Homework 3 - MSSC 6030: Spring 2020

Directions. All work is to be done in *complete sentences*. Assignments must be stapled with a printout of the assignment serving as the first page. Your name is to be written on the *back* of the final page of the assignment. Each problem must be on a *separate* sheet of paper. You are welcome to recycle paper, where one side is crossed out to avoid wasting paper, but your work MUST have **no more than one problem per page**. Each problem write-up must begin with the **full statement of the problem**. While you are encouraged to work through confusion with your classmates, your work must be written in your own words. **The assignment is due in dropbox on Wednesday, April 1, 2020 by 3:15pm.**

---

1. Use the method of Separation of Variables to find the solution to the IBVP below. Graph the solution look like for various values of time using MATLAB.

$$\begin{array}{ll} \text{PDE} & u_t = u_{xx}, \qquad 0 < x < 1, \quad 0 < t < \infty \\[1em] \text{BCs} & \begin{cases} u(0,t) = 0 \\ u(1,t) = 0 \qquad 0 < t < \infty \end{cases} \\[1em] \text{IC} & u(x,0) = \sin(2\pi x) + \frac{1}{3}\sin(4\pi x) + \frac{1}{5}\sin(6\pi x), \qquad 0 \le x \le 1 \end{array}$$

---

2. Repeat problem #1 above now with the IC

$$u(x,0) = \phi(x) = 1 \qquad 0 \le x \le 1$$

Plot the solution for various times t with increasing numbers of terms in your sum as well.

---

3. What is the solution to the vibrating string problem below

$$\begin{array}{ll} \text{PDE} & u_{tt} = \alpha^2 u_{xx}, \qquad 0 < x < L, \quad 0 < t < \infty \\[1em] \text{BCs} & \begin{cases} u(0,t) = 0 \\ u(L,t) = 0 \qquad 0 < t < \infty \end{cases} \\[1em] \text{IC} & \begin{cases} u(x,0) = 0 \\ u_t(x,0) = \sin\left(\frac{3\pi x}{L}\right), \qquad 0 \le x \le 1 \end{cases} \end{array}$$

Letting $\alpha = 1$ and $L = 1$, what does the graph of the solution look like for various values of time? Plot it in MATLAB.

---

4. Use 2D Finite Differences to solve the following BVP

$$u_{xx} + u_{yy} = 0, \qquad 0 < x < 1, \;\; 0 < y < 1$$

subject to $u(x,y) = 0$ on the top, left, and ride sides of the square domain with $u(x,y) = sin(\pi x)$ for $y = 0$ (i.e. the bottom of the square). Use 5 grid points (3 interior points) in each of the x and y directions. Code up your FD method into Matlab and plot the solution. Does your FD solution improve with a finer grid? Is it possible to use too many points? Discuss. Think about how you could compute the 'true' analytic solution.

---

5. Use 2D Finite Differences to solve the following BVP

$$u_{xx} + u_{yy} + 2u = 0, \qquad 0 < x < 1,\ 0 < y < 1$$

subject to the boundary condition $u(x, y) = \sin((x + y)\pi)$ on the boundary. Use 6 grid points (4 interior points) in each of the x and y directions. Code up your FD method into Matlab and plot the solution. Test how your solution changes with the grid size.

---

6. Find the finite-difference solution of the heat-conduction problem

$$\begin{array}{llcll} \text{PDE:} & u_t & = & u_{xx}, & 0 < x < 1, \quad t > 0 \\ \text{BCs:} & u(0,t) & = & 0 & t > 0 \\ & u(1,t) & = & 0 & t > 0 \\ \text{IC:} & u(x,0) & = & \sin(\pi x), & 0 \le x \le 1 \end{array}$$

for $t = 0.005,\ 0.010,\ 0.015$ by the *explicit method*. Let $h = \delta x = 0.1$. Plot the solution at $x = 0, 0.1, 0.2, 0.3, \ldots, 0.9, 1$ for $t = 0.015$.

---

7. Solve the following problem analytically (separation of variables) and evaluate the analytical solution at the grid points: $x = 0, 0.1, 0.2, \ldots, 0.9, 1$ for $t = 0.015$. Compare these results to your numerical solution in #6 above.

$$\begin{array}{llcll} \text{PDE:} & u_t & = & u_{xx}, & 0 < x < 1, \quad t > 0 \\ \text{BCs:} & u(0,t) & = & 0 & t > 0 \\ & u(1,t) & = & 0 & t > 0 \\ \text{IC:} & u(x,0) & = & \sin(\pi x), & 0 \le x \le 1. \end{array}$$

---

8. Consider the problem:

$$\begin{array}{llcll} \text{PDE:} & u_t & = & u_{xx}, & 0 < x < 1, \quad t > 0 \\ \text{BCs:} & u(0,t) & = & 0 & t > 0 \\ & u(1,t) & = & 0 & t > 0 \\ \text{IC:} & u(x,0) & = & \sin(\pi x), & 0 \le x \le 1 \end{array}$$

Solve this problem using the method described in class *(Implicit FD method)* using various values of lambda including $\lambda = 0, 1/4, 1/2, 3/4, 1$ and experiment with step sizes in x and t to check accuracy. Remember that if you use $\lambda = 0$ there are specific guidelines about how small the ratio of $\frac{k}{h^2}$ must be. Compare your results to the previous two problems (#6 and #7).

Plot your solutions for various times (up to at least $t = 0.05$) and compare to the true solution. You may find it instructive to also plot the error between the true and approximate solutions.

Print out your coefficient matrix as well as the RHS vector for a small number of grid points to make sure it looks like you think it does.

1

Use the method of Separation of Variables to find the solution to the IBVP below. Graph the solution look like for various values of time using Matlab.

PDE $\qquad u_t = u_{xx} \qquad 0 < x < 1, \quad 0 < t < \infty$

BCs $\qquad \begin{cases} u(0,1) = 0 \\ u(1,t) = 0 \quad 0 < t < \infty \end{cases}$

IC $\qquad u(x,0) = \sin(2\pi x) + \frac{1}{3}\sin(4\pi x) + \frac{1}{5}\sin(6\pi x) \qquad 0 \le x \le 1$

Let $u(x,t) = X(x)T(t)$, then the PDE become:

$$\frac{\partial}{\partial t}[X(x)T(t)] = \frac{\partial^2}{\partial x^2}[X(x)T(t)]$$

$$X(x)T'(t) = X''(x)T(t)$$

Then divide both side $X(x)T(t)$

$$\frac{X(x)T'(t)}{X(x)T(t)} = \frac{X''(x)T(t)}{X(x)T(t)}$$

$$\frac{T'(t)}{T(t)} = \frac{X''(x)}{X(x)}$$

The only result the equation satified is both RHS and LHS equals to constant k.

$$\frac{X''(x)}{X(x)} = \frac{T'(t)}{T(t)} = k$$

Then

$$\begin{cases} \dfrac{X''(x)}{X(x)} = k \quad (1) \\ \dfrac{T'(t)}{T(t)} = k \quad (2) \end{cases}$$

The solution of (2) is $T(t) = T(0)e^{kt}$ (3)

With equation (1)

$X''(x) - kX(x) = 0$ (4)

Guess with the solution $X(x) = e^{rx}$

Plug the solution into (4):

$r^2e^{rx} - ke^{rx} = 0$
$e^{rx}(r^2 - k) = 0$

so $r^2 - k = 0$

$r^2 = k$

$r = \pm\sqrt{k}$

Then with sign of k, there are three case.

Case 1 $K > 0$

let $K = \lambda^2$

$r = \pm\lambda$

So the solution is:

$X(x) = Ae^{\lambda x} + Be^{-\lambda x}$

Case 2 $K < 0$

let $K = -\lambda^2$

$\frac{X(x)}{X''(x)} = -\lambda^2$

$X(x) = -\lambda^2 X''(x)$

$X(x) = A\sin(\lambda x) + B\cos(\lambda x)$

Case 3 $K = 0$

$r = 0$

$X = 1$

By plug BCs in:

BCs $\begin{cases} u(0,1) = 0 \\ u(1,t) = 0 \quad 0 < t < \infty \end{cases}$

With Case 1 $K > 0, K = \lambda^2$

$u(x,t) = X(x)T(t) = (\mathrm{Ae}^{\lambda x} + \mathrm{Be}^{-\lambda x})T(0)e^{\lambda t}$

By plug $u(0,1) = 0$ and $u(1,t) = 0$ in

$u(0,1) = (\mathrm{Ae}^{\lambda 0} + \mathrm{Be}^{-\lambda 0})T(0)e^{\lambda} = (A+B)T(0)e^{\lambda} = 0$

$u(1,t) = (\mathrm{Ae}^{\lambda} + \mathrm{Be}^{-\lambda})T(0)e^{\lambda t} = 0$

So $T(0) = 0$

Therefore $(\mathrm{Ae}^{\lambda x} + \mathrm{Be}^{-\lambda x})T(0)e^{\lambda t}$=0 in all cituation.

So Case 1 $K > 0$ is not true

Case 2 $K < 0, K = -\lambda^2$

$u(x,t) = X(x)T(t) = (\mathrm{Asin}(\lambda x) + \mathrm{Bcos}(\lambda x))T(0)e^{-\lambda^2 t}$

By plug $u(0,1) = 0$ and $u(1,t) = 0$ in

$u(0,1) = (\mathrm{Asin}(\lambda 0) + \mathrm{Bcos}(\lambda 0))T(0)e^{-\lambda^2} = \mathrm{BT}(0)e^{-\lambda^2} = 0$

$u(1,t) = (\mathrm{Asin}(\lambda) + \mathrm{Bcos}(\lambda))T(0)e^{-\lambda^2 t} = 0$

Since $T(0)$ cannot be 0 from Case 1, $\mathrm{B} = 0$.

$u(x,t) = X(x)T(t) = \mathrm{Asin}(\lambda x)T(0)e^{-\lambda^2 t}$

$u(1,t) = \mathrm{Asin}(\lambda)T(0)e^{-\lambda^2 t} = 0$

$\mathrm{Asin}(\lambda) = 0$

$\lambda = n\pi$

Case 3 $K = 0$

$u(x,t) = X(x)T(t) = T(0)e^{\mathrm{kt}}$

$T(0) = 0$ if we plug $u(0,1) = 0$ in.

Therefore Case 2 $K < 0$ is true

And $u(x,t) = X(x)T(t) = \sum_{n=1}^{\infty} A_n \sin(n\pi x) C_n e^{-(n\pi)^2 t}$

let $A_n = \tilde{A}_n C_n$

$$u(x,t) = \sum_{n=1}^{\infty} \tilde{A}_n \sin(n\pi x) e^{-(n\pi)^2 t}$$

Then plug in IC

IC: $u(x,0) = \sin(2\pi x) + \frac{1}{3}\sin(4\pi x) + \frac{1}{5}\sin(6\pi x)$

$$u(x,0) = \sum_{n=1}^{\infty} \tilde{A}_n \sin(n\pi x) = \sin(2\pi x) + \frac{1}{3}\sin(4\pi x) + \frac{1}{5}\sin(6\pi x)$$

Multiply $\sin(m\pi x)$ on both side and then integral, m=1,2,3,...

$$\left\langle \sum_{n=1}^{\infty} \tilde{A}_n \sin(n\pi x), \sin(m\pi x) \right\rangle = \left\langle \sin(2\pi x) + \frac{1}{3}\sin(4\pi x) + \frac{1}{5}\sin(6\pi x), \sin(m\pi x) \right\rangle$$

$$\text{LHS} = \sum_{n=1}^{\infty} \tilde{A}_n \int_0^1 \sin(n\pi x)\sin(m\pi x)\mathrm{dx} = \begin{cases} 0 & \text{if } m \neq n \\ \frac{1}{2} & \text{if } m = n \end{cases}$$

$$\text{RHS} = \int_0^1 \left(\sin(2\pi x) + \frac{1}{3}\sin(4\pi x) + \frac{1}{5}\sin(6\pi x)\right)\sin(m\pi x)\mathrm{dx}$$

$$\text{RHS} = \int_0^1 \sin(2\pi x)\sin(m\pi x)\mathrm{dx} + \int_0^1 \frac{1}{3}\sin(4\pi x)\sin(m\pi x)\mathrm{dx} + \int_0^1 \frac{1}{5}\sin(6\pi x)\sin(m\pi x)\mathrm{dx}$$

$$\int_0^1 \sin(2\pi x)\sin(m\pi x)\mathrm{dx} = \frac{\tilde{A}_m}{2}$$

$$\int_0^1 \frac{1}{3}\sin(4\pi x)\sin(m\pi x)\mathrm{dx} = \frac{\tilde{A}_m}{2}$$

$$\int_0^1 \frac{1}{5}\sin(6\pi x)\sin(m\pi x)\mathrm{dx} = \frac{\tilde{A}_m}{2}$$

Simplifiy

$$\int_0^1 \sin(2\pi x)\sin(2\pi x)\mathrm{dx} = \frac{\tilde{A}_2}{2} = \frac{1}{2}$$

$$\int_0^1 \frac{1}{3}\sin(4\pi x)\sin(4\pi x)\mathrm{dx} = \frac{\tilde{A}_4}{2} = \frac{1}{3}\frac{1}{2} = \frac{1}{6}$$

$$\int_0^1 \frac{1}{5}\sin(6\pi x)\sin(6\pi x)\mathrm{dx} = \frac{\tilde{A}_6}{2} = \frac{1}{5}\frac{1}{2} = \frac{1}{10}$$

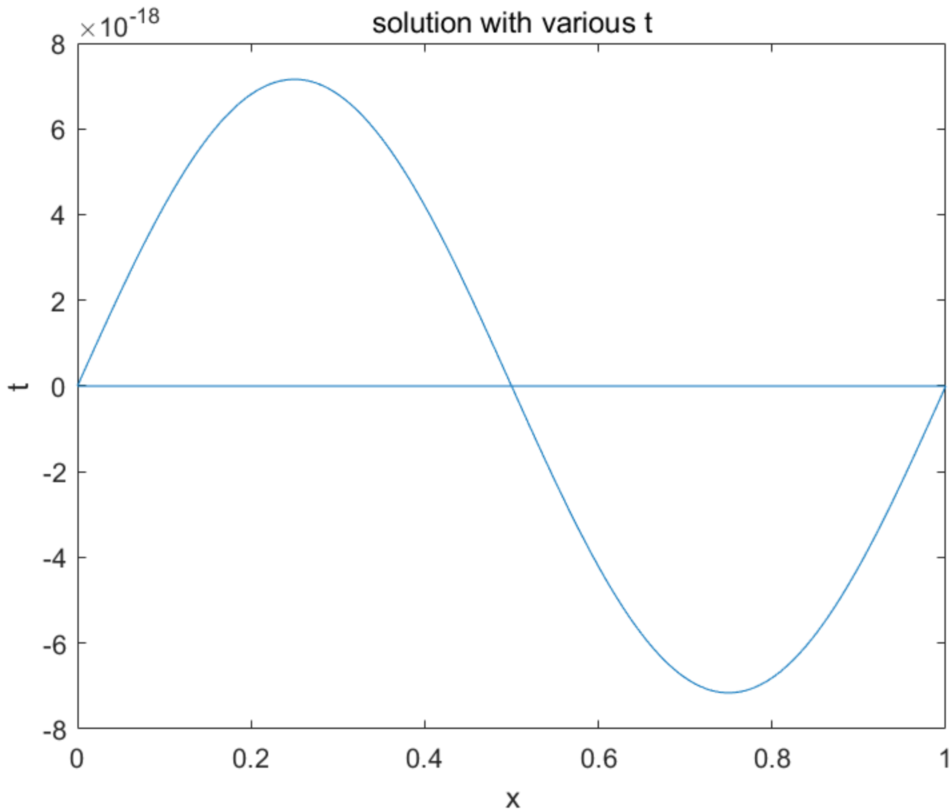
So $\tilde{A}_2 = 1$, $\tilde{A}_4 = \frac{1}{3}$, $\tilde{A}_6 = \frac{1}{5}$, and $\tilde{A}_n = 0$ with other n

So $u(x,t) = \sum_{n=1}^{\infty} \tilde{A}_n \sin(n\pi x) e^{-(n\pi)^2 t} = \sin(2\pi x) e^{-(2\pi)^2 t} + \frac{1}{3}\sin(4\pi x) e^{-(4\pi)^2 t} + \frac{1}{5}\sin(6\pi x) e^{-(6\pi)^2 t}$ (5)

$$u_t = -(2\pi)^2 \sin(2\pi x) e^{-(2\pi)^2 t} - (4\pi)^2 \frac{1}{3}\sin(4\pi x) e^{-(4\pi)^2 t} - (6\pi)^2 \frac{1}{5}\sin(6\pi x) e^{-(6\pi)^2 t}$$

$$= -4\pi^2 \sin(2\pi x) e^{-(2\pi)^2 t} - 16\pi^2 \frac{1}{3}\sin(4\pi x) e^{-(4\pi)^2 t} - 36\pi^2 \frac{1}{5}\sin(6\pi x) e^{-(6\pi)^2 t}$$

```
clear
x=linspace(0,1,500);%generate x according to PDE interval (0,1)

for t=1:50
    %loop for t
    plot(x,u(x,t))%plot with each t
    hold on
end

title("solution with various t")
xlabel("x")
ylabel("u(x,t)")
```




```
function [uf]=u(x,t)
% equation(5)
    uf=sin(2*pi*x).*exp(-4*pi^2*t) ...
```

```
    +1/3*sin(4*pi*x).*exp(-16*pi^2*t) ...
    +1/5*sin(6*pi*x).*exp(-36*pi^2*t);
end
```

2

$u(x,0)=\phi(x)=1$

The solution from last question:

$$u(x,t)=\sum_{n=1}^{\infty}\tilde{A}_n\sin(n\pi x)e^{-(n\pi)^2t}$$

Plug IC:

$$u(x,0)=\sum_{n=1}^{\infty}\tilde{A}_n\sin(n\pi x)=1$$

$$\left\langle\sum_{n=1}^{\infty}\tilde{A}_n\sin(n\pi x),\sin(m\pi x)\right\rangle=\langle\phi(x),\sin(m\pi x)\rangle$$

$$\text{LHS}=\sum_{n=1}^{\infty}\tilde{A}_n\int_0^1\sin(n\pi x)\sin(m\pi x)\mathrm{dx}=\begin{cases}0 & \text{if } m\neq n\\ \frac{1}{2} & \text{if } m=n\end{cases}$$

$$\text{RHS}=\int_0^1\phi(x)\sin(m\pi x)\mathrm{dx}=\frac{\tilde{A}_n}{2}$$

$$\tilde{A}_m=2\int_0^1\phi(x)\sin(m\pi x)\mathrm{dx}$$

Plug $\phi(x)=1$

$$\tilde{A}_m=2\int_0^1\sin(m\pi x)\mathrm{dx}=-\frac{2}{m\pi}\cos(m\pi x)_0^1=-\frac{2}{m\pi}(\cos(m\pi)-\cos(0))=\frac{2}{m\pi}(1-\cos(m\pi))=$$

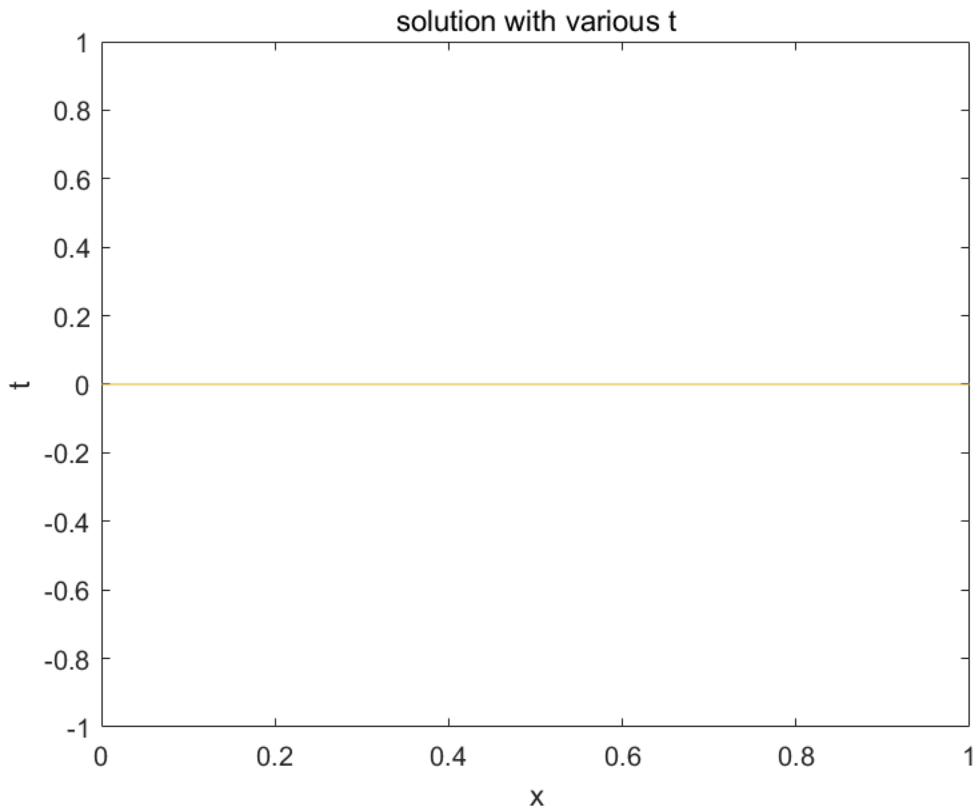
$$\tilde{A}_1=\frac{4}{\pi},\ \tilde{A}_2=0,\ \tilde{A}_3=\frac{4}{3\pi},\ \tilde{A}_4=0,\ \tilde{A}_5=\frac{4}{5\pi},$$

$$u(x,t)=\sum_{n=1}^{\infty}\frac{2}{n\pi}(1-(-1)^n)\sin(n\pi x)e^{-(n\pi)^2t}=\frac{4}{\pi}\sin(\pi x)e^{-\pi^2t}+\frac{4}{3\pi}\sin(3\pi x)e^{-9\pi^2t}+\frac{4}{5\pi}\sin(5\pi x)e^{-25\pi^2t}+\ldots$$

$$u(x,t)=\sum_{n=1}^{\infty}\frac{4}{(2n-1)\pi}\sin((2n-1)\pi x)e^{-((2n-1)\pi)^2t}\qquad(6)$$

```
clear
n=50;%set n=50
x=linspace(0,1,500);%generate x

for t=1:500
    %loop for t
    plot(x,u(x,t,n))%plot with each t
    hold on
end
title("solution with various t")
xlabel("x")
```

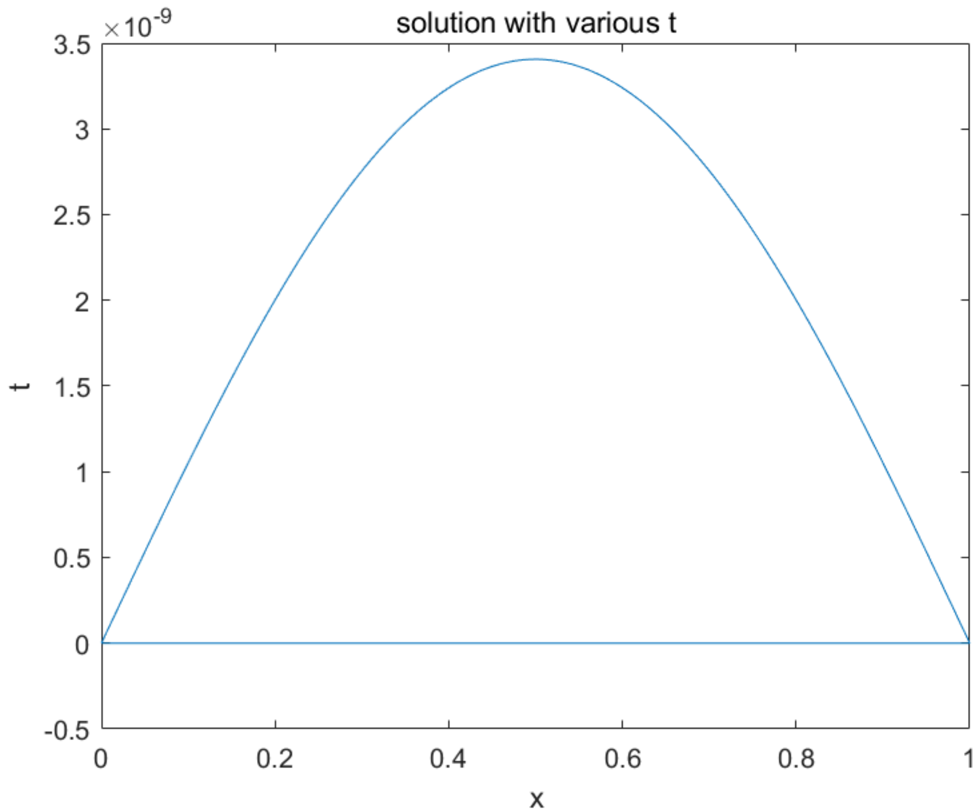
```
ylabel("u(x,t)")
hold off
```



From the graph we could see the solution will not change with t so t can be any number.

So we set t=2

```
t=2;%set t=2
for n=1:50
    %loop for n
    plot(x,u(x,t,n))%plot with each n
    hold on
end
title("solution with various n")
xlabel("x")
ylabel("u(x,t)")
```




```
function [s]=u(x,t,n)
%equation (6)
    s=0;%initialize the sum
    for j = 1:n
        %loop of n
        s=s+4/((2*n-1)*pi)*sin((2*n-1)*x*pi).*exp(-((2*n-1)*pi)^2*t);
        %add sum together
    end
end
```

3

What is the solution to the vibrating string problem below

PDE $u_{tt} = -\alpha^2 u_{xx}$ $0 < x < L, \quad 0 < t < \infty$

BCs $\begin{cases} u(0,t) = 0 \\ u(L,t) = 0 \quad 0 < t < \infty \end{cases}$

IC $\begin{cases} u(x,0) = 0 \\ u_t(x,0) = \sin\left(\frac{3\pi x}{L}\right) \quad 0 \le x \le 1 \end{cases}$

Letting $\alpha = 1$ and $L = 1$ what does the graph of the solution look like for various values of time? Plot it in Matlab

Let $u(x,t) = X(x)T(t)$, then the PDE become:

$$\frac{\partial^2}{\partial x^2}[X(x)T(t)] = -\alpha^2 \frac{\partial^2}{\partial x^2}[X(x)T(t)]$$

$$X(x)T''(t) = -\alpha^2 X''(x)T(t)$$

Then divide both side $X(x)T(t)$

$$\frac{X(x)T''(t)}{X(x)T(t)} = -\alpha^2 \frac{X''(x)T(t)}{X(x)T(t)}$$

$$\frac{T(t)}{T'(t)} = -\alpha^2 \frac{X''(x)}{X(x)}$$

$$-\frac{1}{\alpha^2}\frac{T(t)}{T''(t)} = \frac{X''(x)}{X(x)}$$

$$-\frac{1}{\alpha^2}\frac{T(t)}{T'(t)} = \frac{X''(x)}{X(x)} = k$$

Then

$$\begin{cases} \dfrac{X''(x)}{X(x)} = k & (7) \\ -\dfrac{1}{\alpha^2}\dfrac{T''(t)}{T(t)} = k & (8) \end{cases}$$

Guess the solution $y(t) = e^{rt}$

$y'' + py' + qy = 0$
$r^2 e^{rt} + pre^{rt} + qe^{rt} = 0$, which $p = 0, q = -k$
$e^{rt}(r^2 + pr + q) = 0$

since e^{rt} can not be 0

$(r^2 + pr + q) = 0$

$$r = -\frac{p \pm \sqrt{p^2 - 4q}}{2} = \pm \frac{\sqrt{-4q}}{2}$$

For the situation that one root:

$$r = -\frac{p}{2}$$

$$y = C_1 e^{rt} + tC_2 e^{rt}$$

For the situation that two roots:

$$r_{1,2} = -\frac{p \pm \sqrt{p^2 - 4q}}{2}$$

$$y = C_1 e^{r_1 t} + C_2 e^{r_2 t}$$

For the situation that complex roots:

$$r_{1,2} = a \pm bi = \pm \frac{\sqrt{-4q}}{2} = \pm \frac{2i\sqrt{q}}{2} = \pm \sqrt{q}\, i$$

$$y = C_1 e^{at}\cos(bt) + C_2 e^{at}\sin(bt)$$

Case 1 $k > 0$, $k = \lambda^2$, which is the situation that r has two root.

$$X''(x) = \lambda^2 X(x)$$

$$X(x) = C_1 e^{\lambda x} + C_2 e^{-\lambda x}$$

Plug BCs in:

BCs $\begin{cases} u(0,t) = 0 \\ u(L,t) = 0 \quad 0 < t < \infty \end{cases}$

$$X(0) = C_1 + C_2 = 0$$

$$C_1 = -C_2$$

$$X(L) = C_1 e^{\lambda L} + C_2 e^{-\lambda L} = -C_2 e^{\lambda L} + C_2 e^{-\lambda L} = C_2(e^{-\lambda L} - e^{\lambda L}) = 0$$

So either $C_2 = 0$ or $(e^{-\lambda L} - e^{\lambda L}) = 0$

If $C_2 = 0, C_1 = 0$

Which $X(x) = 0$ and then $u(x,t) = 0$

So $C_2 = 0$ is not true, $(e^{-\lambda L} - e^{\lambda L}) = 0$

$e^{-\lambda L} - e^{\lambda L} = 0,$

$e^{-\lambda L} = e^{\lambda L}$

Which also means $X(x) = 0$ and then $u(x,t) = 0$

So $k > 0$ is not true.

Case 2 $k = 0$, $k = \lambda^2 = 0$,so $\lambda = 0$

$X(x) = C_1 e^{\lambda x} + \mathrm{x} \mathrm{C}_2 e^{\lambda x} = C_1 + C_2 x$

Plug in BCs

BCs $\begin{cases} u(0,t) = 0 \\ u(L,t) = 0 \quad 0 < t < \infty \end{cases}$

$X(0) = C_1 + C_2 0 = C_1 = 0$

$X(L) = C_1 + C_2 L = C_2 L = 0$

So $C_2 = 0$

Therefor both $C_1$and C_2 equal to 0

$X(x) = 0$for any situation and $u(x,t) = 0$

So $k = 0$ is not true.

Case 3 $k < 0$, $k = -\lambda^2, r_{1,2} = \pm\sqrt{-k}\, i = \pm \lambda i$

$a = 0, b = \lambda$

$X(x) = C_1 \cos(\lambda x) + C_2 \sin(\lambda x)$

Plug in BCs

BCs $\begin{cases} u(0,t) = 0 \\ u(L,t) = 0 \quad 0 < t < \infty \end{cases}$

$X(0) = C_1\cos(0) + C_2\sin(0) = C_1 = 0$

$X(L) = C_2\sin(\lambda L) = 0$

$C_2 \neq 0$ since if $C_2 = C_1 = 0$, $X(x) = 0$ and $u(x,t) = 0$

So $\sin(\lambda L) = 0, \lambda L = n\pi, \lambda = \frac{n\pi}{L}$

$X(x) = C_n\sin\left(\frac{n\pi x}{L}\right)$

Then for $T(t), -\frac{1}{\alpha^2}\frac{T''(t)}{T(t)} = k$

$T''(t) = -\alpha^2 k T(t)$

By using $k < 0$, $k = -\lambda^2$

$r_{1,2} = \pm\sqrt{-k\alpha^2}\,i = \pm\lambda\alpha i$

$a = 0, b = \lambda\alpha = \frac{n\pi\alpha}{L}$

$T(t) = a_n\cos\left(\frac{n\pi\alpha t}{L}\right) + b_n\sin\left(\frac{n\pi\alpha t}{L}\right)$

$$u(x,t) = X(x)T(t) = X(x) = C_n\sin\left(\frac{n\pi x}{L}\right)\left[a_n\cos\left(\frac{n\pi\alpha t}{L}\right) + b_n\sin\left(\frac{n\pi\alpha t}{L}\right)\right]$$

$$= C_n\sin\left(\frac{n\pi x}{L}\right)a_n\cos\left(\frac{n\pi\alpha t}{L}\right) + C_n\sin\left(\frac{n\pi x}{L}\right)b_n\sin\left(\frac{n\pi\alpha t}{L}\right)$$

Let $C_n a_n = A_n$ and $C_n b_n = B_n$

$$u(x,t) = \sum_{n=1}^{\infty}\sin\left(\frac{n\pi x}{L}\right)\left[A_n\cos\left(\frac{n\pi\alpha t}{L}\right) + B_n\sin\left(\frac{n\pi\alpha t}{L}\right)\right]$$

Then plug IC in:

IC $\begin{cases} u(x,0) = 0 \\ u_t(x,0) = \sin\left(\frac{3\pi x}{L}\right) \quad 0 \le x \le 1 \end{cases}$

$$u(x,0) = \sum_{n=1}^{\infty}\sin\left(\frac{n\pi x}{L}\right)[A_n] = 0$$

Since $\sin\left(\frac{n\pi x}{L}\right) \neq 0, A_n = 0,$

$$u_t(x,t) = \sum_{n=1}^{\infty} \sin\left(\frac{n\pi x}{L}\right)\left[\frac{n\pi\alpha}{L}B_n\cos\left(\frac{n\pi\alpha t}{L}\right)\right]$$

$$= \sum_{n=1}^{\infty} \sin\left(\frac{n\pi x}{L}\right)\frac{n\pi\alpha}{L}\left[B_n\cos\left(\frac{n\pi\alpha t}{L}\right)\right]$$

$$u_t(x,0) = \sum_{n=1}^{\infty} \sin\left(\frac{n\pi x}{L}\right)\frac{n\pi\alpha}{L}\left[B_n\cos(0)\right] = \sin\left(\frac{3\pi x}{L}\right)$$

$$\sum_{n=1}^{\infty} \sin\left(\frac{n\pi x}{L}\right)\frac{n\pi\alpha}{L}\left[B_n\right] = \sin\left(\frac{3\pi x}{L}\right)$$

$$\frac{3\pi\alpha}{L}\left[B_3\right] = 1$$

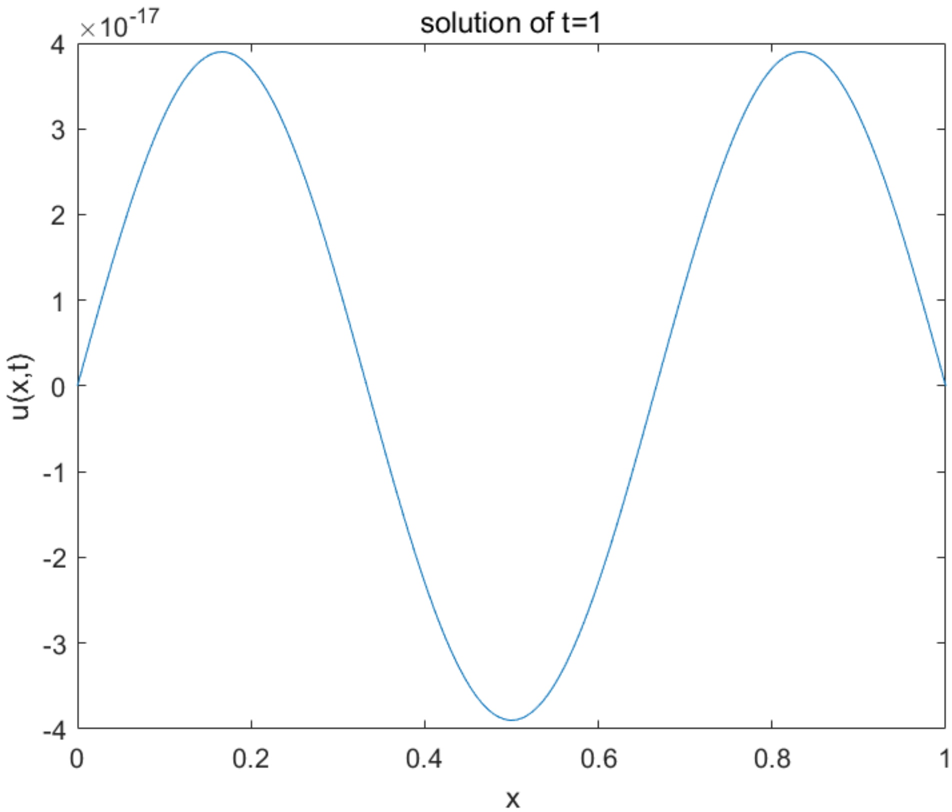
$$B_3 = \frac{L}{3\pi\alpha}$$

So $u(x,t) = \frac{L}{3\pi\alpha}\sin\left(\frac{3\pi x}{L}\right)\sin\left(\frac{3\pi\alpha t}{L}\right)$

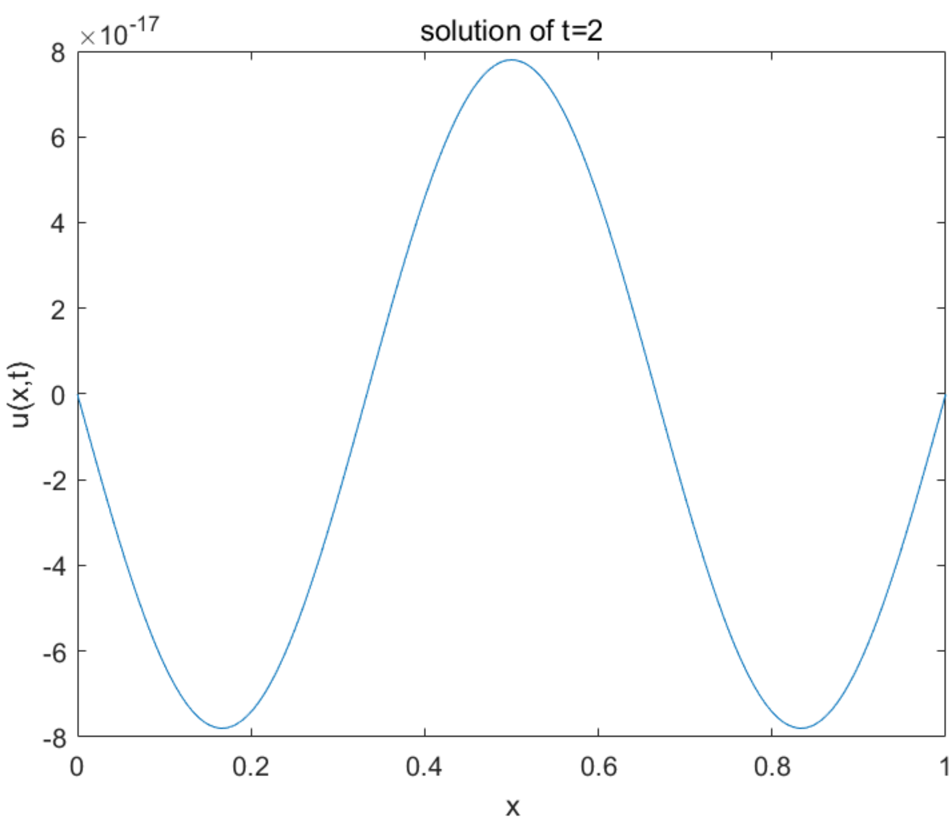
Letting $\alpha = 1$ and $L = 1$

$u(x,t) = \frac{1}{3\pi}\sin(3\pi x)\sin(3\pi t)$ (9)

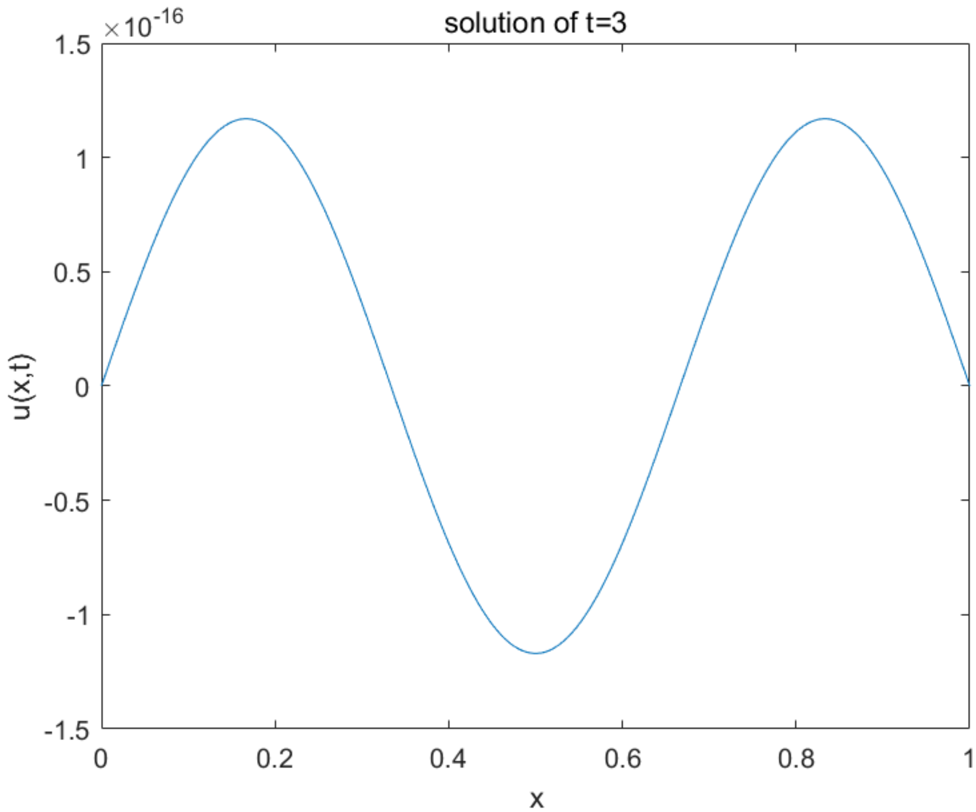
```
clear
x=linspace(0,1,500);%generate x

plot(x,u(x,1))
xlabel("x");ylabel("u(x,t)");title("solution of t=1");
```



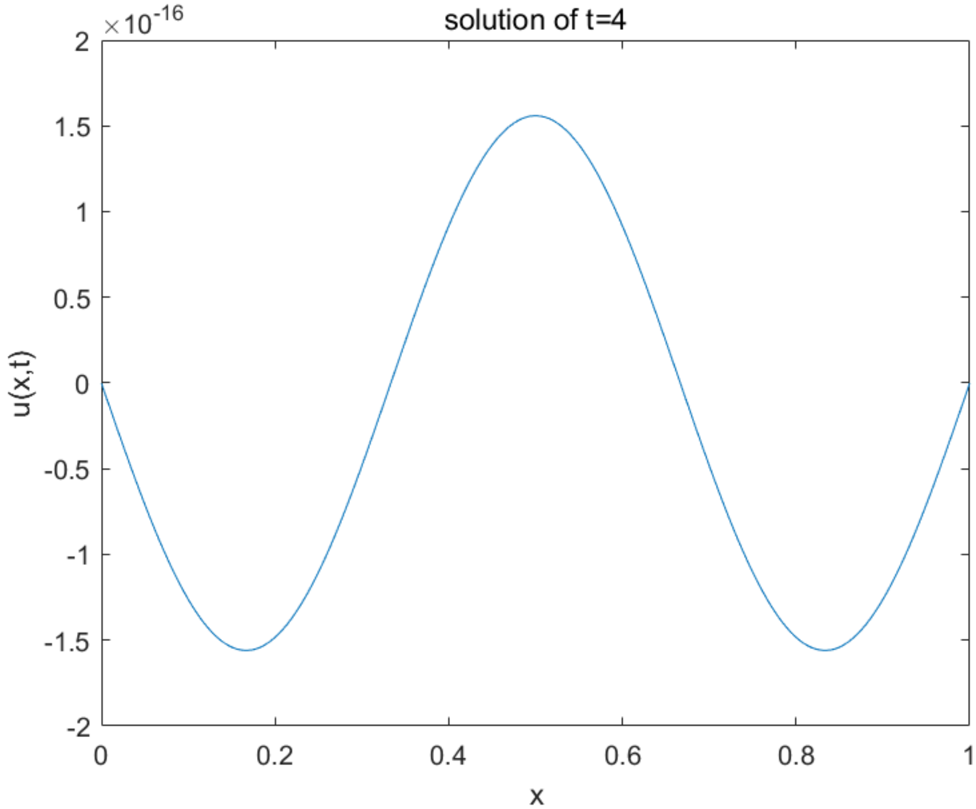

```
plot(x,u(x,2))
xlabel("x");ylabel("u(x,t)");title("solution of t=2");
```



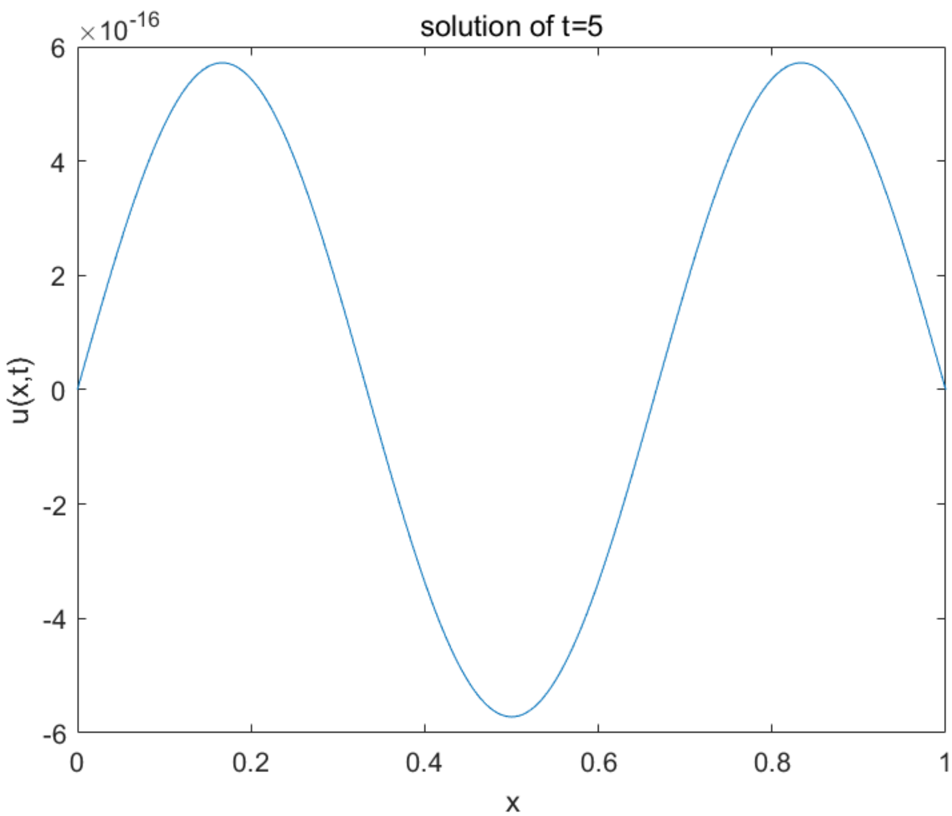
```
plot(x,u(x,3))
xlabel("x");ylabel("u(x,t)");title("solution of t=3");
```




```
plot(x,u(x,4))
xlabel("x");ylabel("u(x,t)");title("solution of t=4");
```




```
plot(x,u(x,5))
xlabel("x");ylabel("u(x,t)");title("solution of t=5");
```

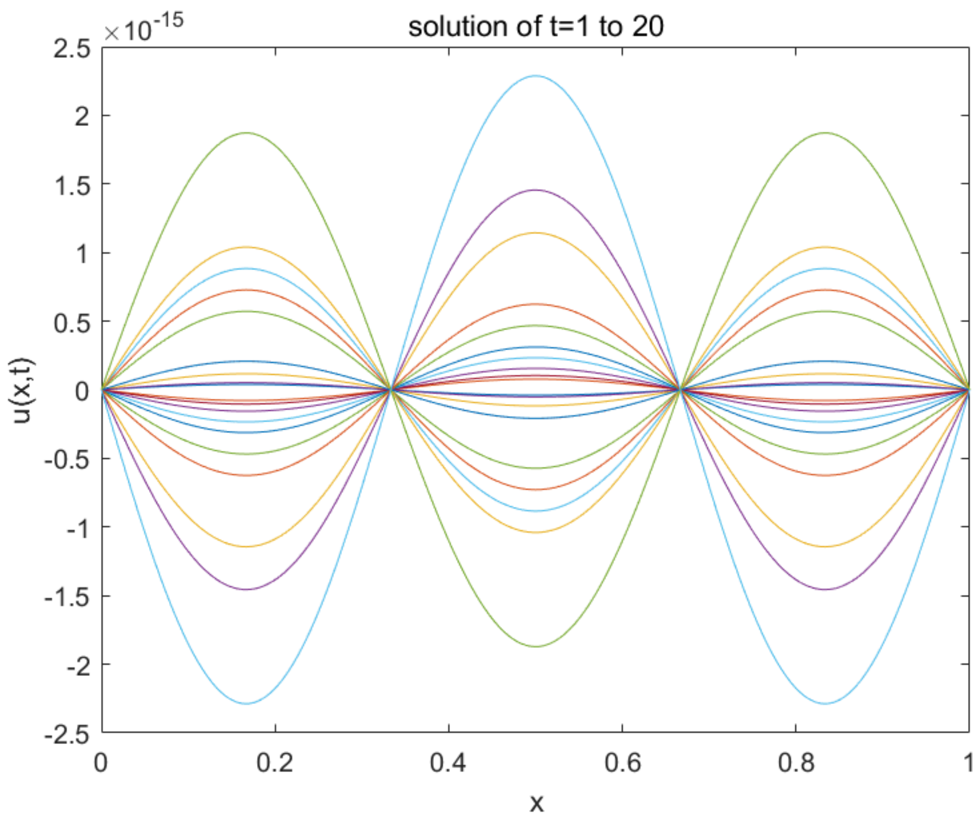


```
%plot for t=1 to 5
```

from the plot above we could see the wave vibrate around 0

So if we test for further t

```
for t=1:20
    %loop for t
    plot(x,u(x,t))%plot with each t
    hold on
end
xlabel("x");ylabel("u(x,t)");title("solution of t=1 to 20");
```




```
function [s]=u(x,t)
%equation (9)
    s=1/(3*pi)*sin(3*pi*x).*sin(3*pi*t);
end
```

4. Use 2D Finite Differences to solve the following BVP

$u_{xx} + u_{yy} = 0, \quad 0 < x < 1, 0 < y < 1$

subject to $u(x, y) = 0$ on the top, left, and ride sides of the square domain with $u(x, y) = \sin(\pi x)$ for $y = 0$ (i.e. the bottom of the square). Use 5 grid points (3 interior points) in each of the x and y directions. Code up your FD method into Matlab and plot the solution. Does your FD solution improve with a finer grid? Is it possible to use too many points? Discuss. Think about how you could compute the 'true' analytic solution

For Finite Differences Method

$$\frac{1}{h^2}(u_{i,j+1} - 2u_{i,j} + u_{i,j-1}) + \frac{1}{h^2}(u_{i+1,j} - 2u_{i,j} + u_{i+1,j}) = 0$$

$$\frac{1}{h^2}(u_{i,j+1} - 4u_{i,j} + u_{i,j-1} + u_{i+1,j} + u_{i+1,j}) = 0$$

$$\frac{u_{i,j+1}}{h^2} - \frac{4u_{i,j}}{h^2} + \frac{u_{i,j-1}}{h^2} + \frac{u_{i+1,j}}{h^2} + \frac{u_{i+1,j}}{h^2} = 0$$

$$\frac{u_{i,j+1}}{h^2} + \frac{u_{i,j-1}}{h^2} + \frac{u_{i+1,j}}{h^2} + \frac{u_{i+1,j}}{h^2} - \frac{4u_{i,j}}{h^2} = 0$$

$$\frac{u_{i,j+1}}{h^2} + \frac{u_{i,j-1}}{h^2} + \frac{u_{i+1,j}}{h^2} + \frac{u_{i+1,j}}{h^2} - \frac{4}{h^2}u_{i,j} = 0$$

$$u_{i,j+1} + u_{i,j-1} + u_{i+1,j} + u_{i+1,j} = 4u_{i,j}$$

$$\frac{1}{4}(u_{i,j+1} + u_{i,j-1} + u_{i+1,j} + u_{i+1,j}) = u_{i,j}$$

$u_{i,j} = \frac{1}{4}(u_{i,j+1} + u_{i,j-1} + u_{i+1,j} + u_{i+1,j})$ for each i,j in the interval points

So for 5 grid points (3 interior points) there will be 9 unknows

The points are known if they are on the side.

Therefore we need to a 9 by 9 matrix to solve all of those 9 point.

Which is:

V*a=b

Where V is 9 by 9, a is vector contain 9 unknown interior point, and b is the known point from BC.

If we write up the equation for $i = 2$ and $j = 2$

$$u_{2,2} = \frac{1}{4}(u_{2,3} + u_{2,1} + u_{3,2} + u_{3,2})$$

$$4u_{2,2} = u_{2,3} + u_{2,1} + u_{3,2} + u_{3,2}$$

$$4u_{2,2} - u_{2,3} - u_{2,1} - u_{3,2} - u_{3,2} = 0$$

Since $u_{2,1}$ is on the boundary

$$4u_{2,2} - u_{2,3} - u_{3,2} - u_{3,2} = u_{2,1}$$

If we arrange the unknown like:

$$a = \begin{bmatrix} u_{2,2} \\ u_{2,3} \\ u_{2,4} \\ u_{3,2} \\ u_{3,3} \\ u_{3,4} \\ u_{4,2} \\ u_{4,3} \\ u_{4,4} \end{bmatrix}$$

So the first row of V is

$$\begin{bmatrix} 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The first element of b is $u_{2,1}$

And then if we write up for the second row

$$u_{2,3} = \frac{1}{4}(u_{2,4} + u_{2,2} + u_{3,3} + u_{3,3})$$

$$4u_{2,3} - u_{2,4} - u_{2,2} - u_{3,3} - u_{3,3} = 0$$

So the second row of V is

$$\begin{bmatrix} -1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

we could see the number 4 and -1 appears lots of times.

4 is the coeffecient on the point we need to know and -1 is the coeffecient on the point left, right, up and down of it the point on the BC will move to the correspounding element in b.

If we form the above in a matrix:

$$k = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

So we can keep load the matrix k into a all zero matrix and cut off the BC point then flat the matrix left to a row.

Which will become the row for matrix V.

The method I generate the matrix V, b and calculate result is:

grid point on x=nx

grid point on x=ny

set $k = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$

step 1: from the point $(2, 2)$ to $(\mathrm{nx} - 1, \mathrm{ny} - 1)$:

step 2: build up an all 0 matrix M_z with dimension nx by ny (in this case is 5 by 5).

change the point and all the point around it (in total 9 points) to matrix k.

step 3: cut the matrix from $M_z(2, 2)$ to $M_z(\mathrm{nx} - 1, \mathrm{ny} - 1)$

step 4 flat the cut matrix to 1 row and restore is to the matrix V.

step 5: for correspomding element in b, we could just add up all the point around it from the matrix of IC.

(therefore only BC will be added since the other point is 0)

step 6: calculate a by V\b then reform a to $(\mathrm{nx} - 1)$ by $(\mathrm{ny} - 1)$. Then put reformed a back into IC matrix.

By compute true solution

PDE $\quad u_{\mathrm{xx}} = -u_{\mathrm{yy}} \qquad 0 < x < 1, \quad 0 < y < 1$

BCs $\quad \begin{cases} u(1, y) = 0 & 0 < y < 1 \\ u(x, 1) = 0 & 0 < x < 1 \end{cases}$

IC $\quad \begin{cases} u(0, y) = 0 & \\ u(x, 0) = \sin(\pi x) & 0 \le x \le 1 \end{cases}$

$u = X(x)Y(y)$

$X''(x)Y(y) = -X(x)Y''(y)$

$$\frac{X''(x)}{X(x)} = -\frac{Y''(y)}{Y(y)} = k$$

$$\frac{X''(x)}{X(x)} = k$$

$$\frac{Y''(y)}{Y(y)} = -k$$

Case 1 $k > 0$, $k = \lambda^2$,

$X''(x) = \lambda^2 X(x)$

$X(x) = C_1 e^{\lambda x} + C_2 e^{-\lambda x}$

IC $\begin{cases} u(0,y) = 0 \\ u(x,0) = \sin(\pi x) \quad 0 \le x \le 1 \end{cases}$

$X(0) = C_1 e^{\lambda 0} + C_2 e^{-\lambda 0} = C_1 + C_2 = 0$

$C_1 = -C_2$

And by BCs

BCs $\begin{cases} u(1,y) = 0 \quad 0 < y < 1 \\ u(x,1) = 0 \quad 0 < x < 1 \end{cases}$

$X(1) = C_1 e^{\lambda 1} + C_2 e^{-\lambda 1} = -C_2 e^{\lambda 1} + C_2 e^{-\lambda 1} = C_2(e^{-\lambda} - e^{\lambda}) = 0$

So either $C_2 = 0$ or $(e^{-\lambda} - e^{\lambda}) = 0$

If $C_2 = 0, C_1 = 0$

Which $X(x) = 0$ and then $u(x,y) = 0$

So $C_2 = 0$ is not true, $(e^{-\lambda} - e^{\lambda}) = 0$

$e^{-\lambda} - e^{\lambda} = 0,$

$e^{-\lambda} = e^{\lambda}$

The only λ that satisfies this equation is $\lambda = 0$

Which means $X(x) = C_1 e^{0x} + C_2 e^{-0x}$

Which also means $X(x) = 0$ and then $u(x,t) = 0$

So $k > 0$ is not true.

Case 2 $k = 0$, $k = \lambda^2 = 0$,so $\lambda = 0$

$X(x) = C_1 e^{\lambda x} + \mathrm{x}\mathrm{C}_2 e^{\lambda x} = C_1 + C_2 x$

Plug in ICs

IC $\begin{cases} u(0,y) = 0 \\ u(x,0) = \sin(\pi x) \quad 0 \le x \le 1 \end{cases}$

$X(0) = C_1 + C_2 0 = C_1 = 0$

and BCs:

BCs $\begin{cases} u(1,y) = 0 \quad 0 < y < 1 \\ u(x,1) = 0 \quad 0 < x < 1 \end{cases}$

$X(1) = C_1 + C_2 = C_2 = 0$

So $C_2 = 0$

Therefor both C_1 and C_2 equal to 0

$X(x) = 0$ for any situation and $u(x,t) = 0$

So $k = 0$ is not true.

Case 3 $k < 0$, $k = -\lambda^2, r_{1,2} = \pm\sqrt{-k}\, i = \pm\lambda i$

$a = 0, b = \lambda$

$X(x) = C_1\cos(\lambda x) + C_2\sin(\lambda x)$

Plug in ICs

IC $\begin{cases} u(0,y) = 0 \\ u(x,0) = \sin(\pi x) \quad 0 \le x \le 1 \end{cases}$

$X(0) = C_1\cos(0) + C_2\sin(0) = C_1 = 0$

and BCs:

BCs $\begin{cases} u(1,y) = 0 \quad 0 < y < 1 \\ u(x,1) = 0 \quad 0 < x < 1 \end{cases}$

$X(1) = C_2\sin(\lambda) = 0$

$\lambda = n\pi$

$X(x) = C_n\sin(n\pi x)$

Then for $Y(t), -\dfrac{Y''(y)}{Y(y)} = k$

for $k < 0$

$Y(y) = a_n e^{\lambda y} + b_n e^{-\lambda y}$

BCs $\begin{cases} u(1,y) = 0 \quad 0 < y < 1 \\ u(x,1) = 0 \quad 0 < x < 1 \end{cases}$

$Y(1) = a_n e^{\lambda} + b_n e^{-\lambda} = 0$

So

$$U(x,y) = X(x)Y(y) = \sum_{1}^{\infty} C_n \sin(n\pi x)\left[a_n e^{\lambda y} + b_n e^{-\lambda y}\right]$$

Let $C_n a_n = A_n\, C_n b_n = B_n$

$$u(x,y) = \sum_{n=1}^{\infty} \sin(n\pi x)\left[A_n e^{n\pi y} + B_n e^{-n\pi y}\right]$$

Then plug IC in:

IC $\begin{cases} u(0,y) = 0 \\ u(x,0) = \sin(\pi x) \quad 0 \le x \le 1 \end{cases}$

$$u(x,0) = \sum_{n=1}^{\infty} \sin(n\pi x)[A_n + B_n] = \sin(\pi x)$$

n only could be 1

$$A_n + B_n = 1$$
$$A_n = 1 - B_n$$

then from BCs

BCs $\begin{cases} u(1,y) = 0 \quad 0 < y < 1 \\ u(x,1) = 0 \quad 0 < x < 1 \end{cases}$

$$u(x,1) = \sum_{n=1}^{\infty} \sin(n\pi x)\left[A_n e^{n\pi} + B_n e^{-n\pi}\right] = 0$$

since $\sin(n\pi x)$ could not be 0

$$A_n e^{n\pi} + B_n e^{-n\pi} = 0$$
$$(1 - B_n)e^{n\pi} + B_n e^{-n\pi} = 0$$
$$e^{n\pi} - B_n e^{n\pi} + B_n e^{-n\pi} = 0$$
$$e^{n\pi} - B_n(e^{n\pi} + e^{-n\pi}) = 0$$
$$e^{n\pi} = B_n(e^{n\pi} + e^{-n\pi})$$
$$\frac{e^{n\pi}}{e^{n\pi} + e^{-n\pi}} = B_n$$
$$B_n = 1 - \frac{1}{e^{2n\pi} + 1}$$

So $A_n = \dfrac{1}{e^{2n\pi} + 1}$

since n only could be 1

$$u(x,y) = \sin(\pi x)\left[\frac{1}{e^{2n\pi}+1}e^{\pi y} + \left(1 - \frac{1}{e^{2n\pi}+1}\right)e^{-\pi y}\right] \quad (10)$$

```
nx=5;%set grid point of x is 5
ny=5;%set grid point of y is 5

%general initial condition
x=linspace(0,1,nx);
y=linspace(0,1,ny);
u=zeros(nx,ny);
u(ny,:)=sin(pi*x);%set u(x,y)=sin(pi*x)
```

```
u = 5×5
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0    0.7071    1.0000    0.7071    0.0000
```

```
[uk]=FD(nx,ny,u)%calculate result
```

```
uk = 5×5
         0    0.7071    1.0000    0.7071    0.0000
         0    0.3318    0.4693    0.3318         0
         0    0.1509    0.2134    0.1509         0
         0    0.0584    0.0825    0.0584         0
         0         0         0         0         0
```

```
mesh(uk)
title("solution from Finite Difference Method with grid 5")
xlabel('x')
ylabel('y')
zlabel('z')

%test for more grid point
nx50=50;
ny50=50;

x50=linspace(0,1,nx50);
y50=linspace(0,1,ny50);
u50=zeros(nx50,ny50);
u50(ny50,:)=sin(pi*x50);

[uk50]=FD(nx50,ny50,u50)

mesh(uk50)
title("solution from Finite Difference Method with grid 50")
xlabel('x')
ylabel('y')
zlabel('z')
```

From just looking at the graph we could see the graph with more grid is more smooth than it with less grid.

Now for testing the accuracy. We need to compute, the difference between the approximate value with the true value with different numbers of grid.

```
%true value for 5 grid point
[xx,yy]=meshgrid(x,y);
%equation (10)
tu=sin(pi*xx).*(1/(exp(2*pi)+1)*exp(pi.*yy)+(1-1/(exp(2*pi)+1))*exp(-pi.*yy))
%plot the result
mesh(tu)
%calculate the average difference between true value and approximate value
Diff_grid_5 = sum(abs(tu-uk),'all')/((nx-1)*(ny-1))

%true value for 50 grid point
[xx50,yy50]=meshgrid(x50,y50);
%equation (10)
tu50=sin(pi*xx50).*(1/(exp(2*pi)+1)*exp(pi.*yy50)+(1-1/(exp(2*pi)+1))*exp(-pi.*yy50));
%plot the result
mesh(tu50)
%calculate the average difference between true value and approximate value
Diff_grid_50 = sum(abs(tu50-uk50),'all')/((nx-1)*(ny-1))
```

So as result the more grid we have the less accuracy we will have.

```
function [uk] = FD(nx, ny, u)
%function of processing finite difference
%set k
    k = [0 -1 0;
        -1 4 -1;
        0 -1 0];
    V = [];
    b = [];
%step 1 loop from (2,2) to (nx-1),(ny-1)
    for j = 2:ny - 1
        for l = 2:nx - 1
            %step 2 set all 0 matrix
            ut = zeros(ny, nx);
            %step 2 change the points around to k
            ut(j - 1:j + 1, l - 1:l + 1) = k;
            %step 3 cut the matrix from (2,2) to (nx-1),(ny-1)
            uc = ut(2:nx - 1, 2:ny - 1);
            %step 4 flat the cut matrix to 1 row and store it to V
            V = [V;reshape(uc', 1, (nx - 2)*(ny - 2))];
            %step 5 calculate b by adding the around point from IC matrix
            %together
            b = [b;u(j + 1, l) + u(j, l - 1) + u(j - 1, l) + u(j, l + 1)];
        end
    end
    uin = V\b;%step 6 calculate the result.
    u(2:nx - 1, 2:ny - 1) = reshape(uin, (nx - 2), (ny - 2))';
    %put the solution back into IC matrix.
    uk = flipud(u);
end
```

5. Use 2D Finite Differences to solve the following BVP

$$u_{xx} + u_{yy} + 2u = 0, \quad 0 < x < 1, 0 < y < 1$$

subject to the boundary condition $u(x, y) = \sin((x + y)\pi)$ on the boundary. Use 6 grid points(4 interior points) in each of the x and y direction. Code up your FD method in to Matlab and plot the solution, test how your solution changes with grid size.

With Finite Differences Method

$$\frac{1}{h^2}(u_{i,j+1} - 2u_{i,j} + u_{i,j-1}) + \frac{1}{h^2}(u_{i+1,j} - 2u_{i,j} + u_{i+1,j}) - 2u_{i,j} = 0$$

$$\frac{1}{h^2}(u_{i,j+1} - 4u_{i,j} + u_{i,j-1} + u_{i+1,j} + u_{i+1,j}) - 2u_{i,j} = 0$$

$$\frac{u_{i,j+1}}{h^2} - \frac{4u_{i,j}}{h^2} + \frac{u_{i,j-1}}{h^2} + \frac{u_{i+1,j}}{h^2} + \frac{u_{i+1,j}}{h^2} - \frac{2h^2u_{i,j}}{h^2} = 0$$

$$\frac{u_{i,j+1}}{h^2} + \frac{u_{i,j-1}}{h^2} + \frac{u_{i+1,j}}{h^2} + \frac{u_{i+1,j}}{h^2} - \left(\frac{4u_{i,j}}{h^2} + \frac{2h^2u_{i,j}}{h^2}\right) = 0$$

$$\frac{u_{i,j+1}}{h^2} + \frac{u_{i,j-1}}{h^2} + \frac{u_{i+1,j}}{h^2} + \frac{u_{i+1,j}}{h^2} - u_{i,j}\left(\frac{4}{h^2} + 2\right) = 0$$

$$u_{i,j+1} + u_{i,j-1} + u_{i+1,j} + u_{i+1,j} = u_{i,j}(4 + 2h^2)$$

$$\frac{1}{4 + 2h^2}(u_{i,j+1} + u_{i,j-1} + u_{i+1,j} + u_{i+1,j}) = u_{i,j}$$

$$u_{i,j} = \frac{1}{4 + 2h^2}(u_{i,j+1} + u_{i,j-1} + u_{i+1,j} + u_{i+1,j})$$

So compare to the last question the only thing changed is the coefficient of $u_{i,j}$

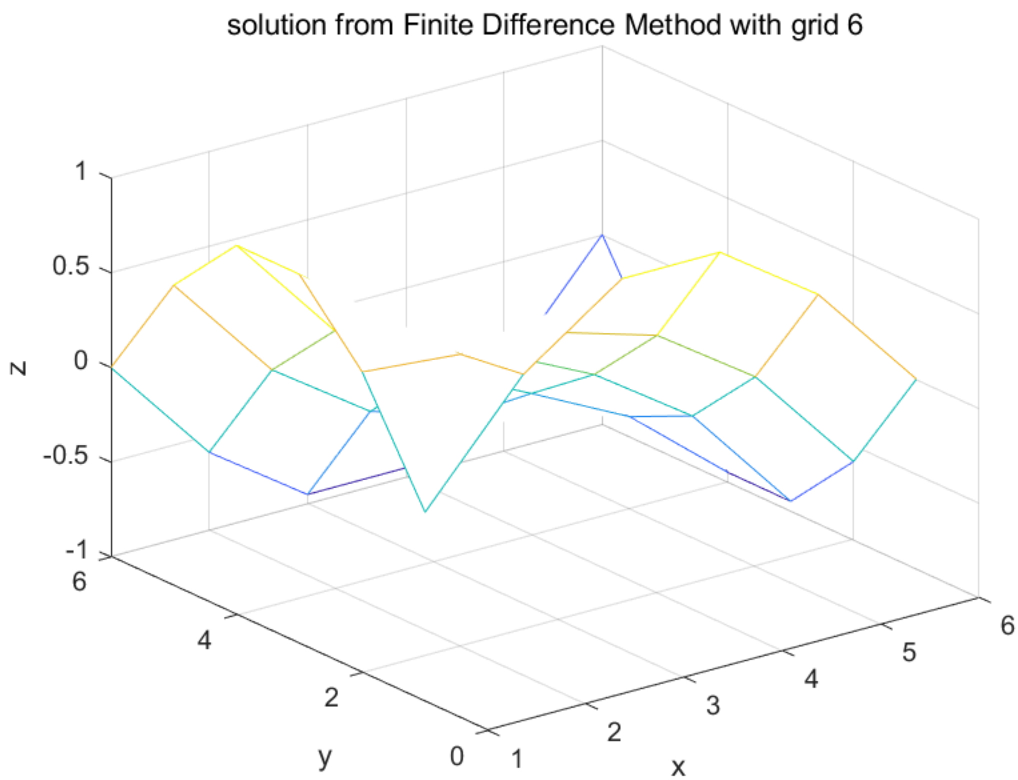
Therefore we will use the same mathod to generate the matrix V and b.

The only change in the method is:

We will set $k = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 + 2h^2 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ (11)

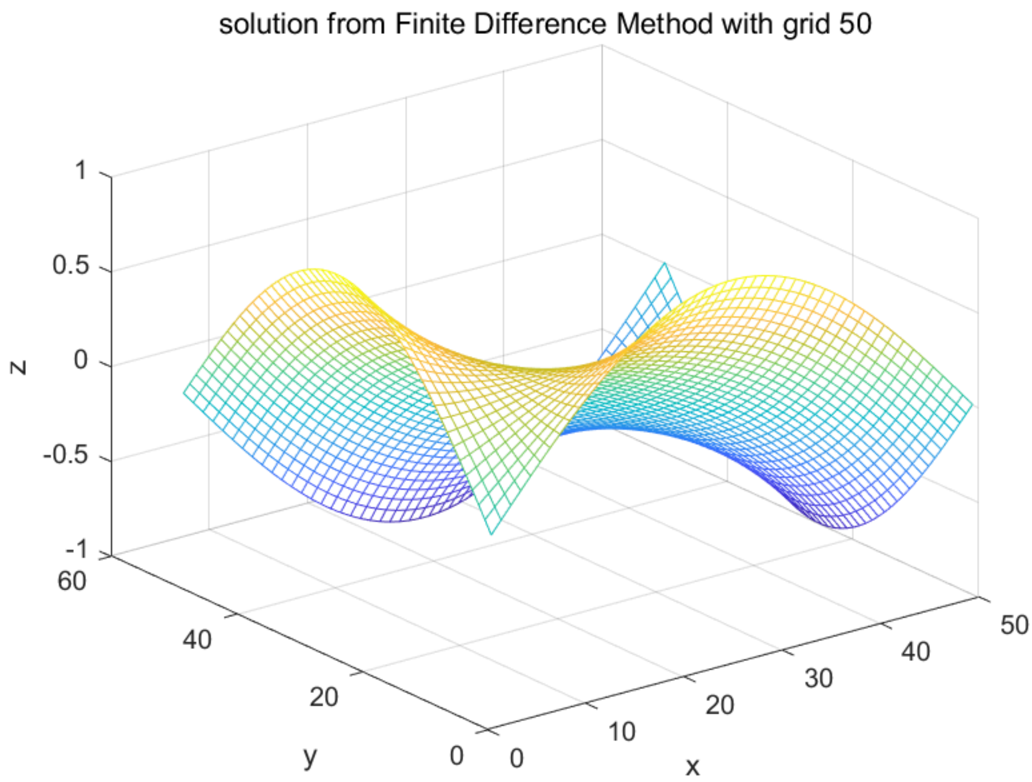
```
clear
%set number of grid=6
nx = 6;
ny = 6;

u = zeros(nx, ny);
x = linspace(0, 1, nx);
y = linspace(0, 1, ny);
%set the initial boundary condition
u(nx, :) = sin(pi*(x + 1));
u(:, ny) = sin(pi*(1 + y));
u(1, :) = sin(pi*(x + 0));
u(:, 1) = sin(pi*(0 + y));
```

```
u = flipud(u);

[uk] = FD(nx, ny, u);
%plot
mesh(uk)
title("solution from Finite Difference Method with grid 6")
xlabel('x')
ylabel('y')
zlabel('z')
```




```
%test for larger grid point
nx50 = 50;
ny50 = 50;

u50 = zeros(nx50, ny50);
x50 = linspace(0, 1, nx50);
y50 = linspace(0, 1, ny50);
%set the initial boundary condition
u50(nx50, :) = sin(pi*(x50 + 1));
u50(:, ny50) = sin(pi*(1 + y50));
u50(1, :) = sin(pi*(x50 + 0));
u50(:, 1) = sin(pi*(0 + y50));
u50 = flipud(u50);

[uk50] = FD(nx50, ny50, u50);

mesh(uk50)
```

```
title("solution from Finite Difference Method with grid 50")
xlabel('x')
ylabel('y')
zlabel('z')
```



As result the solution will get smooth if we use more grid point.

```
function [uk] = FD(nx, ny, u)
%function of Finite Different
    h = 1/(nx - 1);%calculate h
    %adjust k with equation (11)
    k=[0 -1 0;
        -1 4 + 2*h^2 -1;
        0 -1 0];
    V = [];
    b = [];
    for j = 2:ny - 1
        for l = 2:nx - 1
            ut = zeros(ny, nx);
            ut(j - 1:j + 1,l - 1:l + 1) = k;
            uc = ut(2:nx - 1, 2:ny - 1);
            V = [V;reshape(uc', 1, (nx - 2)*(ny - 2))];
            b = [b;u(j + 1, l) + u(j, l - 1) + u(j - 1, l) + u(j, l + 1)];
        end
    end
    uin = V\b;
    u(2:nx - 1, 2:ny - 1) = reshape(uin, (nx - 2), (ny - 2))';
```

```
    uk = flipud(u);
end
```

6. Find the finite-difference solution of the heat-conduction problem

PDE $u_{tt} = u_{xx}$ $\quad 0 < x < 1, \quad t > 0$

BCs $\begin{cases} u(0,t) = 0 & t > 0 \\ u(1,t) = 0 & t > 0 \end{cases}$

IC $u(x,0) = \sin(\pi x) \quad 0 \le x \le 1$

for t=0.005, 0.010, 0.015 by the explicit method. Let $h = \delta x = 0.1$. Plot the solution at x=0, 0.1, 0.2, 0.3, ..., 0.9, 1 for t=0.015.

Solving the problem according to the following step

step 1: Input H=0.1,K=0.005 and end point of t,

step 2: calculate number of grid point on x-direction N and t-direction M.

N=1/H+1;

M=1/K+1;

step 3: compute the ratio $R = \frac{K}{H^2}$

step 4: for i=1:M-1

for j=2:N-1

compute u(i + 1, j) = u(i, j) + R*[u(i, j + 1) - 2*u(i, j) + u(1, j - 1)]

end

u(i + 1, N) = [u(i + 1, N - 1) + HG(i + 1)]/(H + 1)

since in this problem the boundary on the left and right side is 0

u(i + 1, N) = 0

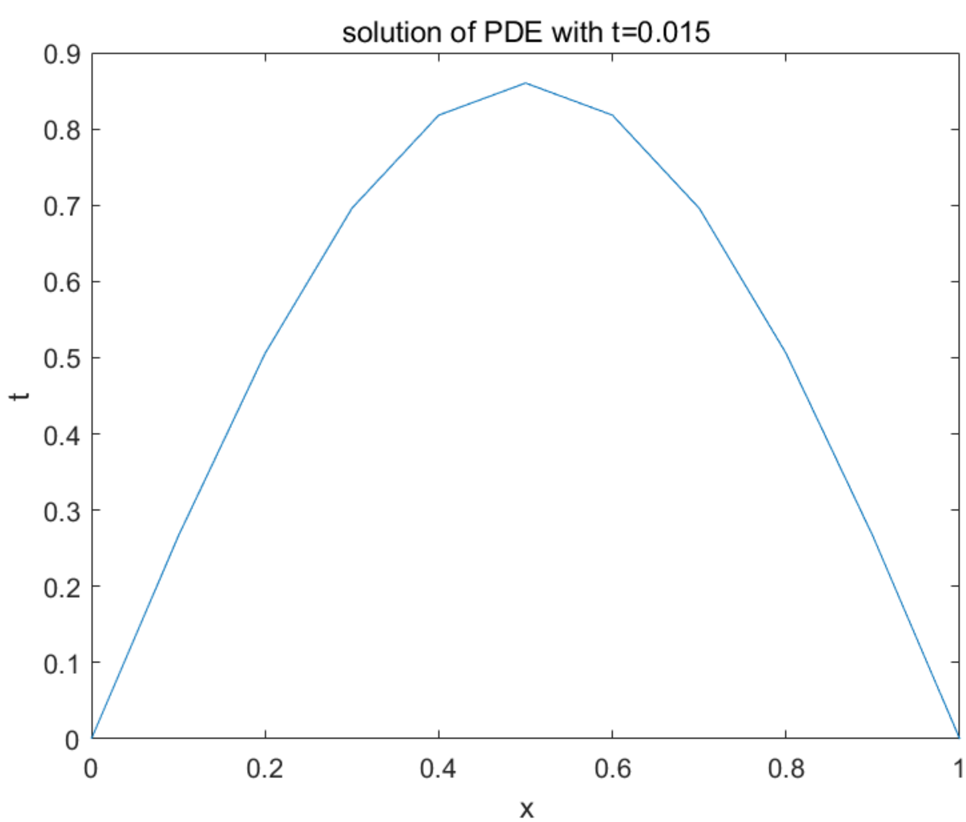
end

```
clear
%step 1 input H, K and ending point p
H = 0.1;
K = 0.005;
p = 0.015;

%step 2, calculate N, M
N = 1/H + 1;
M = p/K + 1;

%step 3, calculate R
R = K/(H*H);

x = linspace(0, 1, N);%generate x = 0, 0.1, 0.2, ... ,0.9, 1
y = linspace(K, p, M);%generate t = 0.005, 0.010, 0.015
```

```
%initial condition
u = zeros(M, N);%initial the matrix
u(1, :) = sin(pi*x);%set initial condition u(x, 0) = sin(pi*x)

%step 4
for i = 1:M - 1
    %loop for i:M-1
    for j = 2:N - 1
        %loop for j=2:N-1
        u(i + 1, j) = u(i, j) + R*(u(i, j + 1) - 2*u(i, j) + u(i, j - 1));
    end
    u(i + 1, N) = 0;
end

u = flipud(u);%since the matrix is upside down with the problem we need to flip it up down
result_t_015 = u(1, :)%the result of PDE with t=0.015;
```

```
result_t_015 = 1×11
         0    0.2658    0.5056    0.6959    0.8181    0.8602    0.8181    0.6959 ···
```

```
plot(x, u(1, :))%only plot the row fot t = 0.015
title('solution of PDE with t=0.015')
xlabel('x')
ylabel('t')
```



7. Solve the following problem analytically (separation of variables) and evaluate the analytical solution

at the grid points: x=0, 0.1, 0.2, ... , 0.9, 1 for t =0.015. Compare these results to your numerical solution in #6 above.

$u_t = u_{xx} \qquad 0 < x < 1 \;\; t > 0$

$u(0, t) = 0 \qquad t > 0$

$u(1, t) = 0 \qquad t > 0$

$u(x, 0) = \sin(\pi x) \qquad 0 \leq x \leq 1$

$$U = X(x)T(t)$$
$$X(x)T'(t) = X''(x)T(t)$$
$$\frac{X(x)T'(t)}{X(x)T(t)} = \frac{X''(x)T(t)}{X(x)T(t)}$$
$$\frac{T'(t)}{T(t)} = \frac{X''(x)}{X(x)}$$
$$\frac{T'(t)}{T(t)} = \frac{X''(x)}{X(x)} = k$$

Then

$$\begin{cases} \dfrac{X''(x)}{X(x)} = k & (12) \\ \dfrac{T'(t)}{T(t)} = k & (13) \end{cases}$$

The solution of (13) is $T(t) = T(0)e^{\mathrm{kt}}$ (14)

With equation (12)

$X''(x) - \mathrm{k}X(x) = 0$ (15)

Guess with the solution $X(x) = e^{\mathrm{rx}}$

Plug the solution into (15):

$$r^2 e^{\mathrm{rx}} - \mathrm{ke}^{\mathrm{rx}} = 0$$
$$e^{\mathrm{rx}}(r^2 - k) = 0$$

so $r^2 - k = 0$

$r^2 = k$

$r = \pm\sqrt{k}$

Then with sign of k, there are three case.

Case 1 $K > 0$

let $K = \lambda^2$

$r = \pm\lambda$

So the solution is:

$X(x) = \mathrm{Ae}^{\lambda x} + \mathrm{Be}^{-\lambda x}$

Case 2 $K < 0$

let $K = -\lambda^2$

$\frac{X(x)}{X''(x)} = -\lambda^2$

$X(x) = -\lambda^2 X''(x)$

$X(x) = \mathrm{Asin}(\lambda x) + \mathrm{Bcos}(\lambda x)$

Case 3 $K = 0$

$r = 0$

$X = 1$

By plug BCs in:

BCs $\begin{cases} u(0,t) = 0 \\ u(1,t) = 0 \quad 0 < t < \infty \end{cases}$

With Case 1 $K > 0, K = \lambda^2$

$u(x,t) = X(x)T(t) = (\mathrm{Ae}^{\lambda x} + \mathrm{Be}^{-\lambda x})T(0)e^{\lambda t}$

By plug $u(0,t) = 0$ and $u(1,t) = 0$ in

$u(0,t) = (\mathrm{Ae}^{\lambda 0} + \mathrm{Be}^{-\lambda 0})T(0)e^{\lambda^2 t} = (A + B)T(0)e^{\lambda^2 t} = 0$

$u(1,t) = (\mathrm{Ae}^{\lambda} + \mathrm{Be}^{-\lambda})T(0)e^{\lambda^2 t} = 0$

So $T(0) = 0$

Therefore $(\mathrm{A}e^{\lambda x}+\mathrm{B}e^{-\lambda x})T(0)e^{\lambda^2 t}$=0 in all cituation.

So Case 1 $K>0$ is not true

Case 2 $K<0, K=-\lambda^2$

$u(x,t)=X(x)T(t)=(\mathrm{A}\sin(\lambda x)+\mathrm{B}\cos(\lambda x))T(0)e^{-\lambda^2 t}$

By plug $u(0,t)=0$ and $u(1,t)=0$ in

$u(0,t)=(\mathrm{A}\sin(\lambda 0)+\mathrm{B}\cos(\lambda 0))T(0)e^{-\lambda^2 t}=\mathrm{B}\mathrm{T}(0)e^{-\lambda^2 t}=0$

$u(1,t)=(\mathrm{A}\sin(\lambda)+\mathrm{B}\cos(\lambda))T(0)e^{-\lambda^2 t}=0$

Since $T(0)$ cannot be 0 from Case 1

$\mathrm{B}=0$.

$u(x,t)=X(x)T(t)=\mathrm{A}\sin(\lambda x)T(0)e^{-\lambda^2 t}$

$u(1,t)=\mathrm{A}\sin(\lambda)T(0)e^{-\lambda^2 t}=0$

$\mathrm{A}\sin(\lambda)=0$

$\lambda=n\pi$

Case 3 $K=0$

$u(x,t)=X(x)T(t)=T(0)e^{\mathrm{kt}}$

$T(0)=0$ if we plug $u(0,t)=0$ in.

Therefore Case 2 $K<0$ is true

And $u(x,t)=X(x)T(t)=\sum_{n=1}^{\infty}(A_n\sin(n\pi x))C_n e^{-(n\pi)^2 t}$

let $\widetilde{A}_n=A_nC_n$

$u(x,t)=\sum_{n=1}^{\infty}\widetilde{A}_n\sin(n\pi x)e^{-(n\pi)^2 t}$

Then plug in IC

IC: $u(x,0) = \sin(\pi x)$

$$u(x,0) = \sum_{n=1}^{\infty} \tilde{A}_n \sin(n\pi x) = \sin(\pi x)$$

So n=1 and $\tilde{A}_n = 1$

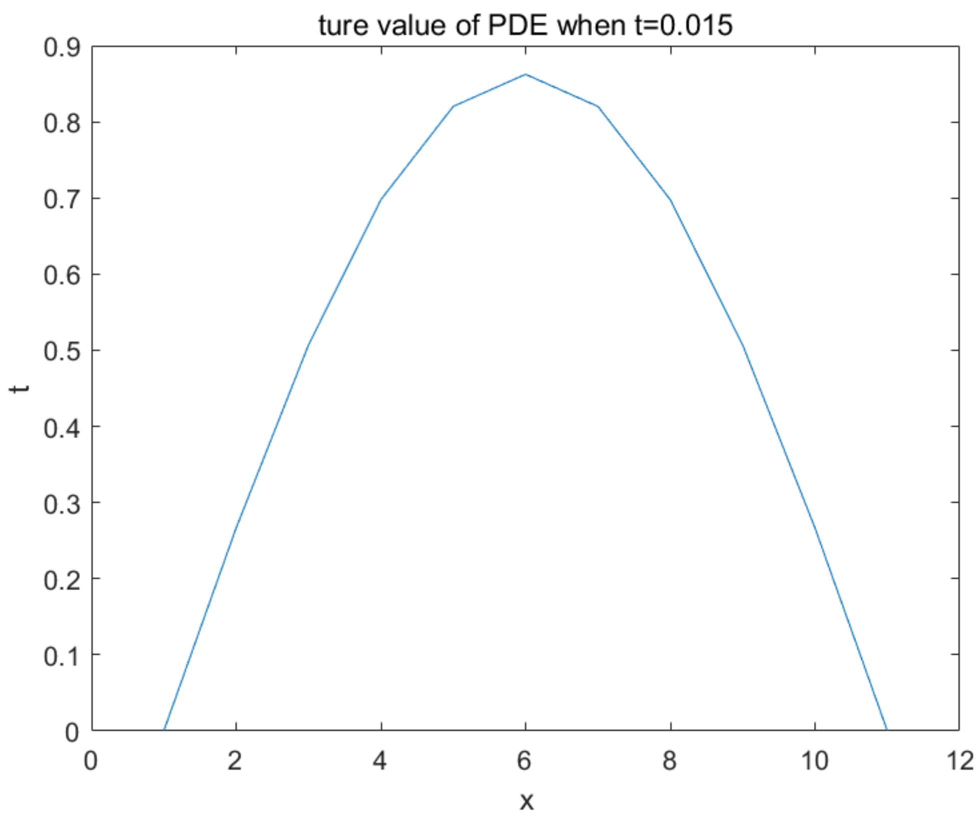
$$u(x,t) = \sin(\pi x) e^{-\pi^2 t} \qquad (16)$$

so for t=0.015

$$u(x,t) = \sin(\pi x) e^{-0.015\pi^2}$$

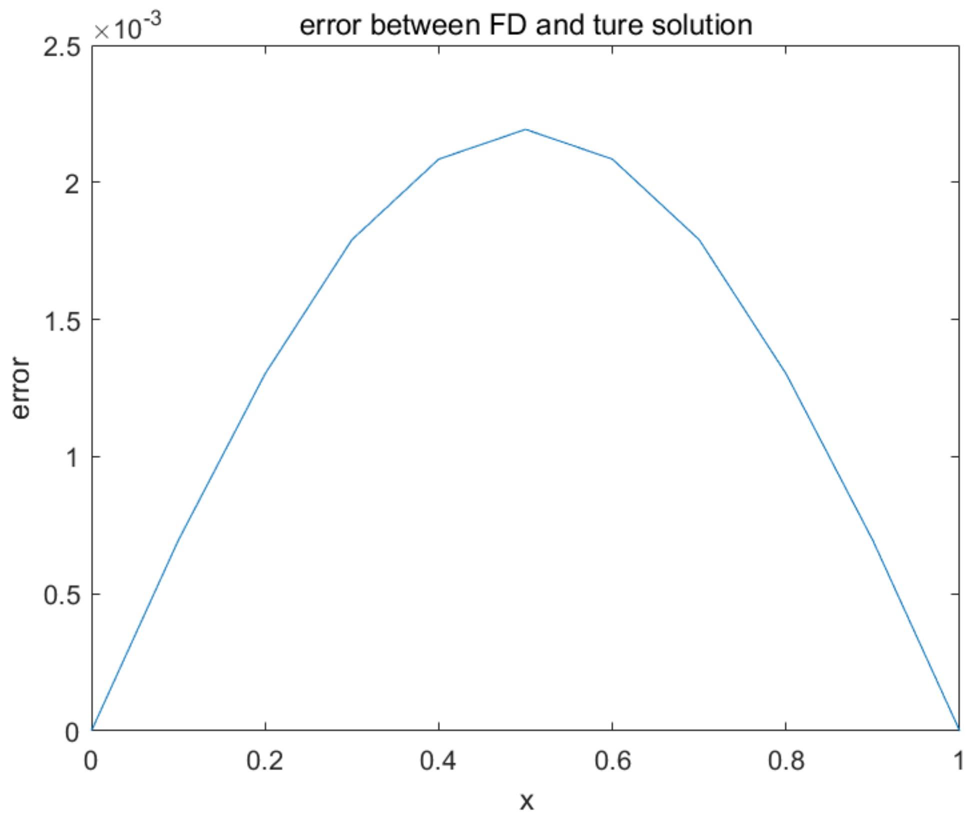
Then plot it.

```
clear
H=0.1;%set H
N=1/H+1;%compute N
x=linspace(0,1,N);%generate same numbers of x with last question

ut=u(x,0.015);%compute the true value
plot(ut)%plot
title('true value of PDE when t=0.015')
xlabel('x')
ylabel('t')
```



From the graph we could see the true value are lower than the approximate value.

```
%get approximate solution from last problem
up = [0 0.2658 0.5056 0.6959 0.8181 0.8602 0.8181 0.6959 0.5056 0.2658 0];
%compute the difference
err = abs(ut - up);
plot(x, err)
title('error between FD and true solution')
xlabel('x')
ylabel('error')
```



From the plot we could clearly saw that the error with explicit method is small.

```
function [ut]=u(x,t)
%funtion of equation (16)
   ut=sin(pi*x)*exp(-pi^2*t);
end
```

8. Consider the problem

PDE $u_{tt} = u_{xx}$ $0 < x < 1,\quad t > 0$

BCs $\begin{cases} u(0,t) = 0 & t > 0 \\ u(1,t) = 0 & t > 0 \end{cases}$

IC $u(x,0) = \sin(\pi x) \quad 0 \le x \le 1$

Solve this problem using the method described in class (Implicit FD method) using various values of lambda including $\lambda = 0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1$ and experiment with step sizes in x and t to check accuracy. Remember that if you use $\lambda = 0$ there are specific guidelines about how small the ratio of $\frac{k}{h^2}$ must be. Compare your results to the previous two problems (#6 and #7).

Plot your solutions for various times (up to at least t=0.05) and compare to the true solution. You may find it instructive to also plot the error between the true and approximate solutions.

Print out your coeffcient matrix as well as the RHS vector for a small number of grid points to make sure it looks like you think it does.

The PDE can be written as:

$$\frac{1}{k}(u_{i+1,j} - u_{ij}) = \frac{\lambda}{h^2}[u_{i+1,j+1} - 2u_{i+1,j} + u_{i+1,j-1}] + \frac{(1-\lambda)}{h^2}[u_{i,j+1} - 2u_{ij} + u_{i,j-1}]$$
$$\frac{1}{k}u_{i+1,j} - \frac{1}{k}u_{ij} = \frac{\lambda}{h^2}u_{i+1,j+1} - \frac{\lambda}{h^2}2u_{i+1,j} + \frac{\lambda}{h^2}u_{i+1,j-1} + \frac{(1-\lambda)}{h^2}u_{i,j+1} - \frac{(1-\lambda)}{h^2}2u_{ij} + \frac{(1-\lambda)}{h^2}u_{i,j-1}$$
$$u_{i+1,j} - u_{ij} = \frac{\lambda k}{h^2}u_{i+1,j+1} - \frac{\lambda k}{h^2}2u_{i+1,j} + \frac{\lambda k}{h^2}u_{i+1,j-1} + \frac{(1-\lambda)k}{h^2}u_{i,j+1} - \frac{(1-\lambda)k}{h^2}2u_{ij} + \frac{(1-\lambda)k}{h^2}u_{i,j-1}$$

let $\frac{k}{h^2} = r$

$$u_{i+1,j} - u_{ij} = \lambda r u_{i+1,j+1} - \lambda r 2u_{i+1,j} + \lambda r u_{i+1,j-1} + (1-\lambda)ru_{i,j+1} - 2(1-\lambda)ru_{ij} + (1-\lambda)ru_{i,j-1}$$
$$-\lambda \mathrm{r}\mathrm{u}_{i+1,j+1} + (1+2\lambda r)u_{i+1,j} - \lambda \mathrm{r}\mathrm{u}_{i+1,j-1} = (1-\lambda)\mathrm{r}\mathrm{u}_{i,j+1} + (1-2(1-\lambda)r)u_{ij} + (1-\lambda)\mathrm{r}\mathrm{u}_{i,j-1}$$

For $j = 2,3,4,\ldots,n-1$ and fixed i

If we fix i=1

j=2 $\quad -\lambda \mathrm{r}\mathrm{u}_{2,3} + (1+2\lambda r)u_{2,2} - \lambda \mathrm{r}\mathrm{u}_{2,1} = (1-\lambda)\mathrm{r}\mathrm{u}_{1,3} + (1-2(1-\lambda)r)u_{12} + (1-\lambda)\mathrm{r}\mathrm{u}_{1,1} = P_2$

which u_{21}, u_{12}, u_{11} and $u_{13}$are the known boundary condition

j=3 $\quad -\lambda \mathrm{r}\mathrm{u}_{24} + (1+2\lambda r)u_{23} - \lambda \mathrm{r}\mathrm{u}_{22} = (1-\lambda)\mathrm{r}\mathrm{u}_{12} + (1-2(1-\lambda)r)u_{13} + (1-\lambda)\mathrm{r}\mathrm{u}_{12} = P_3$

And so on.

So in general we could write the all equation to (n-2) by (n-2) matrix K

And the unknown vecter u, which is:

Ku=P

$$\begin{bmatrix} (1+2r\lambda) & -\lambda r & 0 & \dots & 0 \\ -\lambda r & (1+2r\lambda) & -\lambda r & 0 & \vdots \\ 0 & -\lambda r & (1+2r\lambda) & \ddots & 0 \\ \vdots & 0 & \ddots & \ddots & -\lambda r \\ 0 & 0 & 0 & -\lambda r & (1+2r\lambda) \end{bmatrix} \begin{bmatrix} u_{22} \\ u_{23} \\ u_{24} \\ \vdots \\ u_{2,n-1} \end{bmatrix} = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ \vdots \\ P_{n-1} \end{bmatrix}$$

where

$$P_j = r(1-\lambda)u_{i,j} + [1-2r(1-\lambda)]u_{i,j} + r(1-\lambda)u_{i,j-1} \qquad (17)$$

for fixed i=1.

if we finish computing the situation i=1 we can then compute the situation i=2,3,...

```
clear
%set H and K
H=0.1;%delta x=0.1
K=0.001;%delta t=0.001
pp=0.05;%max point of t
[N,M,R,u_c]=inti(H,K,pp);%set initial condition
u_c
```

```
u_c = 50×10
         0    0.3420    0.6428    0.8660    0.9848    0.9848    0.8660    0.6428 ···
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0
      ⋮
```

```
lam=0;%set lamda=0
[ck0,u_c0]=imp(N,M,lam,u_c,R);
%show the coefficient martix with lamda=0
coe_matrix_lam_0=ck0
```

```
coe_matrix_lam_0 = 9×9
     1     0     0     0     0     0     0     0     0
     0     1     0     0     0     0     0     0     0
     0     0     1     0     0     0     0     0     0
     0     0     0     1     0     0     0     0     0
     0     0     0     0     1     0     0     0     0
     0     0     0     0     0     1     0     0     0
     0     0     0     0     0     0     1     0     0
     0     0     0     0     0     0     0     1     0
```

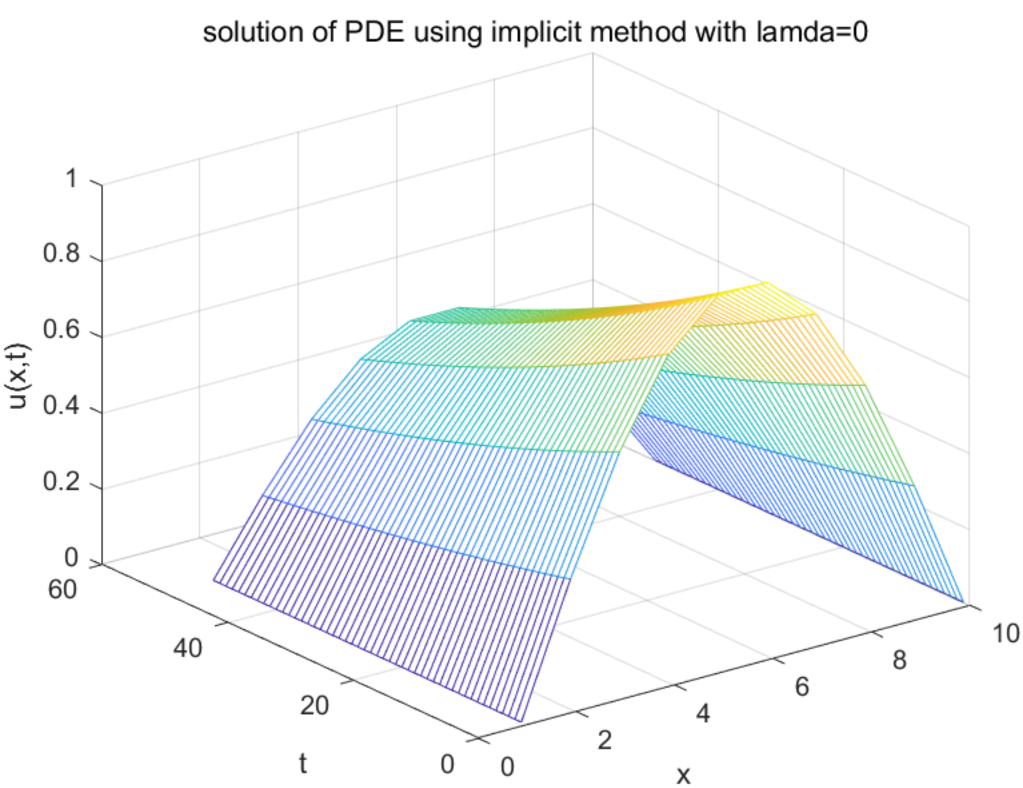
```
     0     0     0     0     0     0     0     0     1
```

```
%show the solution with lamda=0
solution_lam_0=u_c0
```

solution_lam_0 = 50×10

0	0.3420	0.6428	0.8660	0.9848	0.9848	0.8660	0.6428	⋯
0	0.3379	0.6350	0.8556	0.9729	0.9729	0.8556	0.6350	
0	0.3338	0.6274	0.8453	0.9612	0.9612	0.8453	0.6274	
0	0.3298	0.6198	0.8351	0.9496	0.9496	0.8351	0.6198	
0	0.3258	0.6123	0.8250	0.9381	0.9381	0.8250	0.6123	
0	0.3219	0.6049	0.8150	0.9268	0.9268	0.8150	0.6049	
0	0.3180	0.5977	0.8052	0.9157	0.9157	0.8052	0.5977	
0	0.3142	0.5904	0.7955	0.9046	0.9046	0.7955	0.5904	
0	0.3104	0.5833	0.7859	0.8937	0.8937	0.7859	0.5833	
0	0.3066	0.5763	0.7764	0.8829	0.8829	0.7764	0.5763	
⋮								

```
%plot
mesh(u_c0)
title('solution of PDE using implicit method with lamda=0')
xlabel('x')
ylabel('t')
zlabel('u(x,t)')
```



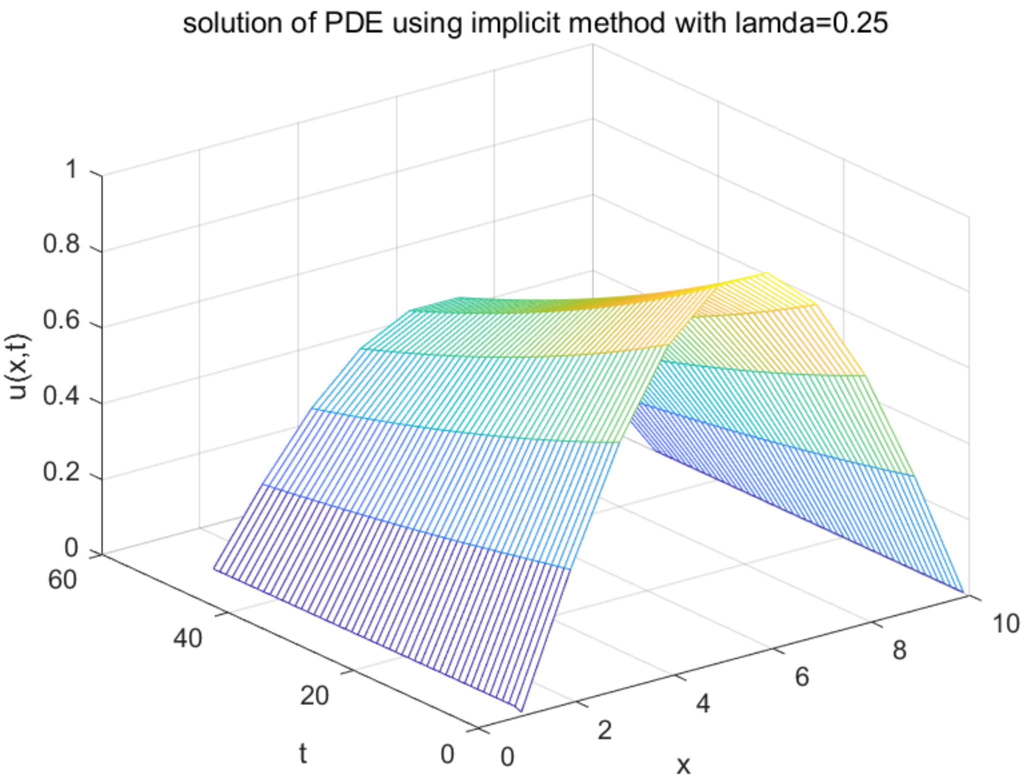

```
lam=1/4;
[ck025,u_c025]=imp(N,M,lam,u_c,R);
%show the coefficient martix with lamda=1/4
coe_matrix_lam_025=ck025
```

```
coe_matrix_lam_025 = 9×9
    1.0500   -0.0250         0         0         0         0         0         0 ···
   -0.0250    1.0500   -0.0250         0         0         0         0         0
         0   -0.0250    1.0500   -0.0250         0         0         0         0
         0         0   -0.0250    1.0500   -0.0250         0         0         0
         0         0         0   -0.0250    1.0500   -0.0250         0         0
         0         0         0         0   -0.0250    1.0500   -0.0250         0
         0         0         0         0         0   -0.0250    1.0500   -0.0250
         0         0         0         0         0         0   -0.0250    1.0500
         0         0         0         0         0         0         0   -0.0250
```

```
%show the solution with lamda=1/4
solution_lam_025=u_c025
```

```
solution_lam_025 = 50×10
         0    0.3420    0.6428    0.8660    0.9848    0.9848    0.8660    0.6428 ···
    0.0080    0.3381    0.6351    0.8556    0.9730    0.9730    0.8556    0.6351
    0.0080    0.3348    0.6275    0.8453    0.9613    0.9613    0.8453    0.6274
    0.0079    0.3313    0.6200    0.8352    0.9497    0.9497    0.8352    0.6199
    0.0078    0.3278    0.6127    0.8251    0.9383    0.9383    0.8251    0.6124
    0.0077    0.3243    0.6055    0.8152    0.9270    0.9270    0.8152    0.6051
    0.0076    0.3208    0.5984    0.8055    0.9159    0.9159    0.8054    0.5978
    0.0076    0.3172    0.5913    0.7958    0.9049    0.9048    0.7957    0.5906
    0.0075    0.3137    0.5844    0.7863    0.8940    0.8940    0.7861    0.5835
    0.0074    0.3101    0.5775    0.7769    0.8833    0.8832    0.7767    0.5765
      ⋮
```

```
%plot
mesh(u_c025)
title('solution of PDE using implicit method with lamda=0.25')
xlabel('x')
ylabel('t')
zlabel('u(x,t)')
```




```
lam=0.5;
[ck05,u_c05]=imp(N,M,lam,u_c,R);
%show the coefficient martix with lamda=1/2
coe_matrix_lam_05=ck05
```

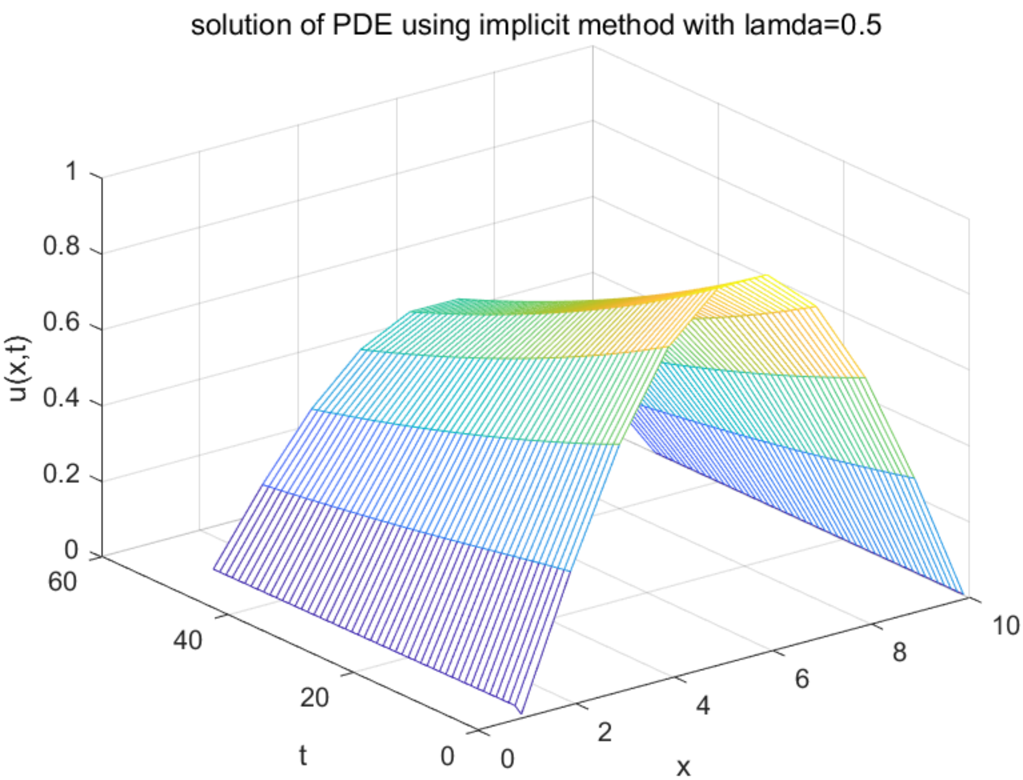
```
coe_matrix_lam_05 = 9×9
    1.1000   -0.0500         0         0         0         0         0         0 ···
   -0.0500    1.1000   -0.0500         0         0         0         0         0
         0   -0.0500    1.1000   -0.0500         0         0         0         0
         0         0   -0.0500    1.1000   -0.0500         0         0         0
         0         0         0   -0.0500    1.1000   -0.0500         0         0
         0         0         0         0   -0.0500    1.1000   -0.0500         0
         0         0         0         0         0   -0.0500    1.1000   -0.0500
         0         0         0         0         0         0   -0.0500    1.1000
         0         0         0         0         0         0         0   -0.0500
```

```
%show the solution with lamda=1/2
solution_lam_05=u_c05
```

```
solution_lam_05 = 50×10
         0    0.3420    0.6428    0.8660    0.9848    0.9848    0.8660    0.6428 ···
    0.0154    0.3386    0.6351    0.8556    0.9730    0.9730    0.8556    0.6351
    0.0153    0.3358    0.6276    0.8454    0.9613    0.9613    0.8454    0.6275
    0.0151    0.3329    0.6203    0.8353    0.9498    0.9498    0.8352    0.6199
    0.0150    0.3298    0.6131    0.8253    0.9384    0.9384    0.8252    0.6125
    0.0148    0.3266    0.6061    0.8155    0.9272    0.9272    0.8153    0.6052
    0.0147    0.3233    0.5991    0.8058    0.9161    0.9161    0.8056    0.5979
    0.0145    0.3200    0.5922    0.7962    0.9051    0.9051    0.7959    0.5907
    0.0144    0.3167    0.5854    0.7868    0.8943    0.8942    0.7864    0.5837
    0.0142    0.3133    0.5787    0.7774    0.8836    0.8835    0.7769    0.5767
```

⋮

```
%plot
mesh(u_c05)
title('solution of PDE using implicit method with lamda=0.5')
xlabel('x')
ylabel('t')
zlabel('u(x,t)')
```




```
lam=3/4;
[ck075,u_c075]=imp(N,M,lam,u_c,R);
%show the coefficient martix with lamda=3/4
coe_matrix_lam_075=ck075
```

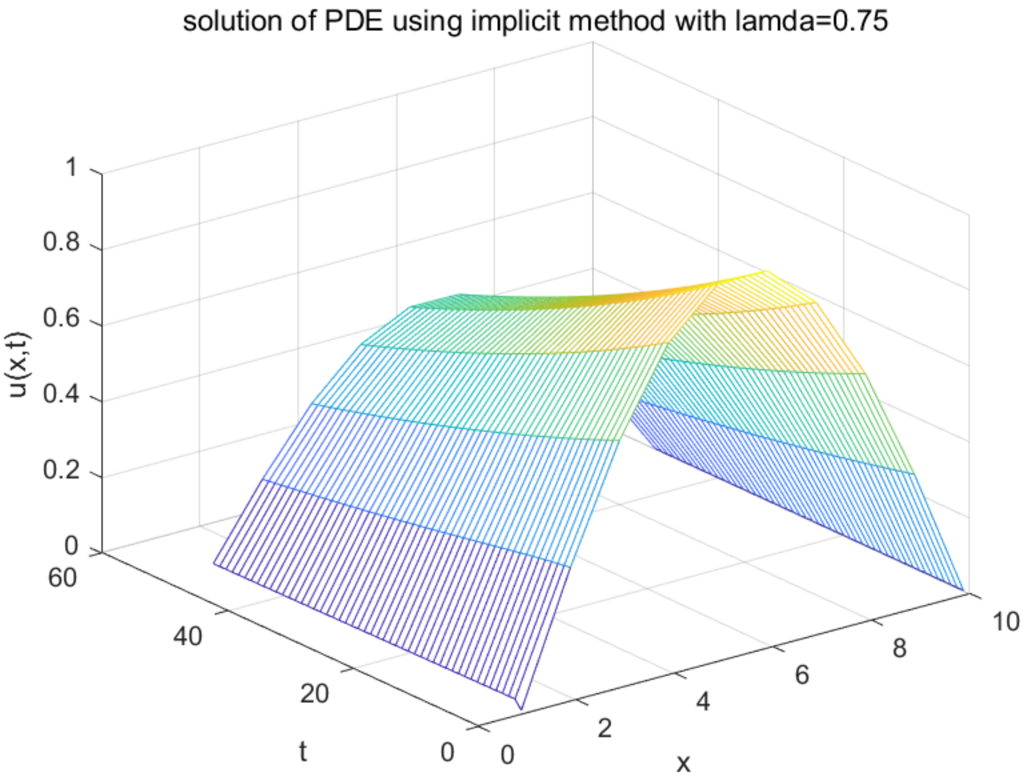
```
coe_matrix_lam_075 = 9×9
    1.1500   -0.0750         0         0         0         0         0         0 ···
   -0.0750    1.1500   -0.0750         0         0         0         0         0
         0   -0.0750    1.1500   -0.0750         0         0         0         0
         0         0   -0.0750    1.1500   -0.0750         0         0         0
         0         0         0   -0.0750    1.1500   -0.0750         0         0
         0         0         0         0   -0.0750    1.1500   -0.0750         0
         0         0         0         0         0   -0.0750    1.1500   -0.0750
         0         0         0         0         0         0   -0.0750    1.1500
         0         0         0         0         0         0         0   -0.0750
```

```
%show the solution with lamda=3/4
solution_lam_075=u_c075
```

```
solution_lam_075 = 50×10
```

0	0.3420	0.6428	0.8660	0.9848	0.9848	0.8660	0.6428	···
0.0221	0.3394	0.6352	0.8557	0.9730	0.9730	0.8557	0.6351	
0.0220	0.3370	0.6278	0.8455	0.9614	0.9614	0.8454	0.6275	
0.0218	0.3344	0.6206	0.8354	0.9499	0.9499	0.8353	0.6200	
0.0216	0.3317	0.6136	0.8255	0.9386	0.9386	0.8254	0.6126	
0.0214	0.3288	0.6067	0.8157	0.9274	0.9273	0.8155	0.6053	
0.0212	0.3257	0.5999	0.8061	0.9163	0.9163	0.8057	0.5980	
0.0210	0.3227	0.5931	0.7966	0.9054	0.9053	0.7961	0.5909	
0.0208	0.3195	0.5865	0.7872	0.8946	0.8945	0.7866	0.5838	
0.0206	0.3163	0.5799	0.7780	0.8840	0.8838	0.7772	0.5769	
⋮								

```
%plot
mesh(u_c075)
title('solution of PDE using implicit method with lamda=0.75')
xlabel('x')
ylabel('t')
zlabel('u(x,t)')
```




```
lam=1;
[ck1,u_c1]=imp(N,M,lam,u_c,R);
%show the coefficient martix with lamda=1
coe_matrix_lam_1=ck1
```

coe_matrix_lam_1 = 9×9

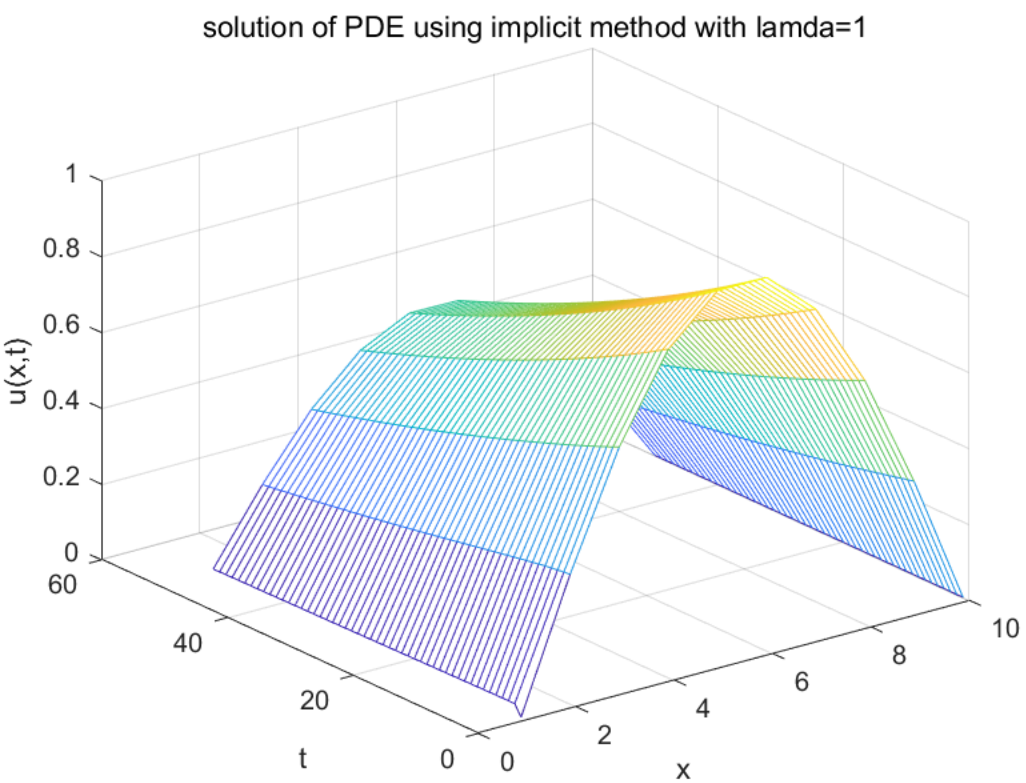
1.2000	-0.1000	0	0	0	0	0	0	···
-0.1000	1.2000	-0.1000	0	0	0	0	0	
0	-0.1000	1.2000	-0.1000	0	0	0	0	
0	0	-0.1000	1.2000	-0.1000	0	0	0	
0	0	0	-0.1000	1.2000	-0.1000	0	0	

```
         0         0         0         0   -0.1000    1.2000   -0.1000         0
         0         0         0         0         0   -0.1000    1.2000   -0.1000
         0         0         0         0         0         0   -0.1000    1.2000
         0         0         0         0         0         0         0   -0.1000
```

```
%show the solution with lamda=1
solution_lam_1=u_c1
```

```
solution_lam_1 = 50×10
         0    0.3420    0.6428    0.8660    0.9848    0.9848    0.8660    0.6428 ···
    0.0284    0.3403    0.6353    0.8557    0.9731    0.9731    0.8557    0.6351
    0.0282    0.3383    0.6281    0.8456    0.9615    0.9615    0.8455    0.6276
    0.0280    0.3360    0.6210    0.8356    0.9500    0.9500    0.8354    0.6201
    0.0278    0.3335    0.6141    0.8257    0.9387    0.9387    0.8255    0.6127
    0.0276    0.3308    0.6073    0.8160    0.9276    0.9275    0.8156    0.6054
    0.0273    0.3280    0.6007    0.8064    0.9165    0.9165    0.8059    0.5982
    0.0271    0.3251    0.5941    0.7970    0.9057    0.9056    0.7963    0.5910
    0.0268    0.3221    0.5875    0.7877    0.8949    0.8948    0.7868    0.5840
    0.0266    0.3191    0.5811    0.7785    0.8843    0.8841    0.7775    0.5770
      ⋮
```

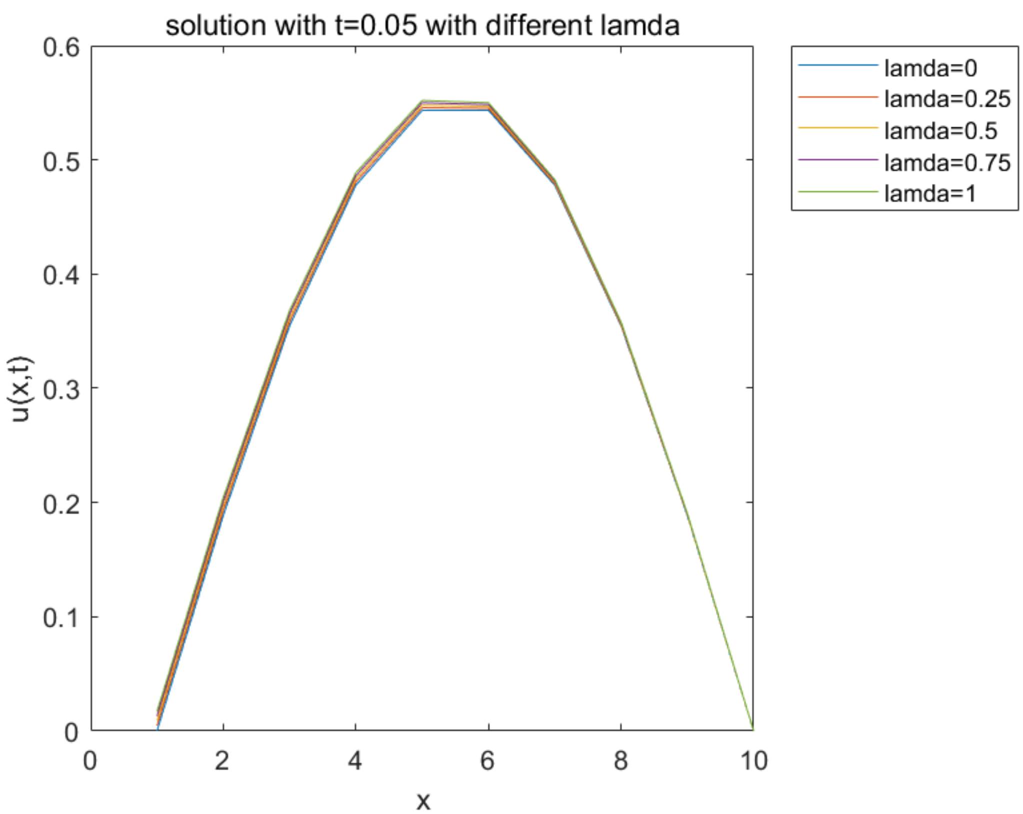
```
%plot
mesh(u_c1)
title('solution of PDE using implicit method with lamda=1')
xlabel('x')
ylabel('t')
zlabel('u(x,t)')
```



And if we plot the solution of t=0.05 together

```
plot(solution_lam_0(end,:))
```

```
hold on
plot(solution_lam_025(end,:))
plot(solution_lam_05(end,:))
plot(solution_lam_075(end,:))
plot(solution_lam_1(end,:))
hold off
legend('lamda=0','lamda=0.25','lamda=0.5','lamda=0.75','lamda=1')
title('solution with t=0.05 with different lamda')
xlabel('x')
ylabel('u(x,t)')
```

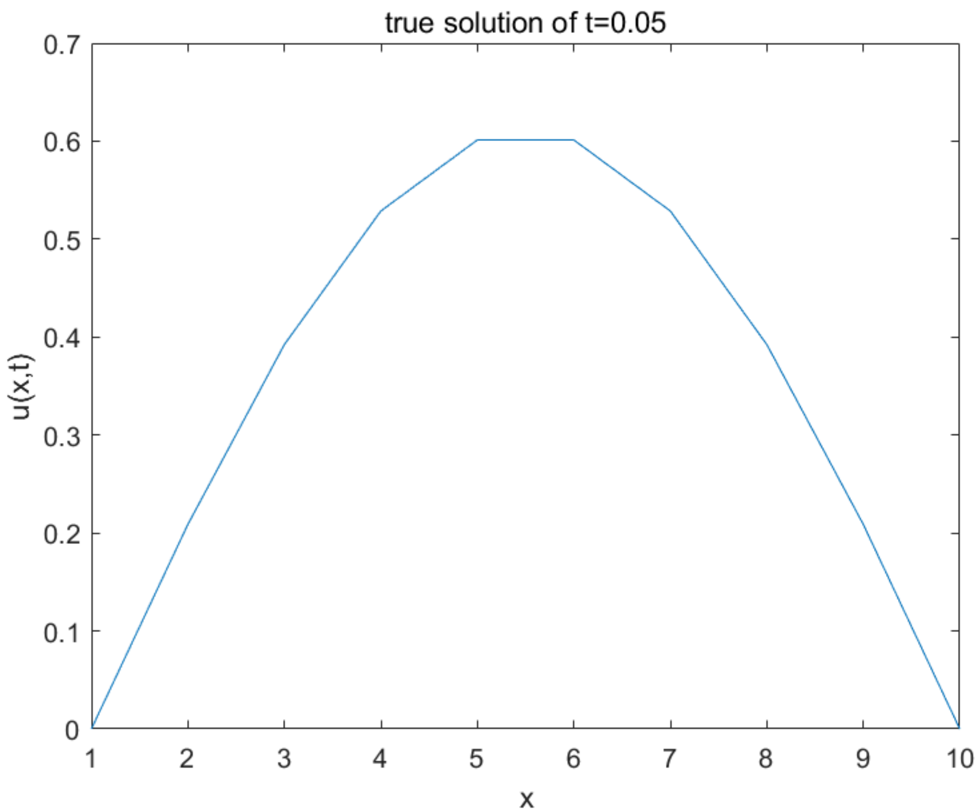


And recall the true solution

$$u(x,t) = \sin(\pi x)e^{-\pi^2 t} \qquad (18)$$

and plot it with t=0.05

```
x=linspace(0,1,N);
[ut]=u_true(x,0.05);%compute the true value
plot(ut)
title('true solution of t=0.05')
xlabel('x')
ylabel('u(x,t)')
```

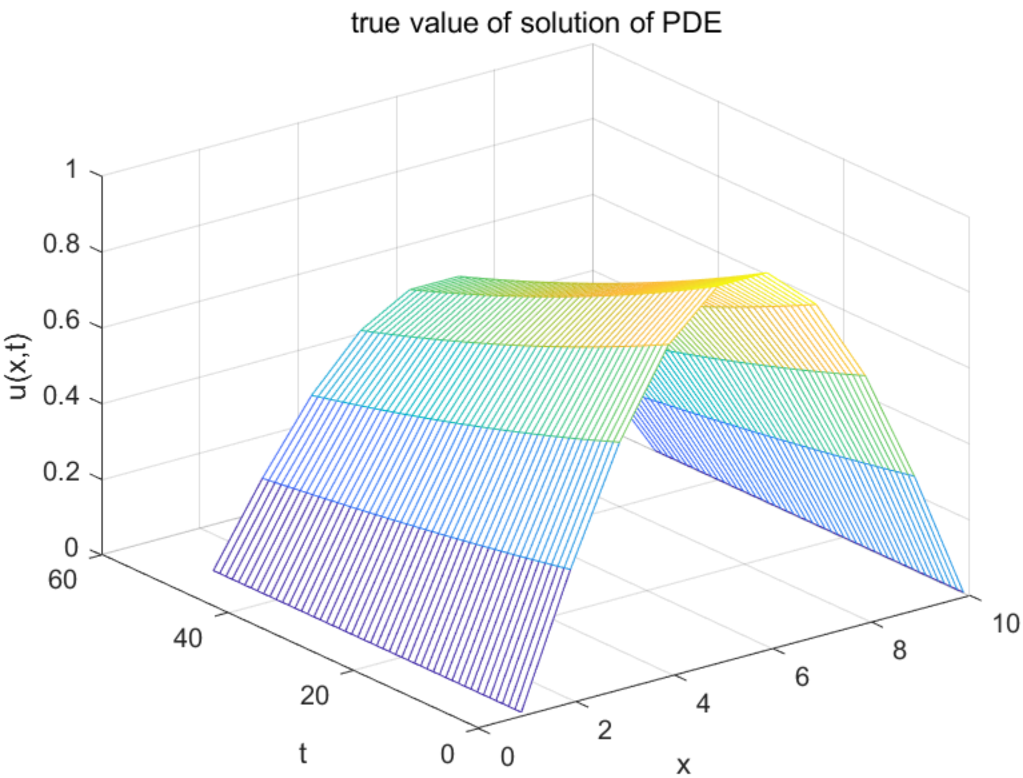


ANd if we plot from t=0 to t=0.05

```
t=linspace(0,pp,M);
[xx,tt]=meshgrid(x,t);
[ut2]=u_true(xx,tt)%compute the true value
```

```
ut2 = 50×10
         0    0.3420    0.6428    0.8660    0.9848    0.9848    0.8660    0.6428 ···
         0    0.3386    0.6363    0.8573    0.9749    0.9749    0.8573    0.6363
         0    0.3352    0.6300    0.8488    0.9652    0.9652    0.8488    0.6300
         0    0.3318    0.6237    0.8403    0.9555    0.9555    0.8403    0.6237
         0    0.3285    0.6174    0.8318    0.9459    0.9459    0.8318    0.6174
         0    0.3252    0.6112    0.8235    0.9364    0.9364    0.8235    0.6112
         0    0.3220    0.6051    0.8152    0.9271    0.9271    0.8152    0.6051
         0    0.3187    0.5990    0.8071    0.9178    0.9178    0.8071    0.5990
         0    0.3155    0.5930    0.7990    0.9086    0.9086    0.7990    0.5930
         0    0.3124    0.5871    0.7910    0.8995    0.8995    0.7910    0.5871
      ⋮
```

```
mesh(ut2)
title('true value of solution of PDE')
xlabel('x')
ylabel('t')
zlabel('u(x,t)')
```




```
function [ut]=u_true(x,t)
%funtion of equation (17)
   ut=sin(pi*x).*exp(-pi^2.*t);
end

function [p]=p(lam,r,u,n)
%function of compute P_j
%input lamda, r: the ratio K/H^2, u: initial condition, n
%output vector P
    for i=2:n-1
        %compute P_2 to P_n-1
        p(i)=r*(1-lam)*u(1,i+1)+(1-2*r*(1-lam))*u(1,i)+r*(1-lam)*u(1,i-1);
    end
end

function [N,M,R,u_c]=inti(H,K,pp)
%function of general initial condition
%with input H K and p out put :
%N number of grid point x direction has
%M number of grid point t direction has
%R the ratio of K/H^2
%u_c the matrix contain the initial condition
    N=1/H+1;
    M=pp/K+1;
    R=K/(H*H);
    M=M-1;
    N=N-1;
```

```
    x=linspace(0,1,N);
    u=zeros(M,N);
    u(1,:)=sin(pi*x);%set boundary condition
    u_c=u;
end

function [ck]=coeMatrix(lam,N,R)
%compute Coefiicient Matrix K
    v=ones(N-1,1)*(1+2*R*lam);%general 1+2r*lamda
    ck=diag(v);%diagonalize the 1+2r*lamda
    cv=ones(N-2,1)*(-lam*R);%general -lamda*r
    pc=diag(cv,1);%diagonalize the -lamda*r on and line up diagonal
    pc=pc+pc';%also set the line down diagonal -lamda*r
    ck=ck+pc;%combine -lamda*r and 1+2r*lamda together.
end

function [ck,u_c]=imp(N,M,lam,u_c,R)
%function of compute the solution Coefiicient Matrix
    ck=coeMatrix(lam,N,R);%load Coefiicient Matrix
    for l=1:M-1
        %loop for i
        P1=p(lam,R,u_c(l,:),N);%compute P_j with different i
        u_c(l+1,1:N-1)=ck\P1';
        %compute the solution and store the vector into a matrix line by line
    end
end
```