Homework Assignment 2

MATH 588 - Introduction to FEM

Student: Lazizbek Sadullaev
ID: 011808389

March 4, 2025

MATH 588
Introduction to FEM

Homework assignment 1
Date assigned: February 12, 2025
Due date: **February 21, 2025**

- Include a cover page and *this* problem sheet
- Include the printout of your program(s) (if any) for completeness

PROBLEM:
Consider the Dirichlet problem

$$\begin{cases} -\Delta u = -\sin \pi x \sin 2\pi y \text{ in } \Omega \\ u = 0 \text{ on } \partial\Omega. \end{cases} \tag{1}$$

Let Ω be square $(0, 1) \times (0, 1)$.

Write down the variational formulation and solve it using **FreeFem++** package. Include plot of the solution with your submission.

HW #2. M 588.

Problem

Consider the Dirichlet prob-m:

(1) $$-\Delta u = -\sin(\pi x)\cdot\sin(2\pi y) \quad \text{in } \Omega \qquad (1)$$

(2) $$u = 0 \quad \text{on} \quad \partial\Omega,$$ where

$$\Omega = \{(x,y)\in\mathbb{R}^2 : \; 0<x<1, \; 0<y<1\}.$$

Sol-n:

(a) Variational formulation:

Consider the following bilinear form & the inner product

(3) $$a(u,v) = \int_\Omega \nabla u\cdot\nabla v \, d\Omega \qquad u, v \in H_0^1$$

where

(4) $$H_0^1 = \{ u\in L^2, \; u|_{\partial\Omega} = 0 \},$$ and

(5) $$(f,v) = \int_\Omega f\cdot v \, d\Omega \qquad v\in H_0^1$$

Then consider (1):

$$-\Delta u = -\sin(\pi x)\cdot\sin(2\pi y) \qquad \Big\{ \text{on both sides (B.S.)} \; \forall v\in V = H^1 = \{u\in L^2\}.$$

$$-\Delta u\cdot v = -\sin(\pi x)\cdot\sin(2\pi y)\cdot v \qquad \Big\{ \int_\Omega(\cdot)\, d\Omega \text{ of BS:}$$

(6) $$\int_\Omega -\Delta u\cdot v \, d\Omega = \int_\Omega -\sin(\pi x)\cdot\sin(2\pi y)\cdot v \, d\Omega$$

Now applying Green's Identity to the L.H.S. of (6):

(7) $$\int_\Omega -\Delta u \cdot v \, d\Omega = \int_\Omega -\nabla(\nabla u)\, v \, d\Omega =$$

$$= \int_\Omega \nabla u \cdot \nabla v \, d\Omega - \int_{\partial\Omega} v \cdot \overset{= 0 \text{ on } \partial\Omega \text{ by } (2)}{u} \, ds$$

$$= \int_\Omega \nabla u \cdot \nabla v \, d\Omega - \underbrace{\int_{\partial\Omega} v \cdot 0 \, ds}_{=0} = \int_\Omega \nabla u \cdot \nabla v \, d\Omega .$$

So replacing this into (6):

$$\int_\Omega -\Delta u \cdot v \, d\Omega = \int_\Omega \nabla u \cdot \nabla v \, d\Omega \quad \&$$

(8) $$\boxed{\int_\Omega \nabla u \cdot \nabla v \, d\Omega = \int_\Omega -\sin(2\pi x)\cdot \sin(2\pi y) \cdot v \, d\Omega}$$

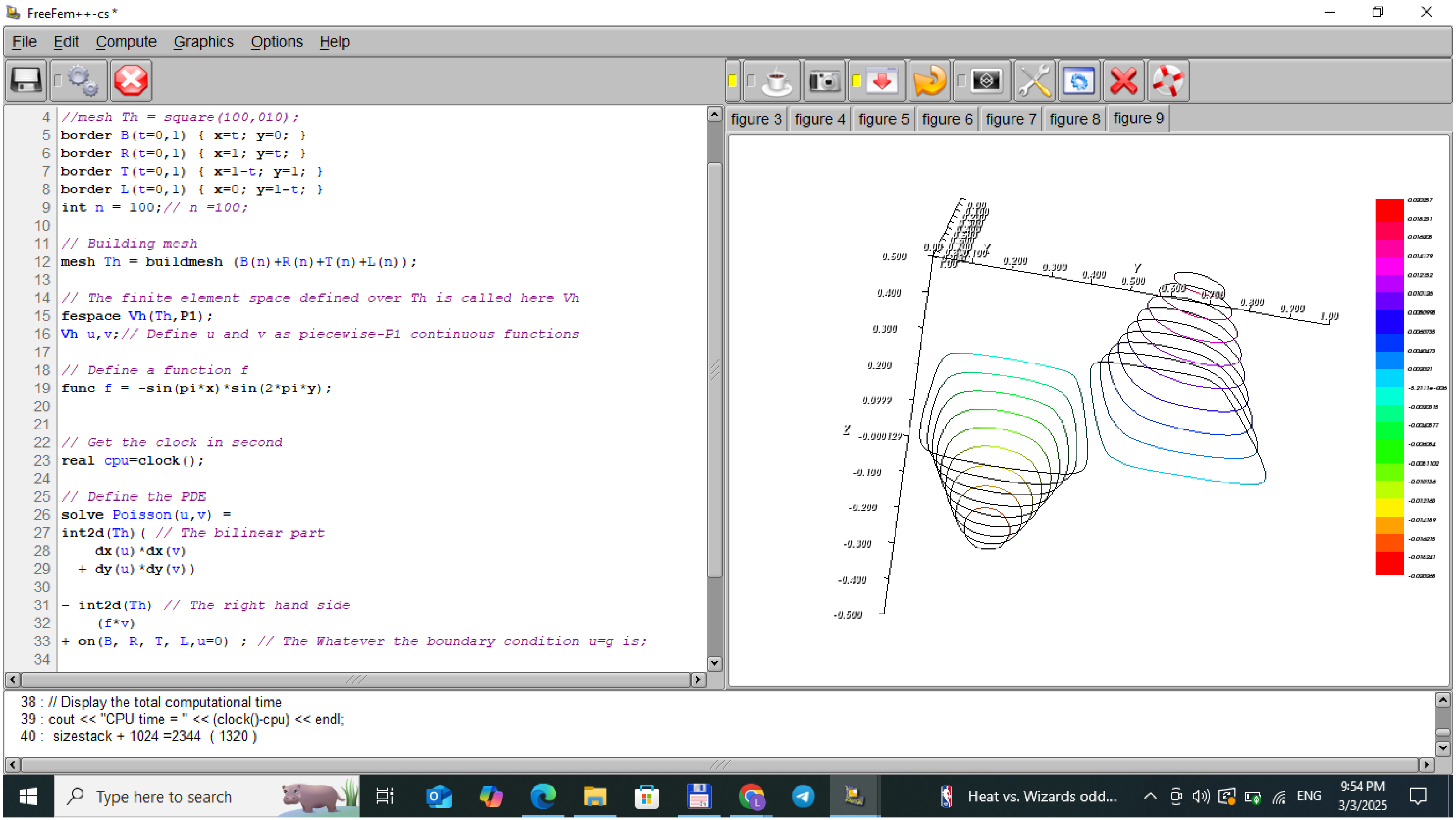
is the variational formulation of (1) + (2).

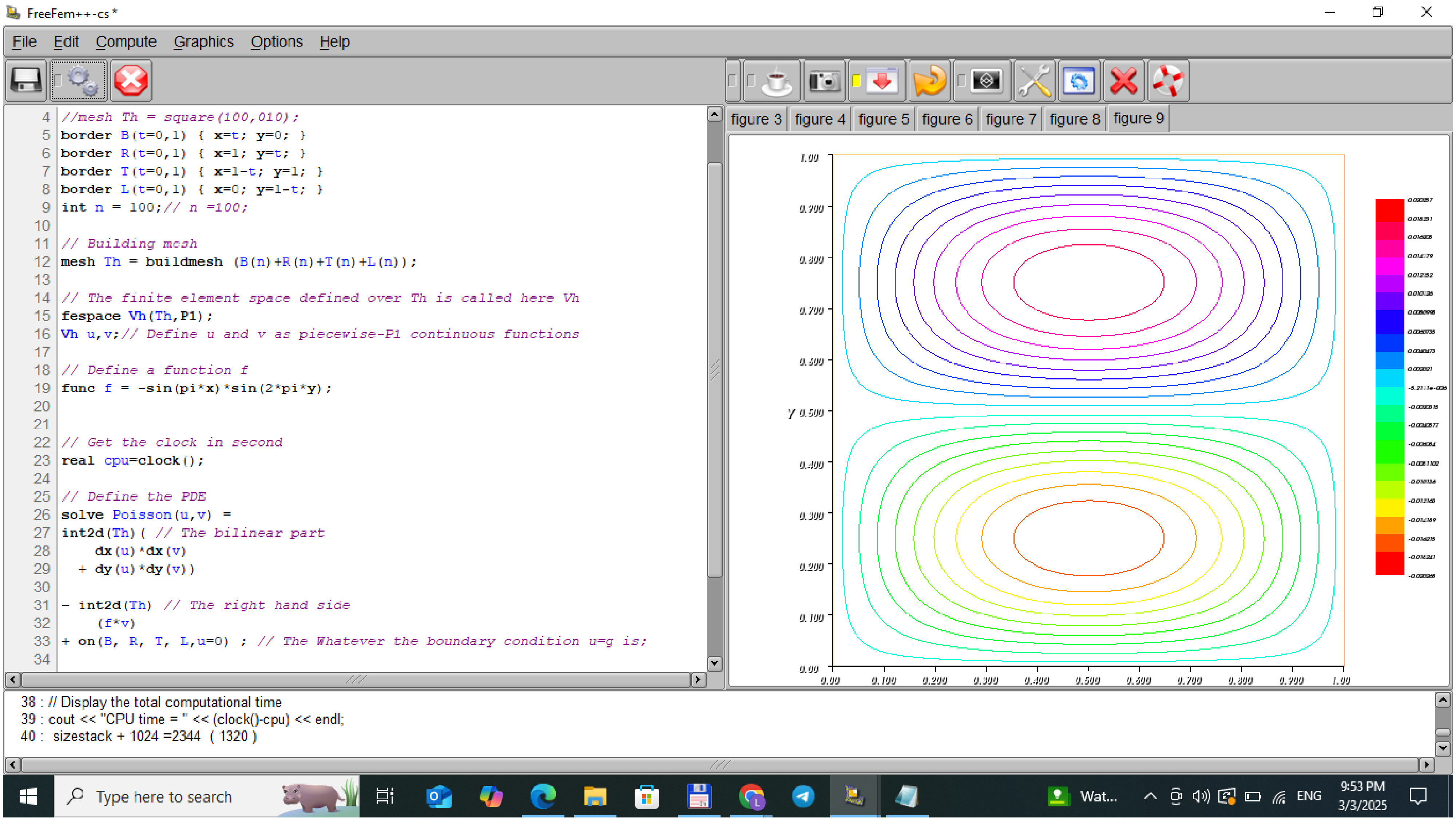
(b) I'll try to attach the code with result as a separate page.

Page for FreeFem++ Code

FreeFem++ Code

```
// Define mesh boundary
//border C(t=0, 2*pi){x=cos(t); y=sin(t);}

//mesh Th = square(100,010);
border B(t=0,1) { x=t; y=0; }
border R(t=0,1) { x=1; y=t; }
border T(t=0,1) { x=1-t; y=1; }
border L(t=0,1) { x=0; y=1-t; }
int n = 100;// n =100;

// Building mesh
mesh Th = buildmesh (B(n)+R(n)+T(n)+L(n));

// The finite element space defined over Th is called here Vh
fespace Vh(Th,P1);
Vh u,v;// Define u and v as piecewise-P1 continuous functions

// Define a function f
func f = -sin(pi*x)*sin(2*pi*y);

// Get the clock in second
real cpu=clock();

// Define the PDE
solve Poisson(u,v) =
int2d(Th)( // The bilinear part
         dx(u)*dx(v)
  + dy(u)*dy(v))

- int2d(Th) // The right hand side
         (f*v)
+ on(B, R, T, L,u=0) ; // The Whatever the boundary condition u=g is;

// Plot the result
plot(u);

// Display the total computational time
cout << "CPU time = " << (clock()-cpu) << endl;
```


FreeFem++-cs *
File Edit Compute Graphics Options Help
figure 3 figure 4 figure 5 figure 6 figure 7 figure 8 figure 9
4 //mesh Th = square(100,010);
5 border B(t=0,1) { x=t; y=0; }
6 border R(t=0,1) { x=1; y=t; }
7 border T(t=0,1) { x=1-t; y=1; }
8 border L(t=0,1) { x=0; y=1-t; }
9 int n = 100;// n =100;
10
11 // Building mesh
12 mesh Th = buildmesh (B(n)+R(n)+T(n)+L(n));
13
14 // The finite element space defined over Th is called here Vh
15 fespace Vh(Th,P1);
16 Vh u,v;// Define u and v as piecewise-P1 continuous functions
17
18 // Define a function f
19 func f = -sin(pi*x)*sin(2*pi*y);
20
21
22 // Get the clock in second
23 real cpu=clock();
24
25 // Define the PDE
26 solve Poisson(u,v) =
27 int2d(Th)(// The bilinear part
28 dx(u)*dx(v)
29 + dy(u)*dy(v))
30
31 - int2d(Th) // The right hand side
32 (f*v)
33 + on(B, R, T, L,u=0) ; // The Whatever the boundary condition u=g is;
34
38 : // Display the total computational time
39 : cout << "CPU time = " << (clock()-cpu) << endl;
40 : sizestack + 1024 =2344 (1320)
Type here to search
Heat vs. Wizards odd...
ENG
9:54 PM
3/3/2025


FreeFem++-cs *
File Edit Compute Graphics Options Help
//mesh Th = square(100,010);
border B(t=0,1) { x=t; y=0; }
border R(t=0,1) { x=1; y=t; }
border T(t=0,1) { x=1-t; y=1; }
border L(t=0,1) { x=0; y=1-t; }
int n = 100;// n =100;
// Building mesh
mesh Th = buildmesh (B(n)+R(n)+T(n)+L(n));
// The finite element space defined over Th is called here Vh
fespace Vh(Th,P1);
Vh u,v;// Define u and v as piecewise-P1 continuous functions
// Define a function f
func f = -sin(pi*x)*sin(2*pi*y);
// Get the clock in second
real cpu=clock();
// Define the PDE
solve Poisson(u,v) =
int2d(Th)(// The bilinear part
dx(u)*dx(v)
+ dy(u)*dy(v))
- int2d(Th) // The right hand side
(f*v)
+ on(B, R, T, L,u=0) ; // The Whatever the boundary condition u=g is;
figure 3
figure 4
figure 5
figure 6
figure 7
figure 8
figure 9
38 : // Display the total computational time
39 : cout << "CPU time = " << (clock()-cpu) << endl;
40 : sizestack + 1024 =2344 (1320)
Type here to search
Wat...
ENG
9:53 PM
3/3/2025