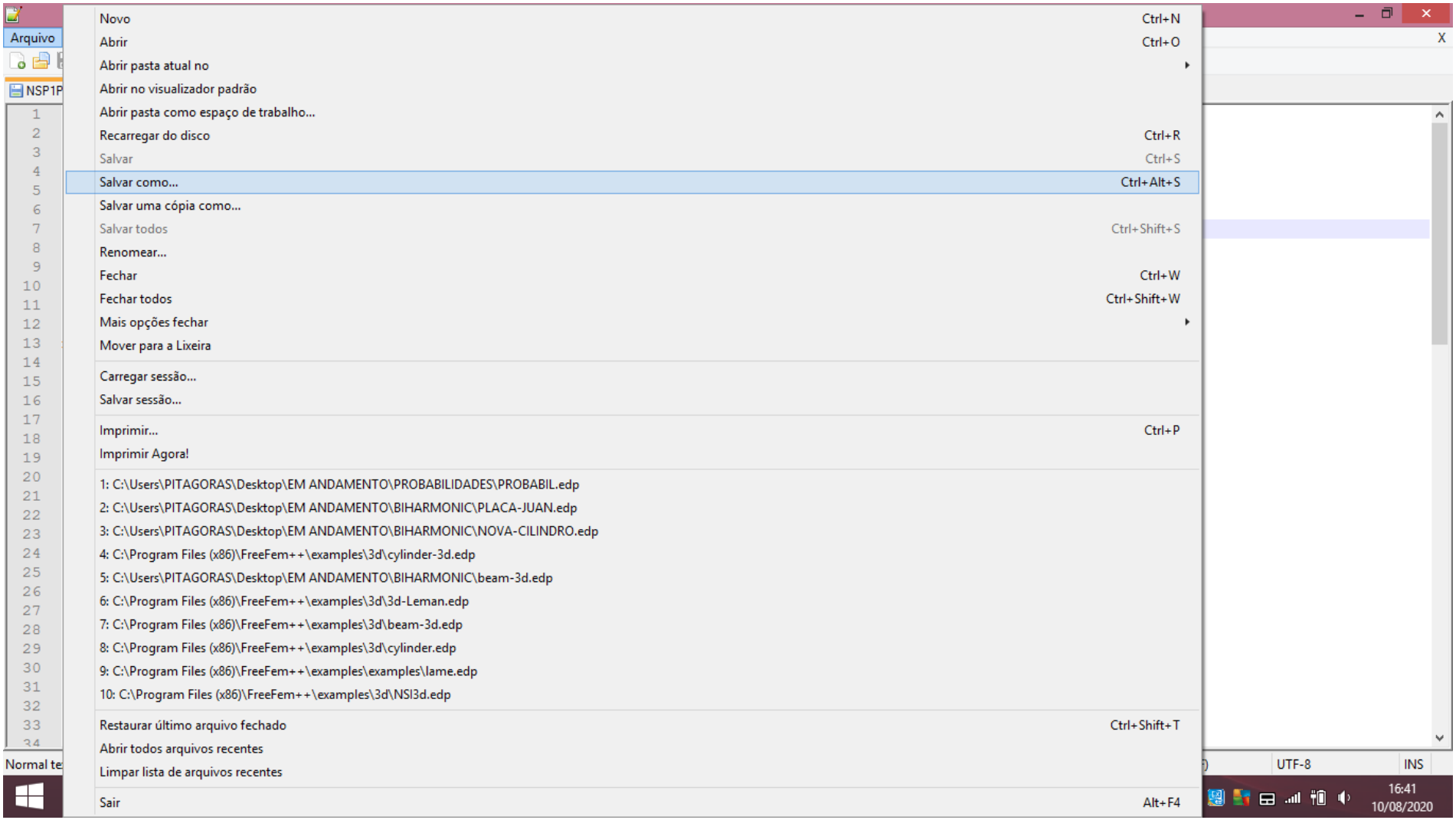
Abrindo um arquivo:



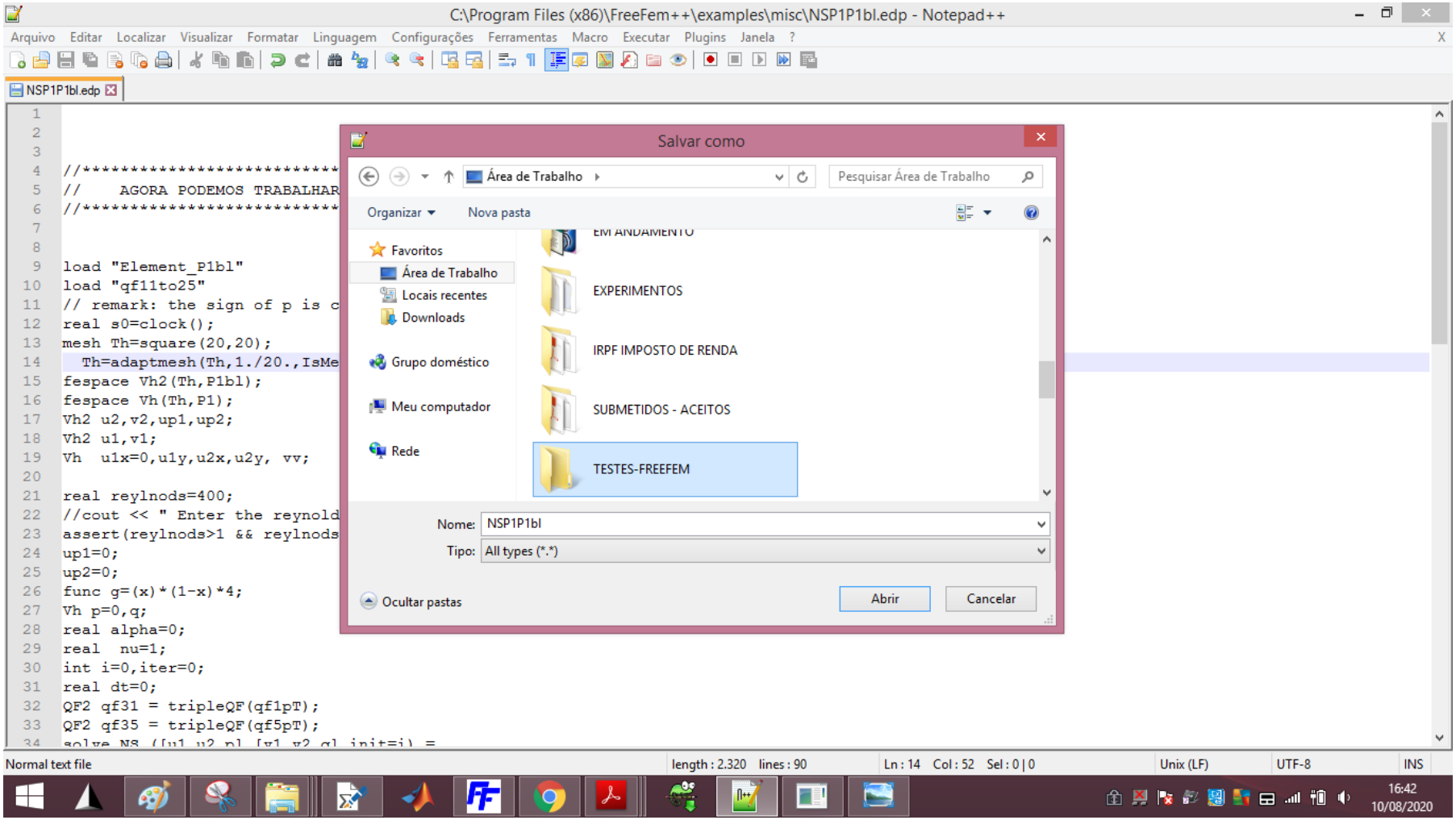
Editando um arquivo :



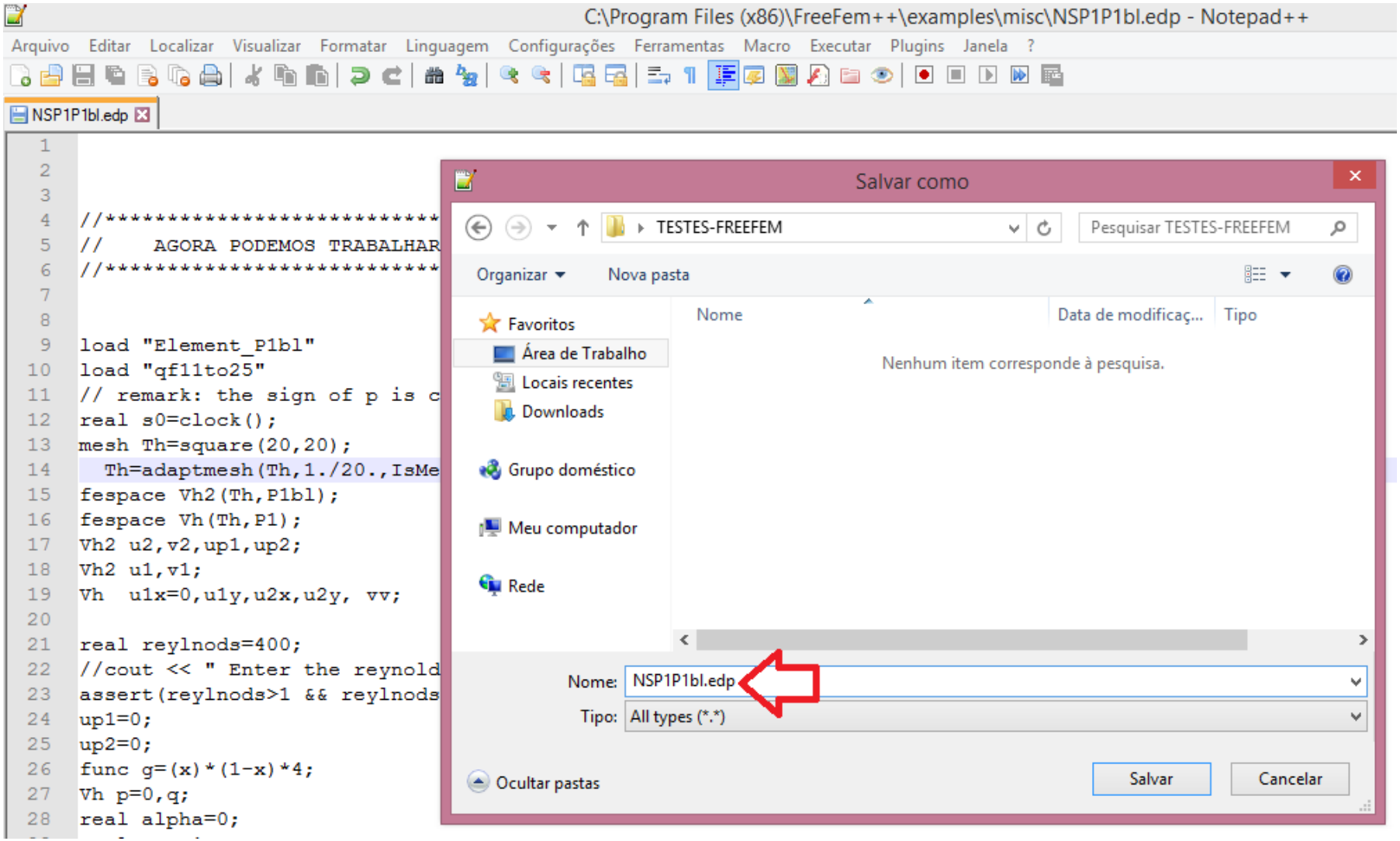
Salvando um arquivo:



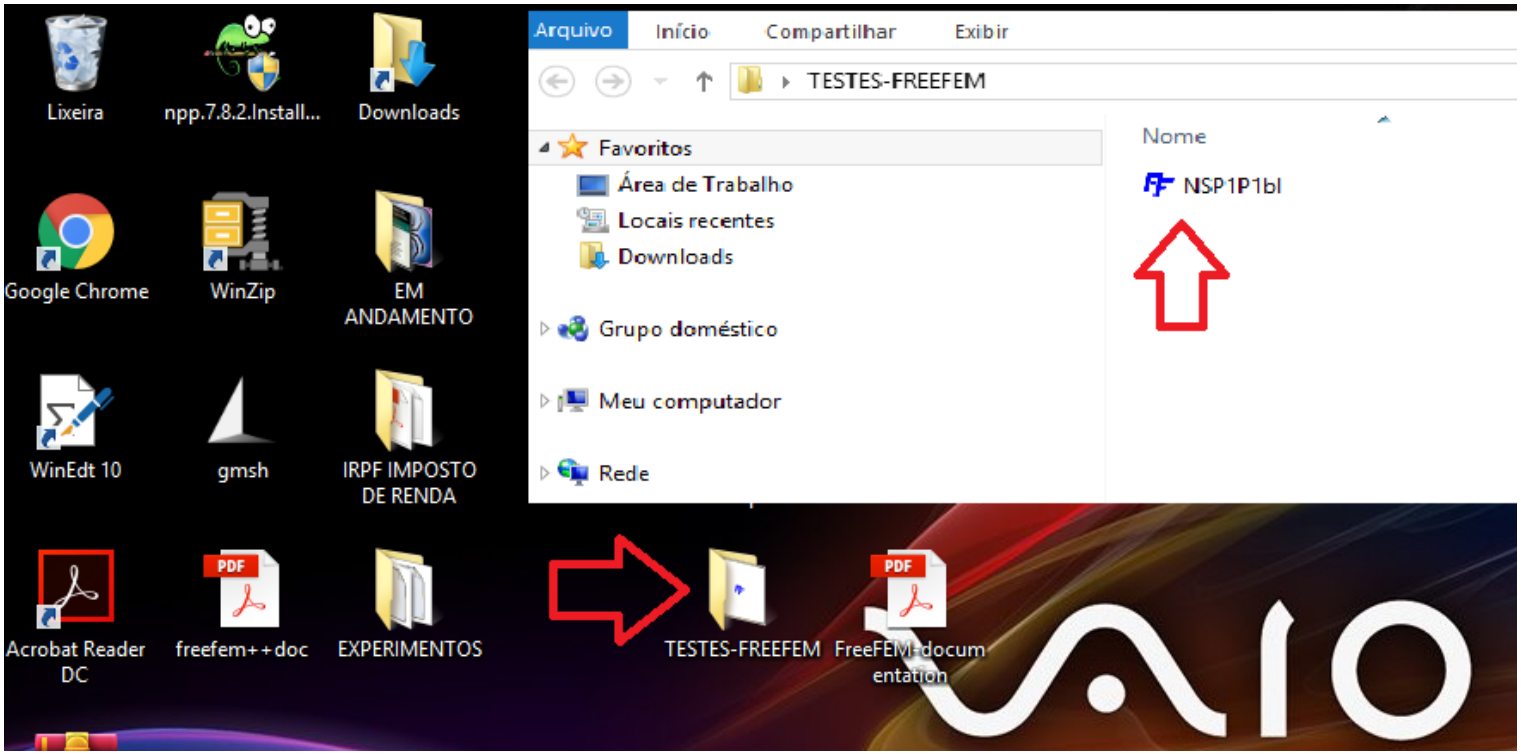
Salvando um arquivo:



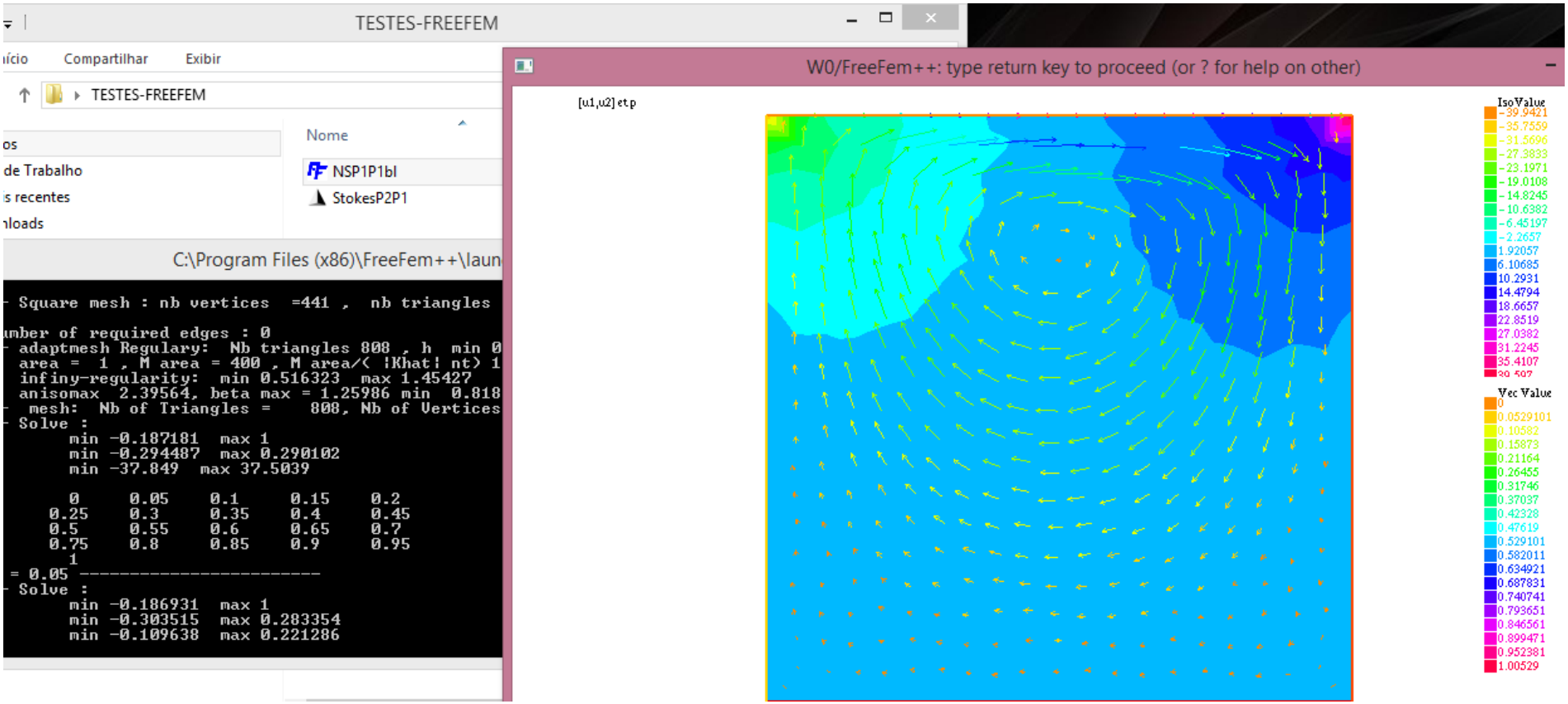
Salvando na extensão **.edp**:



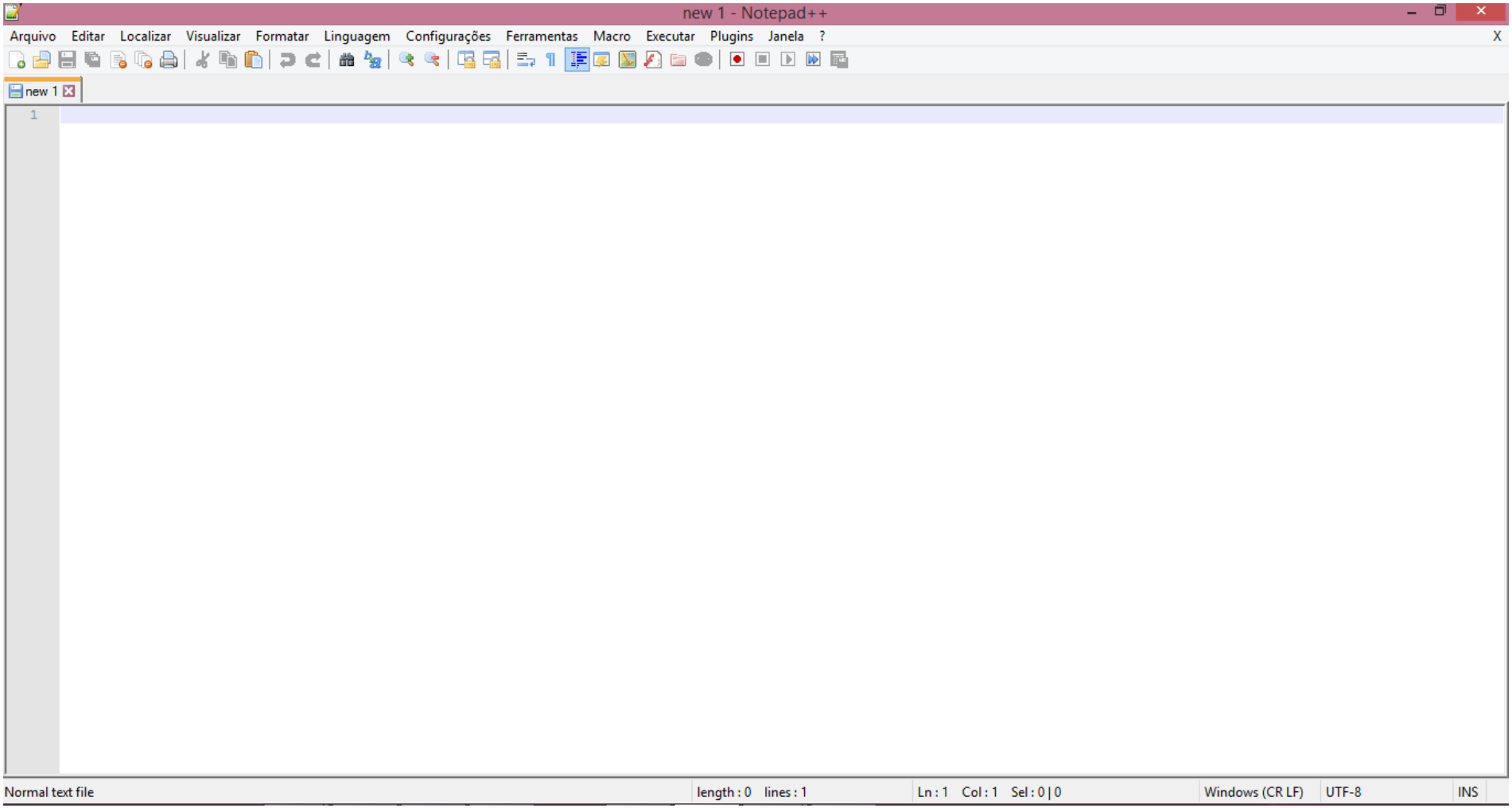
Nosso primeiro arquivo:



Dois cliques e...:



Novo Documento criado no Notepad++:



Alguns comandos iniciais:

- Os programas FreeFem devem ser salvos em arquivos com extensões .edp
- Cada variável deve ser digitada e declarada em uma instrução, que é separada da seguinte por ponto e vírgula ;
- O símbolo // em um programa FreeFem é usado para escrever declarações que não serão computadas. Outra opção também é:
  /*
  Tudo o que estiver aqui, estará comentado.
  */
- func f= $x*(1-x)*y*(1-y)$;
  - Acima, temos a declaração da função f, que depende das variáveis x e y.

Alguns comandos iniciais:

- int i, n = 20; real v =0.3; real [int] xx (n), yy (n);
  - A primeira instrução define i e n como inteiros, em particular definimos $n = 20$;
  - A segunda instrução define v número real, sendo $v = 0.3$;
  - A terceira instrução define xx e yy como vetores de comprimento $n = 20$.
  - As variáveis x, y e z são pré-definidas no programa como variáveis de construção dos eixos.
- border C(t=0,2*pi) { x=cos(t); y=sin(t); label=1; } // (label opcional) define a fronteira do círculo unitário.
  - Acima, declaramos a fronteira (border). A forma como ela será construída vem após a declaração border.

Alguns comandos iniciais (Elementos Finitos):

- mesh Th = buildmesh(C(200));
  - ⊙ mesh = constrói a malha de elementos finitos no círculo unitário C.
  - ⊙ Th = é a triangulação do domínio.
  - ⊙ buildmesh = geração de malha não uniforme.

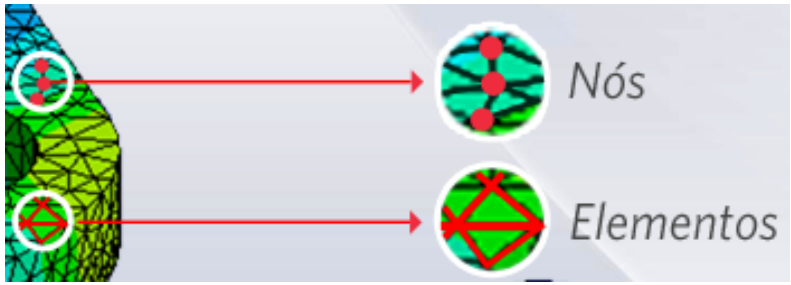


Figura: Elementos finitos

- ⊙ **Observação (Comandos para malhas):**

2D: mesh; **3D volume**: mesh3 ; **superfície 3D:** meshS; **curva 3D:** meshL.

Comandos Iniciais:

```
//——— MEDIT ———

load "medit"

//———- PONTOS NA FRONTEIRA ———

int C0=100;

//——— DESCRIÇÃO DA FRONTEIRA ———

border C00(t=0,2*pi){x=10*cos(t); y=10*sin(t); label=C0;}

//———-MALHA (Delaunay) ———

mesh Th=buildmesh(C00(200));

//—— PLOTANDO A MALHA ——

medit("2D Mesh", Th);
```

PLOTS:

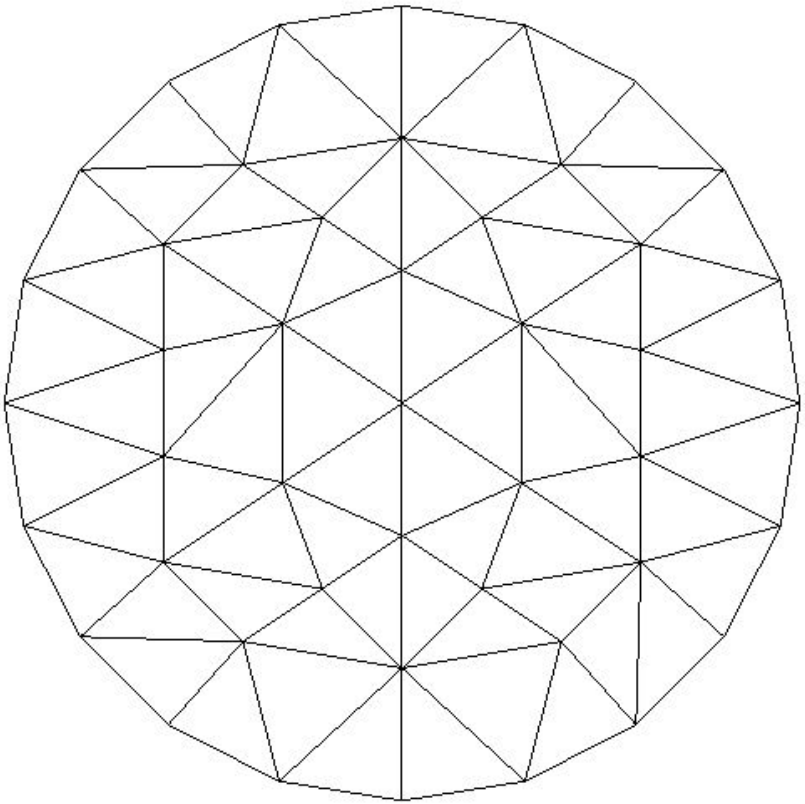


Figura: C(20)

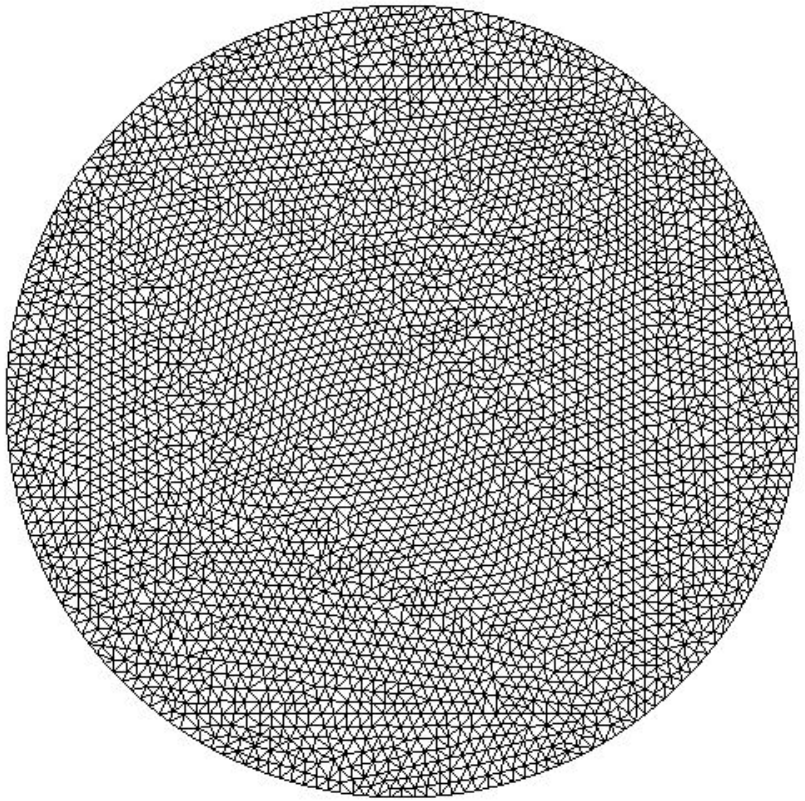


Figura: C(200)

Referências Bibliográficas


F. Hecht. New development in freefem++. J. Numer. Math., 20(3-4):251–265, 2012.


F. Hecht. https://www.ljll.math.upmc.fr/hecht/ftp/ff++/2019-Kenitra/Cours-Kenitra-2019.pdf

Obrigado pela presença na parte 1!