# Table of Contents



```
clear; % clear variables
clc; % clear command window
close all
```

# 3C

```
clear; % clear variables
clc; % clear command window
close all

syms x y
f = exp(-x).*cos(y)- exp(-y).*cos(x);
d2f_dx2 = diff(f, x, 2);
d2f_dy2 = diff(f, y, 2);
disp(d2f_dx2);
disp(d2f_dy2);
laplacian_f = d2f_dx2 + d2f_dy2;
disp(laplacian_f);
```

```
exp(-x)*cos(y) + exp(-y)*cos(x)

- exp(-x)*cos(y) - exp(-y)*cos(x)

0
```

# 3D

```
clear; % clear variables
clc; % clear command window
close all

syms x y
f = x.^2 + 3.*y.^2;
d2f_dx2 = diff(f, x, 2);
d2f_dy2 = diff(f, y, 2);
disp(d2f_dx2);
disp(d2f_dy2);
laplacian_f = d2f_dx2 + d2f_dy2;
disp(laplacian_f);
```

```
2
```

*6*

*8*

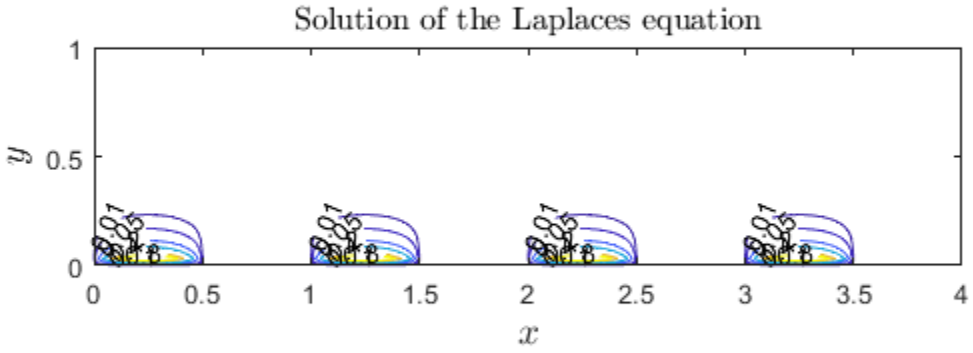
# 7

```
Lx=1; Ly=1; %rectangle dimensions
Nx=100;
Ny=100; %number of intervals in x,y directions
nx=Nx+1;
ny=Ny+1; %number of gridpoints in x,y directions
dx=Lx/Nx;
dy=Ly/Ny; %grid length in x,y directions
x=(0:Nx)*dx;
y=(0:Ny)*dy; %x,y values on grid
```

# 7a

```
clear; % clear variables
clc; % clear command window
close all

Lx=4; Ly=1; %rectangle dimensions
Nx=100;
Ny=100; %number of intervals in x,y directions
nx=Nx+1;
ny=Ny+1; %number of gridpoints in x,y directions
dx=Lx/Nx;
dy=Ly/Ny; %grid length in x,y directions
x=(0:Nx)*dx;
y=(0:Ny)*dy; %x,y values on grid


boundary_index=[1:nx, 1:nx:1+(ny-1)*nx, 1+(ny-1)*nx:nx*ny, nx:nx:nx*ny];
diagmatrix = [4*ones(nx*ny,1), -ones(nx*ny,4)];
A=spdiags(diagmatrix,[0 -1 1 -nx nx], nx*ny, nx*ny);
I=speye(nx*ny);
A(boundary_index,:)=I(boundary_index,:);

b=zeros(nx,ny);
b(:,4)=sin(2.*pi.*x); %BC for the bottom part of the rectangle
b(1,:)=0; %BC for the left part of the rectangle
b(:,ny)=0; %BC for the top part of the rectangle
b(nx,:)=0; %BC for the right part of the rectangle
b=reshape(b,nx*ny,1); %make column vector using natural ordering

u=A\b; %solution by Gaussian elimination
u=reshape(u,nx,ny); %back to (i,j) indexing

[X,Y]=meshgrid(x,y);
v=[2 1 0.8  0.4  0.2 0.05 0.01 0.05]; %contour levels
```

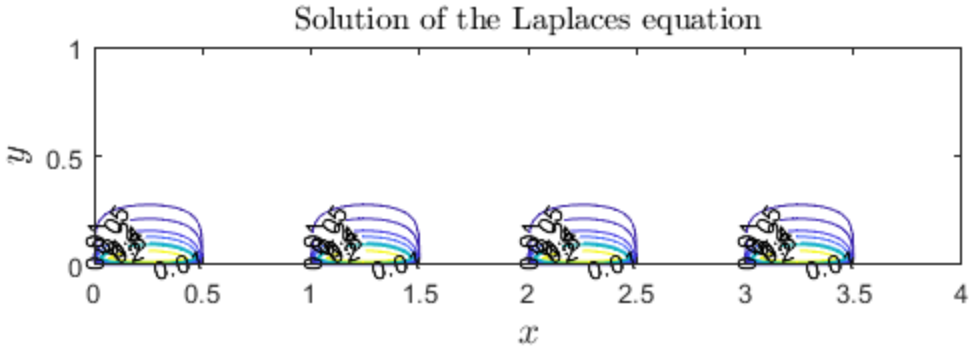
```
contour(X,Y,u',v,'ShowText','on'); %requires transpose
axis equal;
%set(gca,'YTick',[0 0.2 0.4 0.6 0.8 1]);
%set(gca,'XTick',[0 0.2 0.4 0.6 0.8 1]);
xlabel('$x$','Interpreter','latex','FontSize',14 );
ylabel('$y$','Interpreter','latex','FontSize',14);
title('Solution of the Laplaces
 equation','Interpreter','latex','FontSize',12);
```




# 7b

```
clear; % clear variables
clc; % clear command window
close all

Lx=4; Ly=1; %rectangle dimensions
Nx=100;
Ny=100; %number of intervals in x,y directions
nx=Nx+1;
ny=Ny+1; %number of gridpoints in x,y directions
dx=Lx/Nx;
dy=Ly/Ny; %grid length in x,y directions
x=(0:Nx)*dx;
y=(0:Ny)*dy; %x,y values on grid
```

```
boundary_index=[1:nx, 1:nx:1+(ny-1)*nx, 1+(ny-1)*nx:nx*ny, nx:nx:nx*ny];
diagmatrix = [4*ones(nx*ny,1), -ones(nx*ny,4)];
A=spdiags(diagmatrix,[0 -1 1 -nx nx], nx*ny, nx*ny);
I=speye(nx*ny);
A(boundary_index,:)=I(boundary_index,:);

b=zeros(nx,ny);
b(:,4)=3.*sin(2.*pi.*x); %BC for the bottom part of the rectangle
b(1,:)=0; %BC for the left part of the rectangle
b(:,ny)=0; %BC for the top part of the rectangle
b(nx,:)=0; %BC for the right part of the rectangle
b=reshape(b,nx*ny,1); %make column vector using natural ordering

u=A\b; %solution by Gaussian elimination
u=reshape(u,nx,ny); %back to (i,j) indexing

[X,Y]=meshgrid(x,y);
v=[2 1 0.8  0.4  0.2 0.05 0.01 0.05]; %contour levels
contour(X,Y,u',v,'ShowText','on'); %requires transpose
axis equal;
%set(gca,'YTick',[0 0.2 0.4 0.6 0.8 1]);
%set(gca,'XTick',[0 0.2 0.4 0.6 0.8 1]);
xlabel('$x$','Interpreter','latex','FontSize',14 );
ylabel('$y$','Interpreter','latex','FontSize',14);
title('Solution of the Laplaces
 equation','Interpreter','latex','FontSize',12);
```



*Published with MATLAB® R2023a*