Thomas Keller
SDSU ID 828545348

Homework 3

COMP 670
October 27, 2022

Problem 1

Consider example elliptic3D.m

1. What is the domain of the PDE (the region where the PDE holds)?
   7x8x9 cells

2. What is the equation this example is solving?
   Poisson equation
   $$\Delta u = 0 \tag{1}$$

3. What the force function or right hand side?
   $$\begin{aligned} \Delta u &= RHS \\ \nabla^2 u &= 0 \end{aligned} \tag{2}$$

4. What type of boundary conditions?
   Dirichlet, b = 0

5. Change the boundary condition to be 100 on the front and back face of the cube.
   RHS(:,:,1)=100
   RHS(:,:,9)=100

6. Run the example with all the changes you have made.
   On the next pages are the plots for the changes, page 1 to 9.

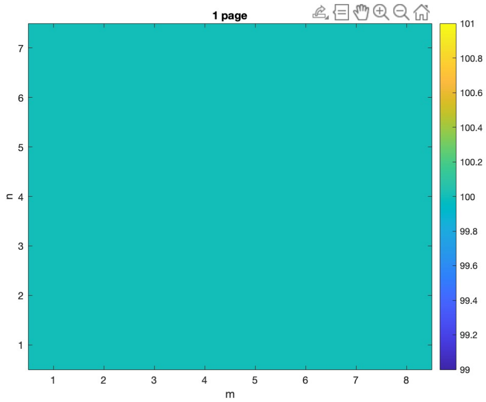

Figure 1: Page 1

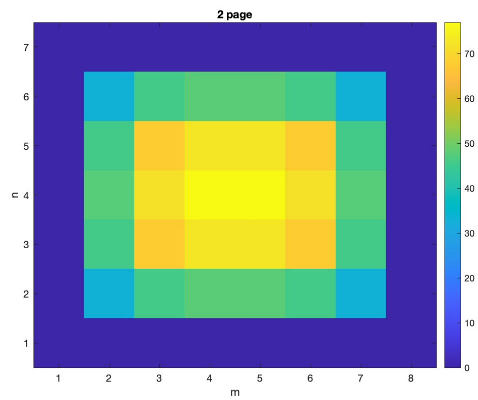

Figure 2: Page 2

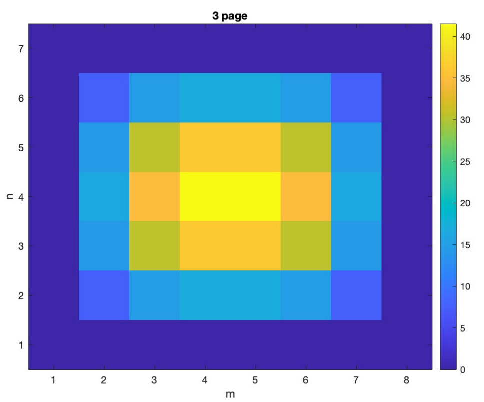

Figure 3: Page 3

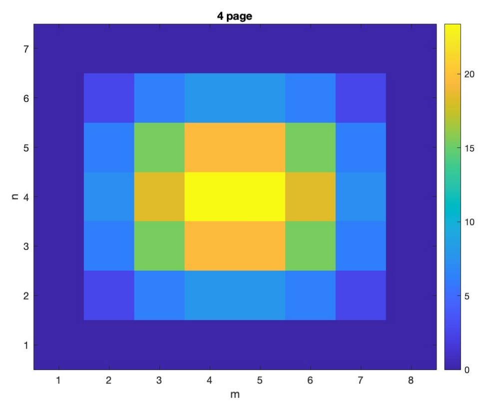

Figure 4: Page 4

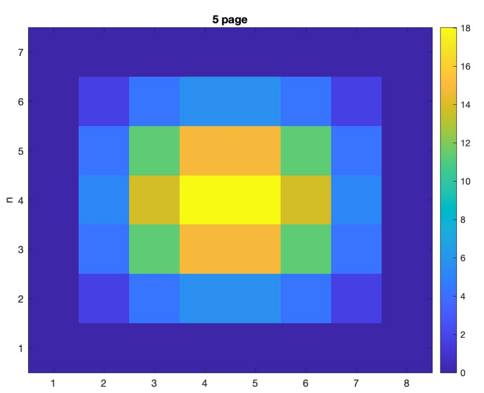

Figure 5: Page 5

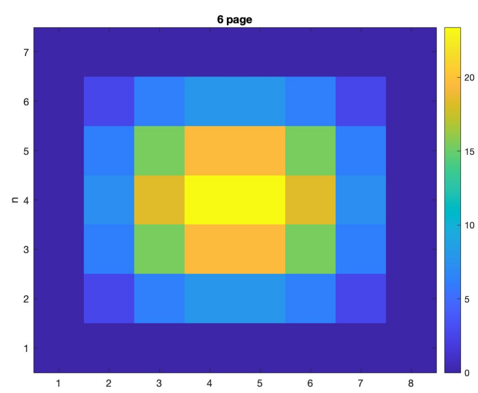

Figure 6: Page 6

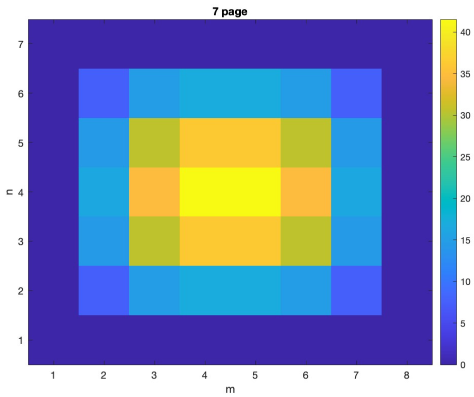

Figure 7: Page 7

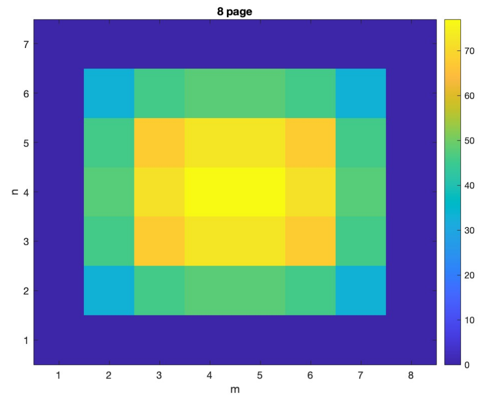

Figure 8: Page 8

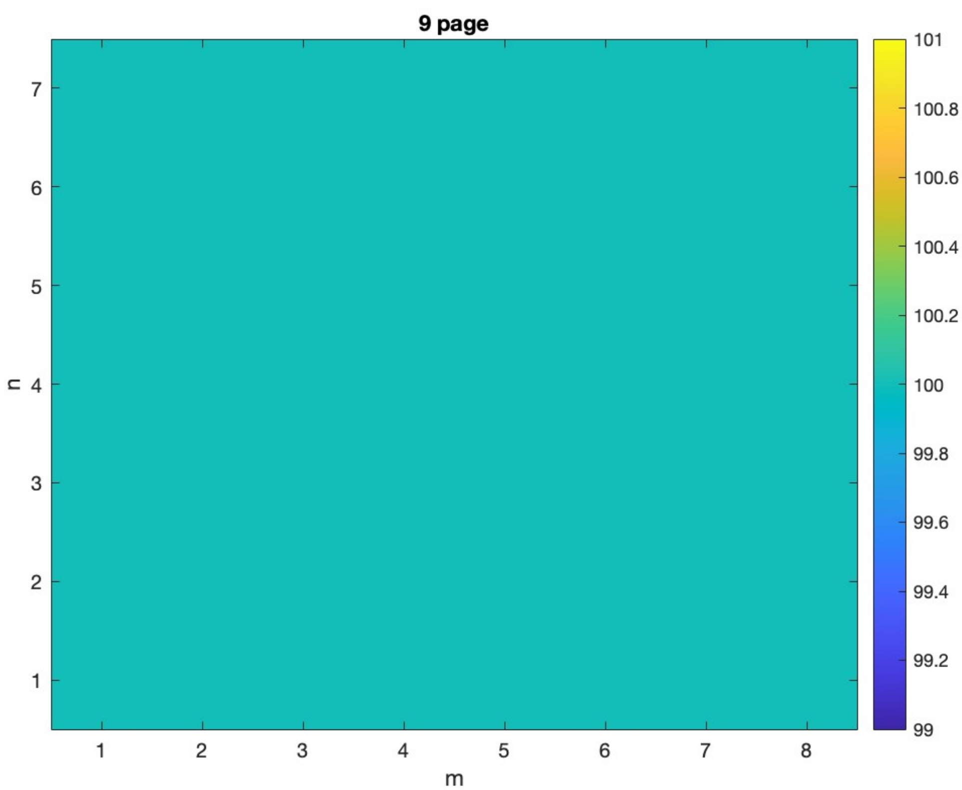

Figure 9: Page 9

7. Please, include all m files you have modified and a document (named elliptic3DWork.docx) describing
explicitly the PDE to a Canvas message with subject COMP670 hw3 elliptic3D.

The Matlab code includes the following changes:

- line 21: RHS(:, :, 9) = 100;

```
1  % 3D Staggering example using a 3D Mimetic laplacian
2  clc
3  close all
4
5  addpath('../mole_MATLAB')
6
7  k = 2; % Order of accuracy
8  m = 5; % -> 7  Number of cells along x-axis
9  n = 6; % -> 8
10 o = 7; % -> 9
11
12 L = lap3D(k, m, 1, n, 1, o, 1); % 3D Mimetic laplacian operator
13 L = L + robinBC3D(k, m, 1, n, 1, o, 1, 1, 0); % Dirichlet BC
14
15 RHS = zeros(m+2, n+2, o+2);
16 % RHS
17
18 RHS(:, :, 1) = 100; % Known value at the cube's front face
19 RHS
20 RHS(:, :, 9) = 100; % Known value at the cube's back face
21 RHS
22
23 RHS = reshape(RHS, (m+2)*(n+2)*(o+2), 1); % Create vector with 6*7*8 rows
24 % RHS
25
26 SOL = L\RHS;
27
28 SOL = reshape(SOL, m+2, n+2, o+2); % reshape back to matrix
29
30 p = (8); % Page to be displayed
31 page = SOL(:, :, p);
32
33 imagesc(page)
34 title([num2str(p) ' page'])
35 xlabel('m')
36 ylabel('n')
37 set(gca, 'YDir', 'Normal')
38 colorbar
39
```

Listing 1: changed Matblab code for file elliptic3D.m

Problem 2

Consider example parabolic1D.m (in this example you will not modify the MATLAB source code)

1. What is the domain of the PDE (the region where the PDE holds)?
   Domain: 0-1

2. What is the equation this example is solving?
   Heat equation

$$\frac{\partial u}{\partial t} = \alpha \Delta u \tag{3}$$

3. What is the initial condition?
   U + zeros(m+2,1); → U= (0 ; 0 ; 0 ; 0 ; 0 ; ...)

4. What are the boundary conditions?
   U(1)=100; U(end)=100; Dirichlet

5. What the force function or right hand side?
   RHS=0

6. Explore and explain what is the Neumann stability criterion.
   Explore:

$$dt = \frac{dx^2}{3 \cdot \alpha} \tag{4}$$

   Explain:
   The system is conditionally stable for the backward euler method. The equation 4 is the criteria which has to be fulfilled for a stable system. Otherwise we get inappropriate behavior due to scheme.

7. How the partial derivative with respect to time is discretized in the explicit scheme of this code?
   Forward Euler

8. How the partial derivative with respect to time is discretized in the implicit scheme of this code?
   Backward Euler

9. Run the example with the explicit scheme and store its numerical solution.
   U =
   1.0e+02 *
   1.000000000000000
   0.999990291533084
   0.999973708183323
   0.999967356462446
   0.999973708183323
   0.999990291533084
   1.000000000000000

10. Run the example with the implicit scheme and store its numerical solution.
    U =
    1.0e+02 *
    1.000000000000000
    0.999964740685190
    0.999904513096742
    0.999881444848386
    0.999904513096743
    0.999964740685190
    1.000000000000000

11. Make a plot of both solutions, explicit in blue and implicit in red. You can leverage on the plotting code that the example has.

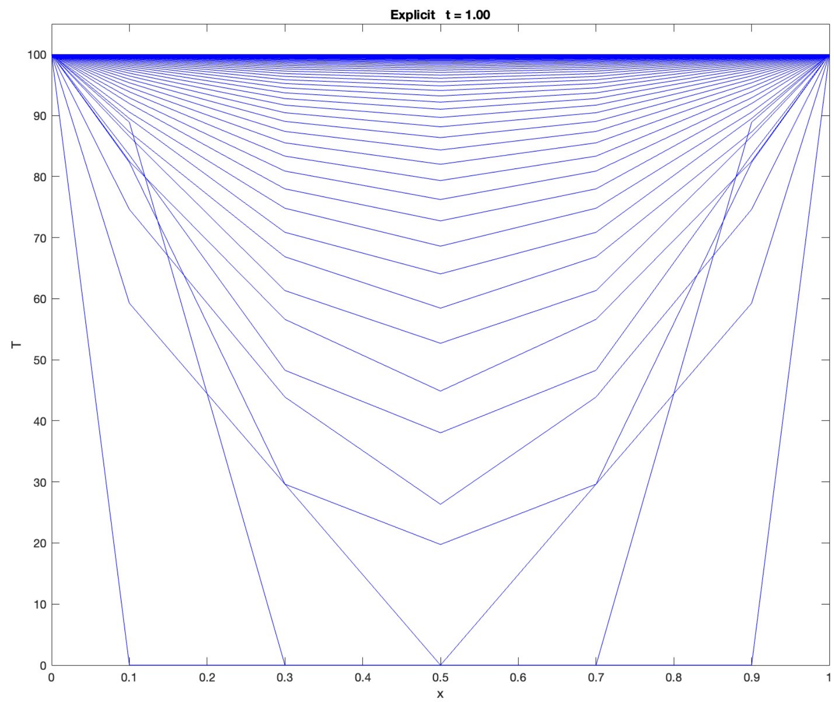

Figure 10: Explicit

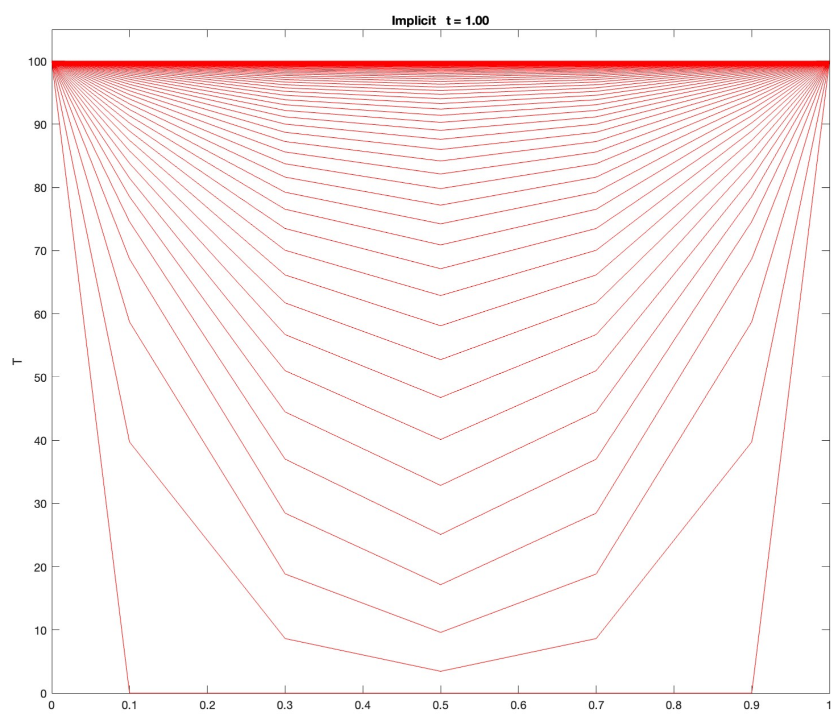

Figure 11: Implicit

12. Please, include the plot and a document (named parabolic1DWork.pdf) describing explicitly the PDE to a Canvas message with subject COMP670 hw3 parabolic1D.

Problem 3

Write an extension of the parabolic1D.m example (name parabolic2D.m) to apply MOLE with explicit time discretization scheme to the following 2D heat equation
Use what you have learned from elliptic3D.m and parabolic1D.m examples.
1. Be sure the code runs successfully.
2. Please, include parabolic2D.m to a Canvas message with subject COMP670 hw3 parabolic2D.

The following Matlab code shows the code for parabolic2D.m file.

```
% Solves the 2D Heat equation, changed example wave2D/parabolic1D
% Thomas Keller, COMP670 HW3
clc; close all;

addpath('../mole_MATLAB')

% Spatial discretization
k = 2;   % Order of accuracy
m = 3*k+1;  % Number of cells along the x-axis
n = m;   % Number of cells along the y-axis
a = 0;   % West
b = 1;   % East
c = 0;   % South
d = 1;   % North
dx = (b-a)/m;  % Step length along the x-axis
dy = (d-c)/n;  % Step length along the y-axis

% 2D Staggered grid
xgrid = [a a+dx/2 : dx : b-dx/2 b];
ygrid = [c c+dy/2 : dy : d-dy/2 d];

% Create 2D meshgrid
[X, Y] = meshgrid(xgrid, ygrid);

% Mimetic operator Laplacian 2D
L = lap2D(k, m, dx, n, dy);

% alpha
alpha = 1;

% Check neumann stability
% Neumann stability criterion
dt = dx^2/(5*alpha);

% BC
U = zeros(m+2, n+2);
U(1,:)=100;
U(:,1)=100;
```

```
    U(end,:)=100;
    U(:,end)=100;
    U
    U  = reshape(U, (m+2)*(n+2), 1);
    U

    % Simulation time
    TIME = 0.3;

    explicit = 1; % Explicit, for 0 -> implicit

    if explicit
        % Explicit
        L = alpha*dt*L + speye(size(L)); % S = speye returns a sparse scalar 1 to L

        % Time integration loop
        for t = 1 : TIME/dt+1
            % Plot result explicit
            mesh(X, Y, reshape(U, m+2, n+2))
            title(['Parabolic, Explicit Solution, Time = ' num2str(dt*t, '%1.2f')])
            xlabel('x')
            ylabel('y')
            zlabel('z')
            colorbar
            caxis([0, 100]) % change colors of plot
            axis([0 1 0 1 0 101])
            drawnow
            U = L*U; % Apply the operator
        end
    else
        % Implicit
        L = -alpha*dt*L + speye(size(L));

        % Time integration loop
        for t = 1 : TIME/dt+1
            % Plot result implicit
            mesh(X, Y, reshape(U, m+2, n+2))
            title(['Parabolic, Implicit Solution, Time = ' num2str(dt*t, '%1.2f')])
            xlabel('x')
            ylabel('y')
            zlabel('z')
            colorbar
            caxis([0, 100]) % change colors of plot
            axis([0 1 0 1 0 101])
            drawnow
            U = L\U; % Solve a linear system of equations (unconditionally stable)
        end
    end

```

Listing 2: changed Matblab code for file parabolic2D.m