Q1. For the numerical scheme

$$\frac{U_j^{n+1} - U_j^{n-1}}{2\Delta t} = \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{(\Delta x)^2}$$

The Truncation Error:

$$T(x_j, t_n) = \frac{u(x_j, t_{n+1}) - u(x_j, t_{n-1})}{2\Delta t} - \frac{u(x_{j+1}, t_n) - 2u(x_j, t_n) + u(x_{j-1}, t_n)}{(\Delta x)^2}$$

Using Taylor Expansion at (x_j, t_n):

$$u(x_j, t_{n+1}) = u(x_j, t_n) + \frac{\partial u}{\partial t}(x_j, t_n)\Delta t + \frac{1}{2}\frac{\partial^2 u}{\partial t^2}(x_j, t_n)(\Delta t)^2 + \frac{1}{6}\frac{\partial^3 u}{\partial t^3}(x_j, t_n)(\Delta t)^3 + O((\Delta t)^4) \quad - \quad (1)$$

$$u(x_j, t_{n-1}) = u(x_j, t_n) - \frac{\partial u}{\partial t}(x_j, t_n)\Delta t + \frac{1}{2}\frac{\partial^2 u}{\partial t^2}(x_j, t_n)(\Delta t)^2 - \frac{1}{6}\frac{\partial^3 u}{\partial t^3}(x_j, t_n)(\Delta t)^3 + O((\Delta t)^4) \quad - \quad (2)$$

$$(1) - (2), we\ have: \quad u(x_j, t_{n+1}) - u(x_j, t_{n-1}) = 2\frac{\partial u}{\partial t}(x_j, t_n)\Delta t + \frac{1}{3}\frac{\partial^3 u}{\partial t^3}(x_j, t_n)(\Delta t)^3 + O((\Delta t)^4)$$

$$\frac{u(x_j, t_{n+1}) - u(x_j, t_{n-1})}{2\Delta t} = \frac{\partial u}{\partial t}(x_j, t_n) + \frac{1}{6}\frac{\partial^3 u}{\partial t^3}(x_j, t_n)(\Delta t)^2 + O((\Delta t)^3)$$

$$u(x_{j+1}, t_n) = u(x_j, t_n) + \frac{\partial u}{\partial x}(x_j, t_n)\Delta x + \frac{1}{2}\frac{\partial^2 u}{\partial x^2}(x_j, t_n)(\Delta x)^2 + \frac{1}{6}\frac{\partial^3 u}{\partial x^3}(x_j, t_n)(\Delta x)^3 + \frac{1}{24}\frac{\partial^4 u}{\partial x^4}(x_j, t_n)(\Delta x)^4 + O((\Delta x)^5) \quad - \quad (3)$$

$$u(x_{j-1}, t_n) = u(x_j, t_n) - \frac{\partial u}{\partial t}(x_j, t_n)\Delta x + \frac{1}{2}\frac{\partial^2 u}{\partial x^2}(x_j, t_n)(\Delta x)^2 - \frac{1}{6}\frac{\partial^3 u}{\partial x^3}(x_j, t_n)(\Delta x)^3 + \frac{1}{24}\frac{\partial^4 u}{\partial x^4}(x_j, t_n)(\Delta x)^4 + O((\Delta x)^5) \quad - \quad (4)$$

$$(3) + (4), we\ have: \quad u(x_{j+1}, t_n) + u(x_{j-1}, t_n) = 2u(x_j, t_n) + \frac{\partial^2 u}{\partial x^2}(x_j, t_n)(\Delta x)^2 + \frac{1}{12}\frac{\partial^4 u}{\partial t^4}(x_j, t_n)(\Delta x)^4 + O((\Delta x)^5)$$

$$\frac{u(x_{j+1}, t_n) - 2u(x_j, t_n) + u(x_{j-1}, t_n)}{(\Delta x)^2} = \frac{\partial^2 u}{\partial x^2}(x_j, t_n) + \frac{1}{12}\frac{\partial^4 u}{\partial t^4}(x_j, t_n)(\Delta x)^2 + O((\Delta x)^3)$$

Hence,

$$T(x_j, t_n) = \frac{u(x_j, t_{n+1}) - u(x_j, t_{n-1})}{2\Delta t} - \frac{u(x_{j+1}, t_n) - 2u(x_j, t_n) + u(x_{j-1}, t_n)}{(\Delta x)^2}$$

$$= \frac{\partial u}{\partial t}(x_j, t_n) + \frac{1}{6}\frac{\partial^3 u}{\partial t^3}(x_j, t_n)(\Delta t)^2 + O((\Delta t)^3) - \frac{\partial^2 u}{\partial x^2}(x_j, t_n) + \frac{1}{12}\frac{\partial^4 u}{\partial t^4}(x_j, t_n)(\Delta x)^2 + O((\Delta x)^3)$$

Since $u(x_j, t_n)$ is the exact solution, $\frac{\partial u}{\partial t}(x_j, t_n) = \frac{\partial^2 u}{\partial x^2}(x_j, t_n)$

$T(x_j, t_n) = \frac{1}{6}\frac{\partial^3 u}{\partial t^3}(x_j, t_n)(\Delta t)^2 + \frac{1}{12}\frac{\partial^4 u}{\partial t^4}(x_j, t_n)(\Delta x)^2 + O((\Delta t)^3) + O((\Delta x)^3)$, which is second order both in time and space.

Q2. Matlab Code:

```
format long
syms x
f = exp(-x^2);
n = [5,10,50,100,500];
integrals = zeros(1,length(n));
for i = 1:length(n)
   h = 1/n(i);
   for j = 1:n(i)
      integrals(i) = integrals(i)+(vpa(subs(f,x,(j-1)*h))+vpa(subs(f,x,j*h)))*h/2;
   end;
end;
integrals
```

Result:

```
integrals =

   0.744368339763667   0.746210796131749   0.746799607189351   0.746818001467970   0.746823887559433
```

Q3(1). $a_m = 2\int_0^1 u_0(x)\sin m\pi x\,dx = 2\int_0^{\frac{1}{2}} 2x\sin m\pi x\,dx + 2\int_{\frac{1}{2}}^1 (2-2x)\sin m\pi x\,dx$

$$= 4\int_0^{\frac{1}{2}} x\sin m\pi x\,dx + 4\int_{\frac{1}{2}}^1 \sin m\pi x\,dx - 4\int_{\frac{1}{2}}^1 x\sin m\pi x\,dx$$

$$= \frac{4}{(m\pi)^2}\int_0^{\frac{m\pi}{2}} y\sin y\,dy + \frac{4}{m\pi}\int_{\frac{m\pi}{2}}^{m\pi} \sin y\,dy - \frac{4}{(m\pi)^2}\int_{\frac{m\pi}{2}}^{m\pi} y\sin y\,dy$$

$$= -\frac{4}{(m\pi)^2}\int_0^{\frac{m\pi}{2}} y d\cos y - \frac{4}{m\pi}[\cos y]_{\frac{m\pi}{2}}^{m\pi} + \frac{4}{(m\pi)^2}\int_{\frac{m\pi}{2}}^{m\pi} y d\cos y$$

$$= -\frac{4}{(m\pi)^2}\left([y\cos y]_0^{\frac{m\pi}{2}} - \int_0^{\frac{m\pi}{2}} \cos y\,dy\right) - \frac{4}{m\pi}\left(\cos m\pi - \cos\frac{m\pi}{2}\right) + \frac{4}{(m\pi)^2}\left([y\cos y]_{\frac{m\pi}{2}}^{m\pi} - \int_{\frac{m\pi}{2}}^{m\pi} \cos y\,dy\right)$$

$$= -\frac{4}{(m\pi)^2}\left([y\cos y]_0^{\frac{m\pi}{2}} - [\sin y]_0^{\frac{m\pi}{2}}\right) - \frac{4}{m\pi}\left(\cos m\pi - \cos\frac{m\pi}{2}\right) + \frac{4}{(m\pi)^2}\left([y\cos y]_{\frac{m\pi}{2}}^{m\pi} - [\sin y]_{\frac{m\pi}{2}}^{m\pi}\right)$$

$$= -\frac{4}{(m\pi)^2}\left(\frac{m\pi}{2}\cos\frac{m\pi}{2} - \sin\frac{m\pi}{2}\right) - \frac{4}{m\pi}\left(\cos m\pi - \cos\frac{m\pi}{2}\right) + \frac{4}{(m\pi)^2}\left(m\pi\cos m\pi - \frac{m\pi}{2}\cos\frac{m\pi}{2} - \sin m\pi + \sin\frac{m\pi}{2}\right)$$

$$= \frac{4}{(m\pi)^2}\left(2\sin\frac{m\pi}{2} + \sin m\pi\right)$$

$$= \frac{8}{(m\pi)^2}\sin\frac{m\pi}{2}$$

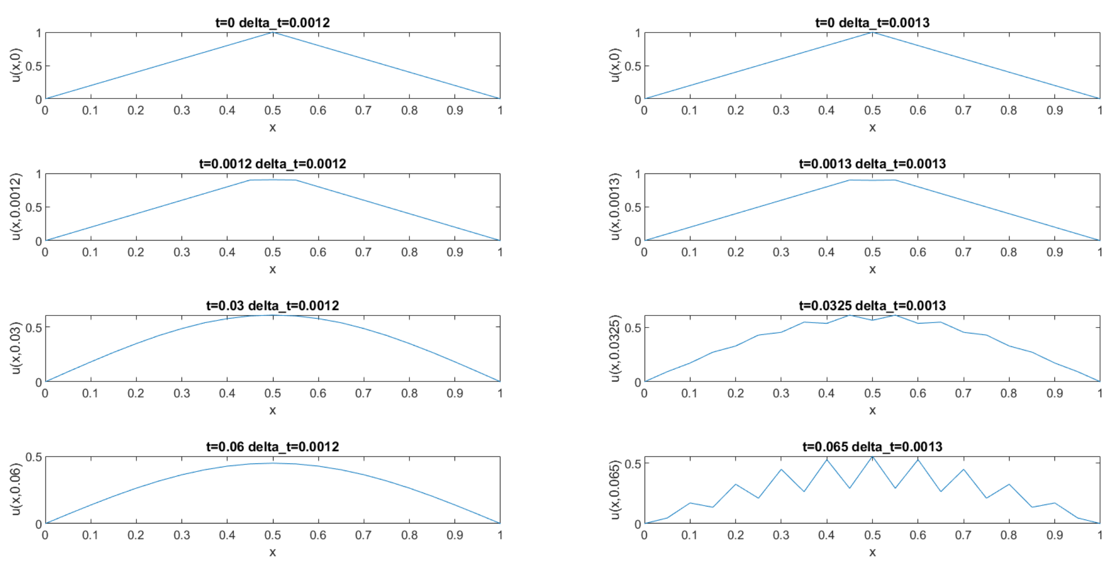
$$= \begin{cases} \frac{8}{(m\pi)^2} & m = 4k+1 \\ 0 & m\ is\ even \\ \frac{-8}{(m\pi)^2} & m = 4k+3 \end{cases}$$

Q3(2). Matlab Code:

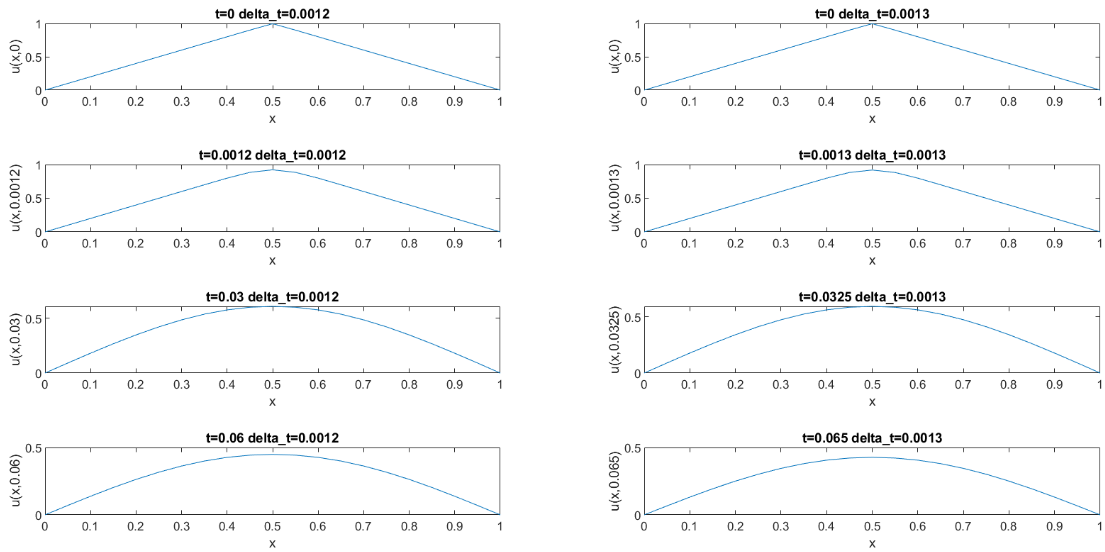
```
clear all;
J=20;
delta_x=1/J;
delta_t=[0.0012,0.0013];
k=[0,1,25,50];
epsilon=1e-3;
% numerical solution
for i=1:length(delta_t)
   mew=delta_t(i)/delta_x^2;
   n=0;
   stop=false;
   for j=1:J+1
      x=(j-1)*delta_x;
      if x<=1/2
         U(i,n+1,j)=2*x;
      else
         U(i,n+1,j)=2-2*x;
      end;
   end;
   while (~stop & n<100)
      n=n+1;
      for j=1:J+1
         x=(j-1)*delta_x;
         if (x==0) | (x==1)
            U(i,n+1,j)=0;
         else
            U(i,n+1,j)=U(i,n,j)+mew*(U(i,n,j+1)-2*U(i,n,j)+U(i,n,j-1));
         end;
      end;
      if max(abs(U(i,n+1,:)-U(i,n,:)))<epsilon
         stop=true;
      end;
   end;
end;
figure(1)
for mm=1:2
   for nn=1:length(k)
      subplot(length(k),2,mm+(nn-1)*2);
      plot((0:20)*delta_x,squeeze(U(mm,k(nn)+1,:)));
      title('t='+string(k(nn)*delta_t(mm))+' delta\_t='+string(delta_t(mm)));
      xlabel('x');
      ylabel('u(x,'+string(k(nn)*delta_t(mm))+')');
   end;
end;
% analytical solution
for i=1:length(delta_t)
   n=0;
```

```
    while n<100
        for ii=1:J+1
            m=1;
            UU(i,n+1,ii)=0;
            while m<=100
                am=8/(m*pi)^2*sin(m*pi/2);
                UU(i,n+1,ii)=UU(i,n+1,ii)+am*exp(-(m*pi)^2*n*delta_t(i))*sin(m*pi*(ii-1)*delta_x);
                m=m+1;
            end;
        end;
        n=n+1;
    end;
end;
figure(2)
for mm=1:2
    for nn=1:length(k)
        subplot(length(k),2,mm+(nn-1)*2);
        plot((0:20)*delta_x,squeeze(UU(mm,k(nn)+1,:)));
        title('t='+string(k(nn)*delta_t(mm))+' delta\_t='+string(delta_t(mm)));
        xlabel('x');
        ylabel('u(x,'+string(k(nn)*delta_t(mm))+')');
    end;
end;
```

Plot of numerical solution:



Plot of analytical solution:

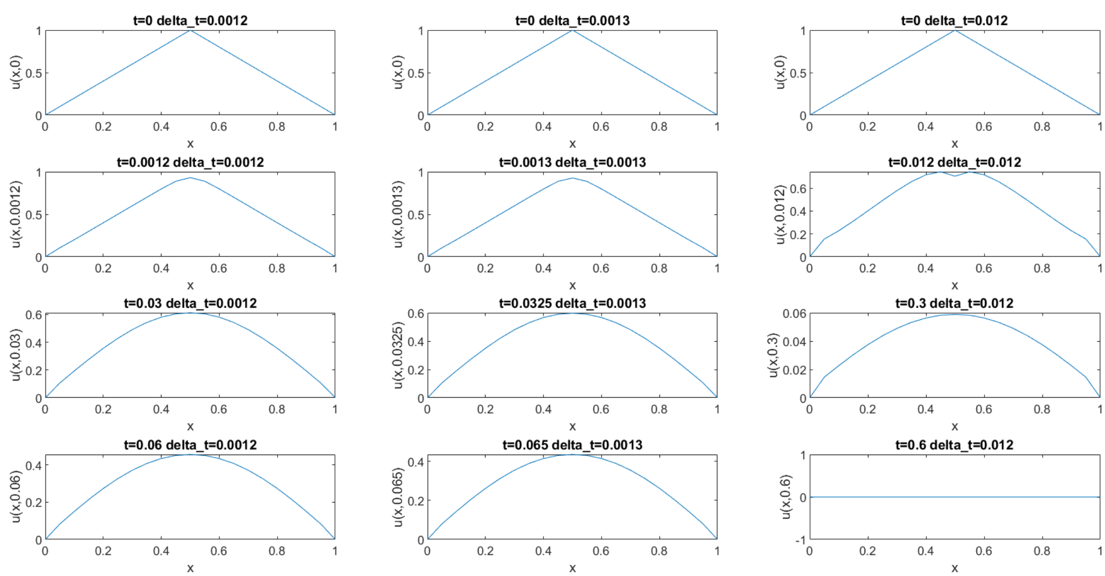


Q4. Matlab Code of Crank-Nicolson method:

```
clear all;
J=20;
delta_x=1/J;
delta_t=[0.0012,0.0013,0.012];
k=[0,1,25,50];
theta=1/2;
Nt=max(k)+1;
epsilon=1e-3;
for i=1:length(delta_t)
   mew=delta_t(i)/delta_x^2;
   d1_left(1:J)=-theta*mew;
   d2_left(1:J+1)=1+2*theta*mew;
   d3_left(1:J)=-theta*mew;
   m_left=diag(d1_left,-1)+diag(d2_left,0)+diag(d3_left,1);
   d1_right(1:J)=(1-theta)*mew;
   d2_right(1:J+1)=1-2*(1-theta)*mew;
   d3_right(1:J)=(1-theta)*mew;
   m_right=diag(d1_right,-1)+diag(d2_right,0)+diag(d3_right,1);
   n=0;
   stop=false;
   for j=1:J+1
      x=(j-1)*delta_x;
      if x<=1/2
         U(i,n+1,j)=2*x;
      else
         U(i,n+1,j)=2-2*x;
      end;
```

```
    end;
    while (~stop & n<100)
        n=n+1;
        U(i,n+1,:)=inv(m_left)*m_right*squeeze(U(i,n,:));
        U(i,n+1,1)=0;
        U(i,n+1,J+1)=0;
        if max(abs(U(i,n+1,:)-U(i,n,:)))<epsilon
            stop=true;
        end;
    end;
end;
for mm=1:3
    for nn=1:length(k)
        subplot(length(k),3,mm+(nn-1)*3);
        plot((0:20)*delta_x,squeeze(U(mm,k(nn)+1,:)));
        title('t='+string(k(nn)*delta_t(mm))+' delta\_t='+string(delta_t(mm)));
        xlabel('x');
        ylabel('u(x,'+string(k(nn)*delta_t(mm))+')');
    end;
end;
```

Plots:



Q5. $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 2(x^2 + y^2 - 2), \quad -1 \le x \le 1, \quad -1 \le y \le 1, \quad u(\pm 1, y) = u(x, \pm 1) = 0$

Using the five-point scheme with a uniform square grid of size 0.5, we have $\Delta x = \Delta y = 0.5$ and the following linear system:

$$\begin{pmatrix} -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -4 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -4 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & -4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 \end{pmatrix} \begin{pmatrix} U_{-0.5,-0.5} \\ U_{-0.5,0} \\ U_{-0.5,0.5} \\ U_{0,-0.5} \\ U_{0,0} \\ U_{0,0.5} \\ U_{0.5,-0.5} \\ U_{0.5,0} \\ U_{0.5,0.5} \end{pmatrix} = \begin{pmatrix} 2((-0.5)^2 + (-0.5)^2 - 2)(0.5)^2 \\ 2((-0.5)^2 + (0)^2 - 2)(0.5)^2 \\ 2((-0.5)^2 + (0.5)^2 - 2)(0.5)^2 \\ 2((0)^2 + (-0.5)^2 - 2)(0.5)^2 \\ 2((0)^2 + (0)^2 - 2)(0.5)^2 \\ 2((0)^2 + (0.5)^2 - 2)(0.5)^2 \\ 2((0.5)^2 + (-0.5)^2 - 2)(0.5)^2 \\ 2((0.5)^2 + (0)^2 - 2)(0.5)^2 \\ 2((0.5)^2 + (0.5)^2 - 2)(0.5)^2 \end{pmatrix} = \begin{pmatrix} -0.75 \\ -0.875 \\ -0.75 \\ -0.875 \\ -1 \\ -0.875 \\ -0.75 \\ -0.875 \\ -0.75 \end{pmatrix}$$

Matlab Code of Jacobi Iteration method:

```
d1(1:6)=1;
d2=[1,1,0,1,1,0,1,1];
d3(1:9)=-4;
A=diag(d1,-3)+diag(d1,3)+diag(d2,-1)+diag(d2,1)+diag(d3,0);
D=diag(c,0);
b=[-0.75,-0.875,-0.75,-0.875,-1,-0.875,-0.75,-0.875,-0.75]';
x0=zeros(9,1);
epsilon=1e-5;
stop=false;
n=0;
while ~stop
   n=n+1;
   x1=inv(D)*(A-D)*x0+inv(D)*b;
   if max(abs(x1-x0))<epsilon
      stop=true;
   end;
   x0=x1;
end;
x0
n
```

After 31 iterations, we obtain:

$$\begin{pmatrix} U_{-0.5,-0.5} \\ U_{-0.5,0} \\ U_{-0.5,0.5} \\ U_{0,-0.5} \\ U_{0,0} \\ U_{0,0.5} \\ U_{0.5,-0.5} \\ U_{0.5,0} \\ U_{0.5,0.5} \end{pmatrix} = \begin{pmatrix} 0.1250 \\ 0.1250 \\ 0.1250 \\ 0.1250 \\ 0.1250 \\ 0.1250 \\ 0.1250 \\ 0.1250 \\ 0.1250 \end{pmatrix}$$

```
x0 =

    0.1250
    0.1250
    0.1250
    0.1250
    0.1250
    0.1250
    0.1250
    0.1250
    0.1250


n =

    31
```