CHE636A Homework-3
Due date: Jan 9th, 2022 at midnight on Mookit

Question 1: Solve the following equation using **implicit finite difference** method and answer the questions given in part (i) & (ii). The initial condition, boundary condition and exact solution of this equation are given below.

$$\frac{\partial^2 T}{\partial x^2} = \frac{\partial T}{\partial t}$$

where 0<x<1, t>0

T=0, x=0, t>0
T=0, x=1, t>0
T=100sin(2πx), t=0, 0≤x≤1

The exact analytical solution of this problem is

$$T(x,t) = 100e^{-4\pi^2 t}\sin(2\pi x)$$

Solve this problem using **implicit** finite difference method as discussed in the class. Plot temperature profiles obtained from exact solute (use dots/circles 'o' to show exact solution) and approximate solution as a function of x **after 0.04 seconds**.

Part 1(a): Using N=20, solve for the following cases (Hint: choose dt accordingly) and show the plot of temperature profile as a function of x (after 0.04 seconds as mentioned above).

(i) $\frac{\Delta t}{\Delta x^2} = 0.25$

(ii) $\frac{\Delta t}{\Delta x^2} = 0.75$

Part 1(b): Further, keeping $\frac{\Delta t}{\Delta x^2} = 0.25$, study the impact of changing N on the RMSE. Take N=10, 20, 30 & 40 and obtain the RMSE (Hint: Note, dt should be changed for each case such that the condition $\frac{\Delta t}{\Delta x^2} = 0.25$ is satisfied). RMSE is defined as shown below:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(T_i^{exact} - T_i^{approximate})^2}{N}}$$

where T_i^{exact} and $T_i^{approximate}$ are the exact and approximate solution, respectively, at some x and N is the number of nodes.

Question 2: Solve the previous problem with the following initial and boundary conditions (using implicit finite difference method). The exact solution using these conditions is also given below. Solve this problem for time **t=0.5 seconds**

$\frac{\partial T}{\partial x} = 0$, x=0, t>0
T=0, x=1, t>0
$T = 100\cos\left(\frac{\pi}{2}x\right)$, t=0, 0≤x≤1

The exact analytical solution of this problem is

$$T(x,t) = 100\cos\left(\frac{\pi}{2}x\right)\exp\left(-\frac{\pi^2}{4}t\right)$$

Question 1 partA

```
%save the first two box in matlab 1st as matrix & 2nd as matrix_solver filename
then run the last code file
```

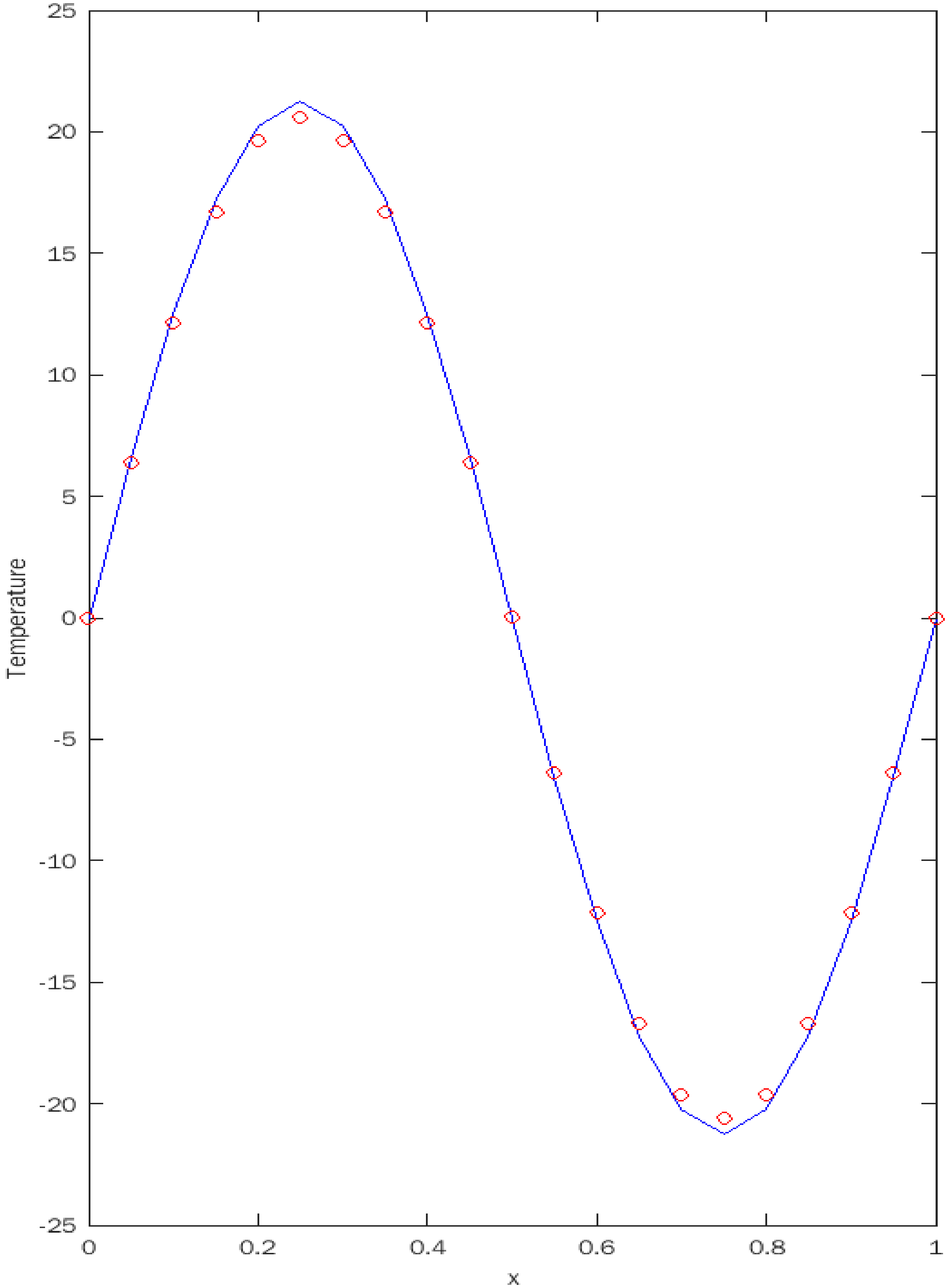
```
function [p,q,r,s] = matrix(N,a,T_old)

p = zeros(N+1,1);
q = zeros(N+1,1);
r = zeros(N+1,1);
s = zeros(N+1,1);

%intial elements using boundary conditions
p(1)=0; q(1)=1;r(1)=0;s(1)=0;
p(N+1)=0;q(N+1)=1;r(N+1)=0;s(N+1)=0;

%make diagonal elements of matrix
for i=2:N
    p(i) = a;
    q(i) = -(1+2*a);
    r(i) = a;
    s(i) = -T_old(i);
end
end
```

```
%to solve matrix
function A = matrix_solver(p,q,r,s,N)
A=zeros(N+1,1);
for i=1:N
    U=p(i+1)/q(i);
    p(i+1)=0;
    q(i+1)=q(i+1)-U*r(i);
    s(i+1)=s(i+1)-U*s(i);
end

A(N+1,1)=s(N+1)/q(N+1);
for i=N:-1:1
    A(i)=(s(i)-r(i)*A(i+1))/q(i);
end
end
```

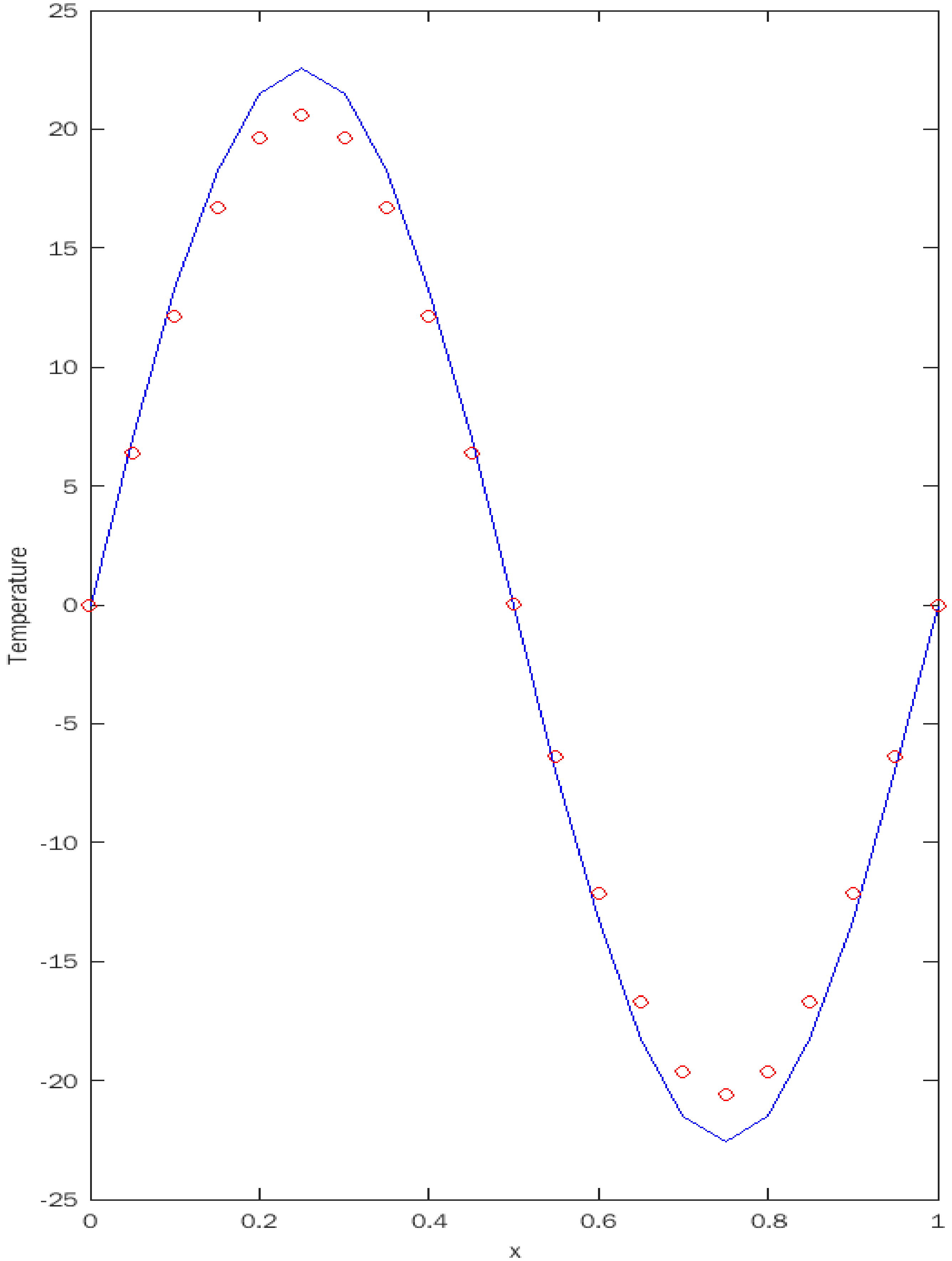
```
clc;
clear ;
close all;
%%below values is for 1st problem change it according the new question
x_0 = 0; x_end = 1;    % for range(length)
N = 40;        % No. of nodes
dx = (x_end-x_0)/N;    % distance between two nodes
t_final=0.04; %final time at which temperature values has to derive
a = 0.25;      % condition given in question dt/(dx)^2=0.25
dt = a*(dx*dx);    % step size in time
m=t_final/dt;
c_max=round(m); % this is final value for count
x = x_0:dx:x_end;     % x axis to plot

T_old = zeros(N+1,1);     % store the old values of T-temperature
T_new = zeros(N+1,1);     % store the new values of T-temperature
```

```
T_a = zeros(N+1,1); % T_a = T-exact from anaylytical solution
for j=1:N+1
    T_a(j)=100*exp(-4*pi*pi*t_final)*sin(2*pi*(j-1)*dx); %Analytical equation
end

c=0;
t = 0;
for k=1:N+1
    T_old(k)=100*sin(2*pi*(k-1)*dx);   % Initial values at t=0
end

while ( t <= t_final)
    [p,q,r,s] = matrix(N,a,T_old); % calling matrix function
    T_new = matrix_solver(p,q,r,s,N); % calling matrix_solver function
    %T_new is T-approximate
    T_old = T_new;    % give new value of temperature for next iteration

    c=c+1;
    t= t + dt;

    if(c==c_max)
        plot(x,T_new,'b',x,T_a,'or') %plot the graph of T-exact & T-apprpximate at
t_final
        xlabel('x'), ylabel('Temperature');
    end

    % to caluclate RMSE
    braket=zeros(N+1,1);
    for s = 1:N+1
        braket(s) = (T_a(s)-T_new(s))^2;
    end
    RMSE = sqrt(sum(braket,1)/N);
end
```

for dt/dx^2 = 0.25 graph of T_approximate vs x

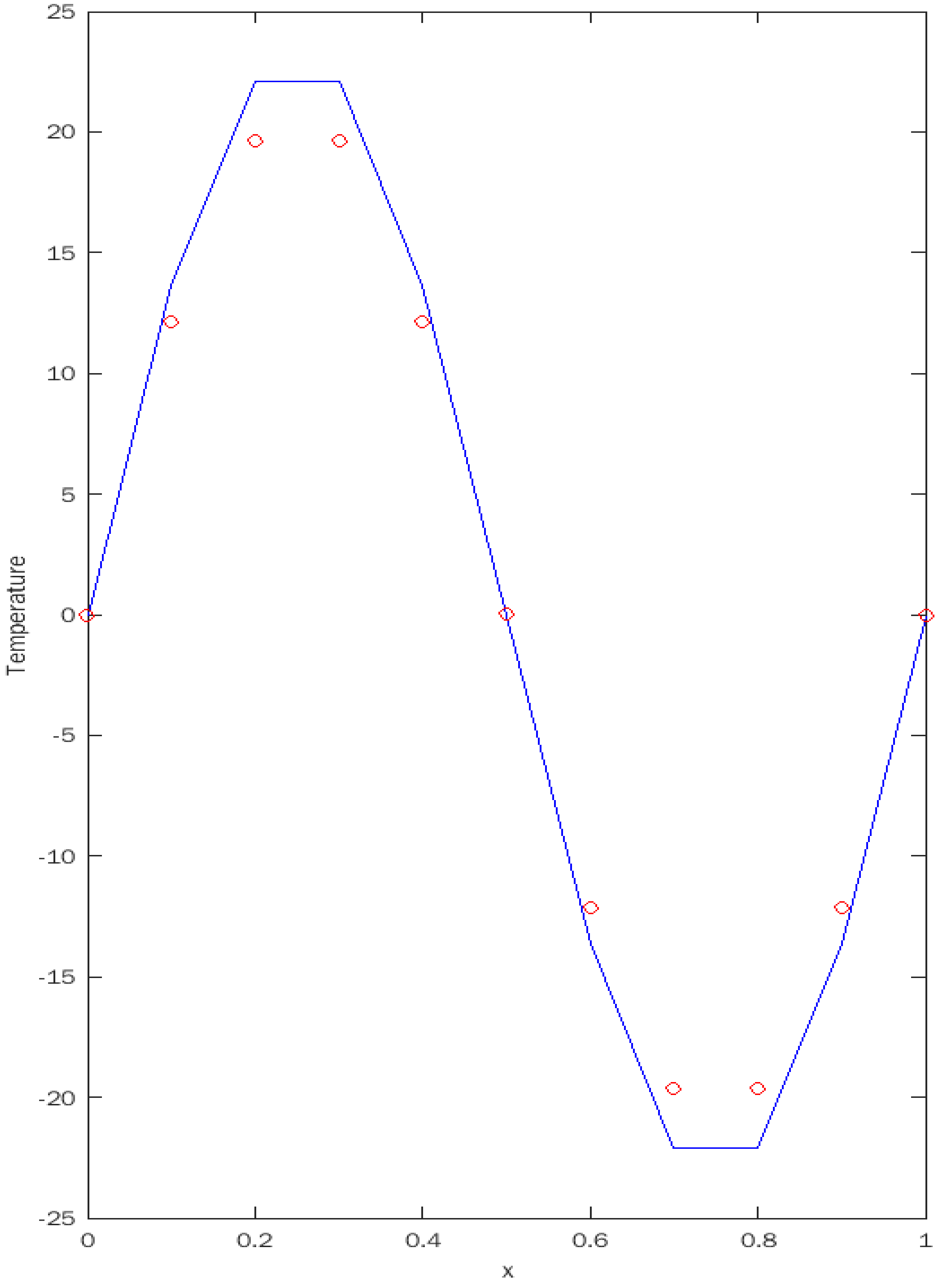


for dt/dx^2 = 0.75 graph of T_approximate vs x

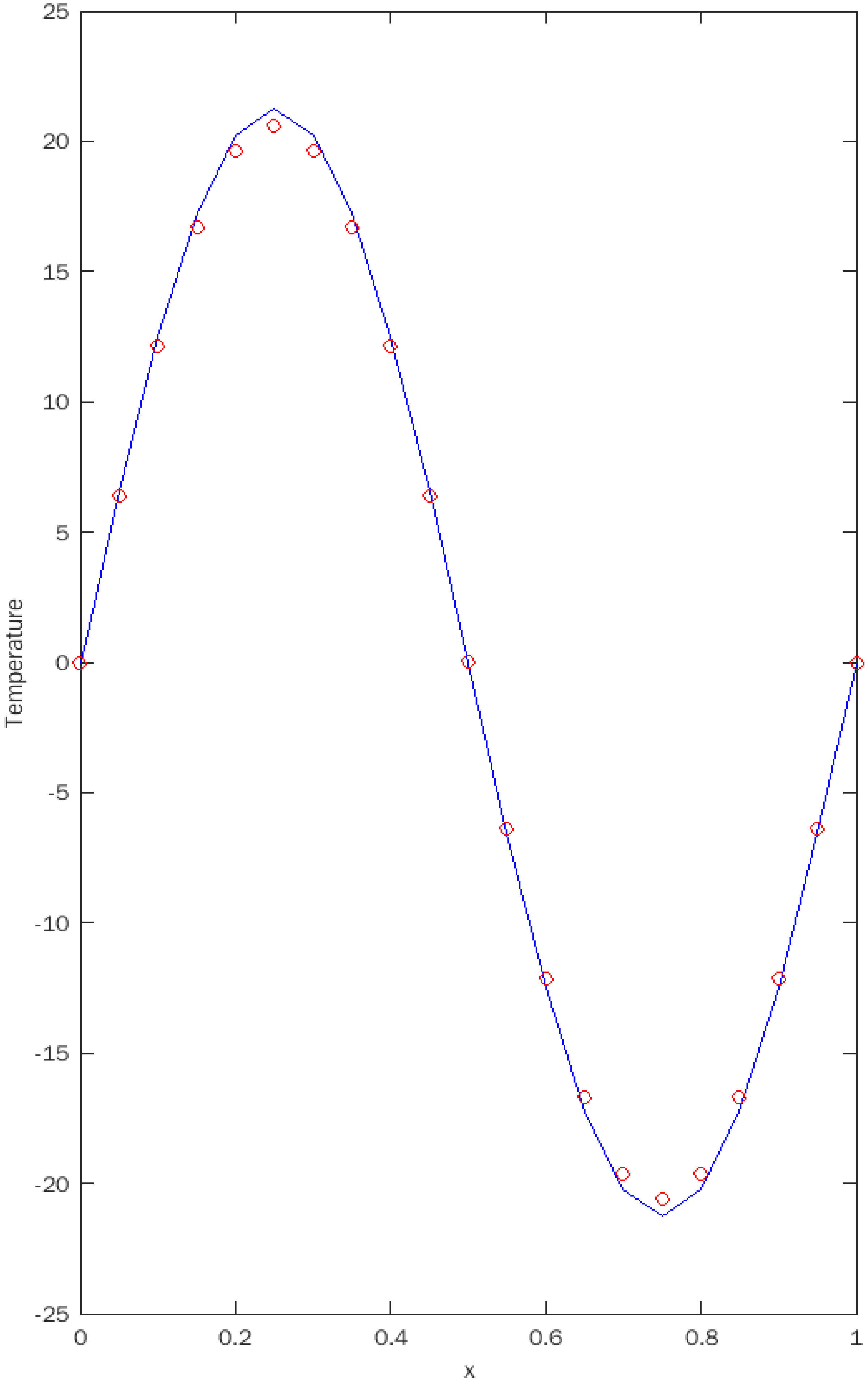


Question 1 partB:-by using above codes

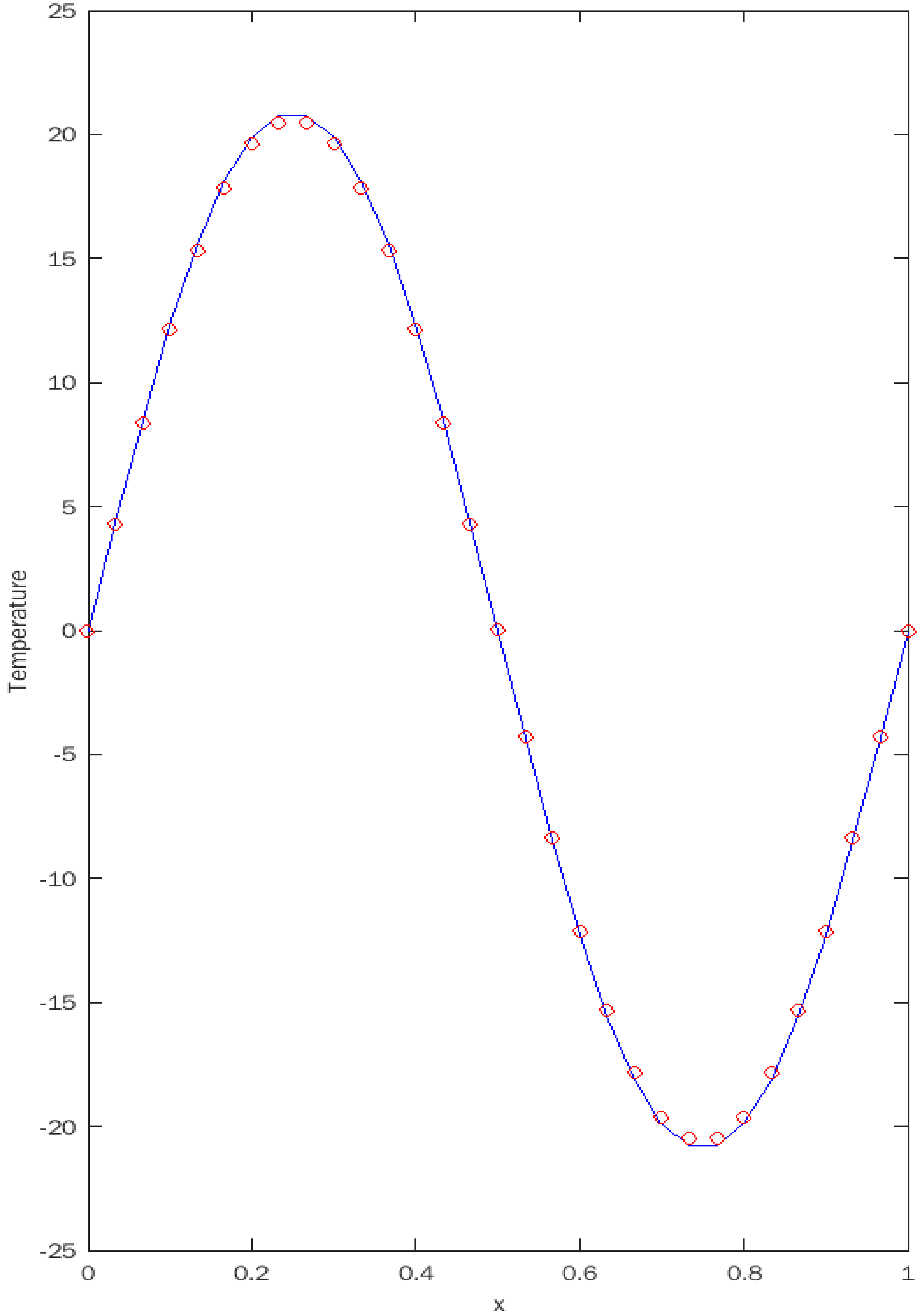
FOR N =10, RMSE = 1.8567



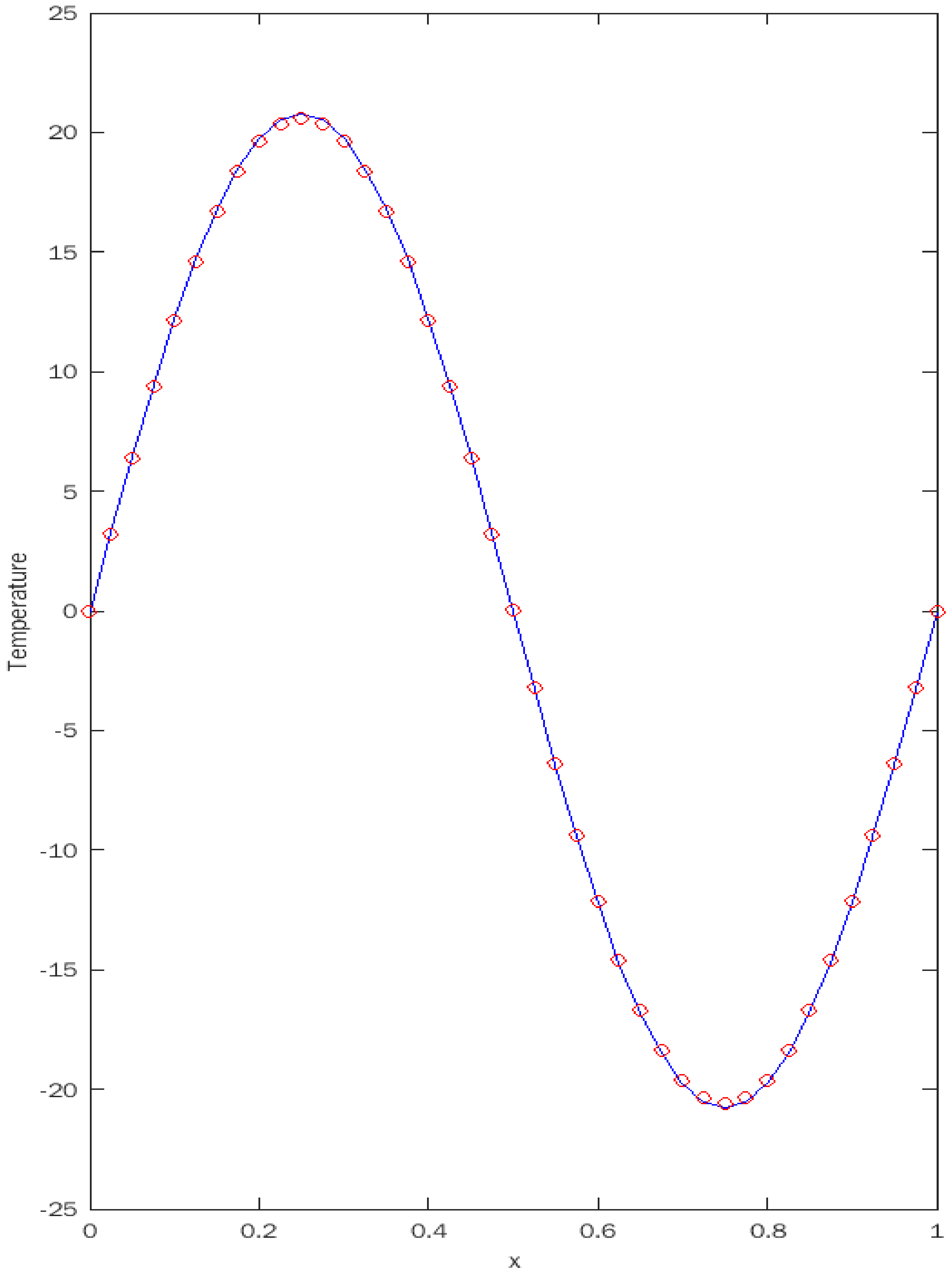
FOR N =20, RMSE = 0.4710



FOR N =30, RMSE = 0.0501



FOR N =40, RMSE = 0.0283



Question 2 partA

```
%save the first two box in matlab 1st as matrix & 2nd as matrix_solver filename
then run the last code file
```

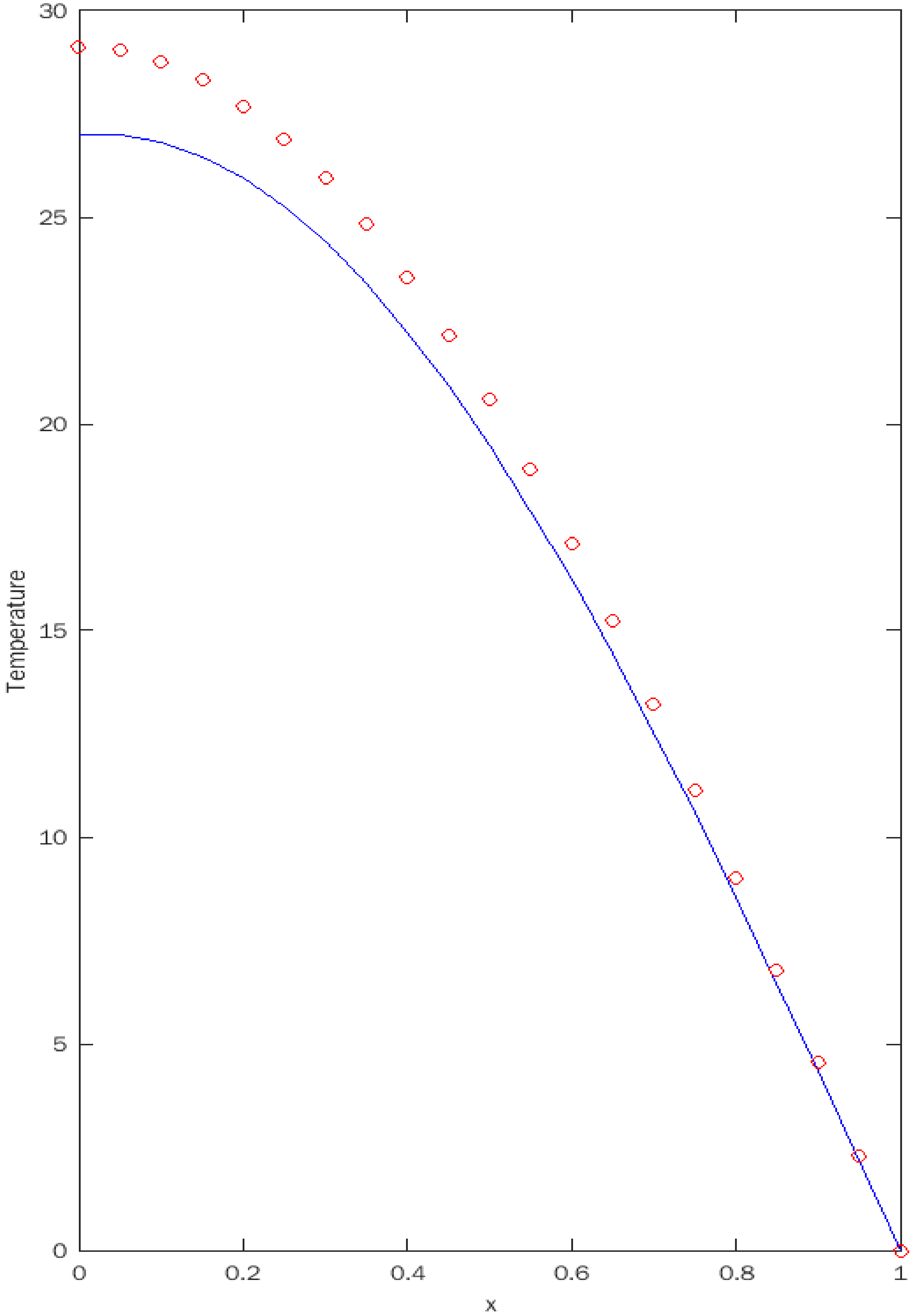
```
function [p,q,r,s] = matrix2(N,a,T_old)

p = zeros(N+1,1);
q = zeros(N+1,1);
r = zeros(N+1,1);
s = zeros(N+1,1);

%intial elements using boundary conditions
p(1)=0; q(1)=1;r(1)=-1;s(1)=0;
p(N+1)=0;q(N+1)=1;r(N+1)=0;s(N+1)=0;

%make diagonal elements of matrix
for i=2:N
    p(i) = a;
    q(i) = -(1+2*a);
    r(i) = a;
    s(i) = -T_old(i);
end
end
```

```
%to solve matrix
function A = matrix_solver(p,q,r,s,N)
A=zeros(N+1,1);
for i=1:N
    U=p(i+1)/q(i);
    p(i+1)=0;
    q(i+1)=q(i+1)-U*r(i);
    s(i+1)=s(i+1)-U*s(i);
end

A(N+1,1)=s(N+1)/q(N+1);
for i=N:-1:1
    A(i)=(s(i)-r(i)*A(i+1))/q(i);
end
end
```

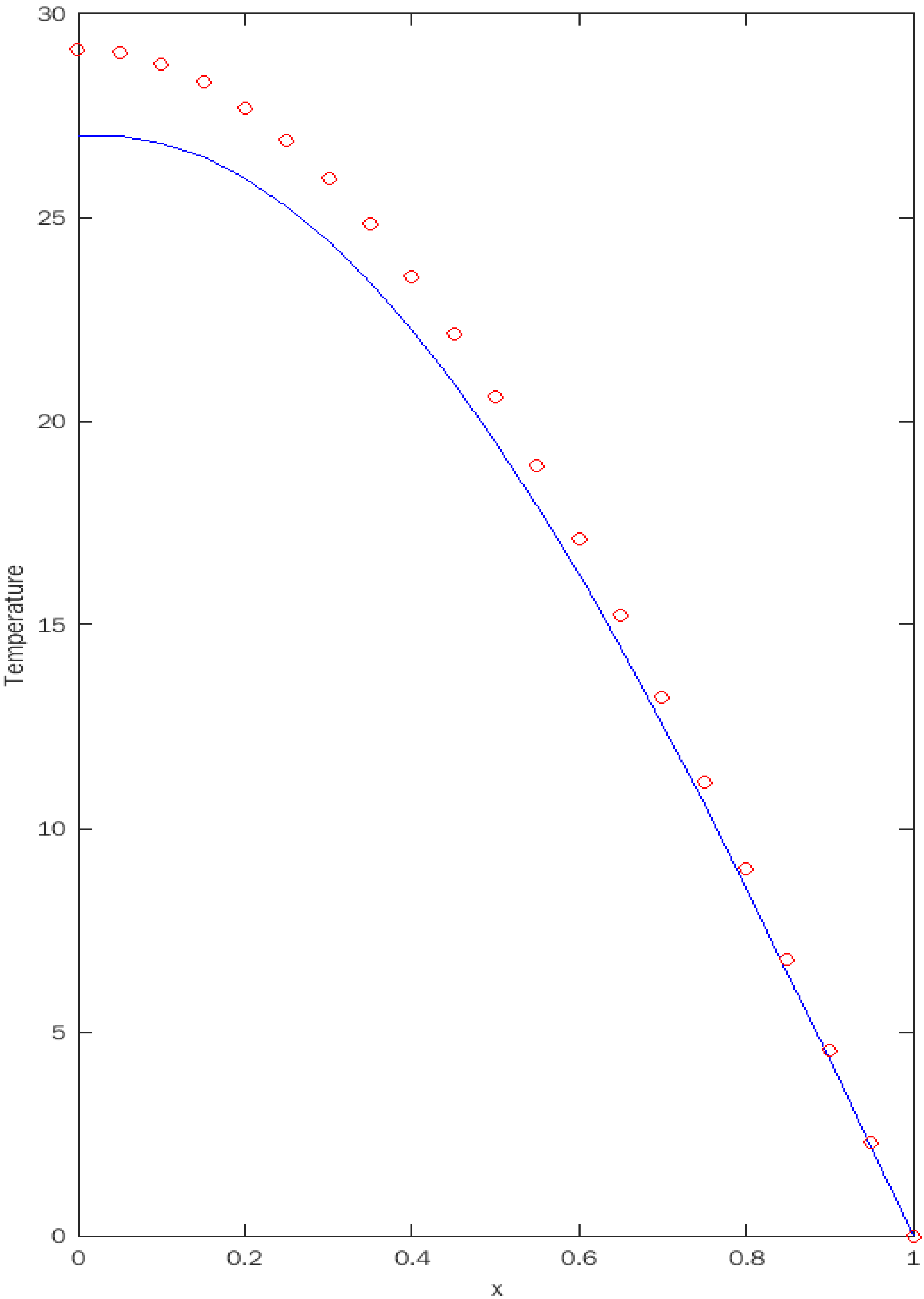
```
clc;
clear;
close all;
%%below values is for 1st problem change it according the new question
x_0 = 0; x_end = 1;  % for range(length)
N = 20;        % No. of nodes
dx = (x_end-x_0)/N;    % distance between two nodes
t_final=0.5; %final time at which temperature values has to derive
a = 0.25;     % condition given in question dt/(dx)^2=0.25
dt = a*(dx*dx);   % step size in time
m=t_final/dt;
c_max=round(m); % this is final value for count
x = x_0:dx:x_end;    % x-axis for plot

T_a = zeros(N+1,1); % T_a = T-exact from anaylytical solution
for j=1:N+1
```

```
    T_a(j)=100*exp(-0.25*pi*pi*t_final)*cos(0.5*pi*(j-1)*dx); %Analytical equation
end

T_old = zeros(N+1,1);    % store the old values of T-temperature
T_new = zeros(N+1,1);    % store the new values of T-temperature

c=0;
t = 0;
for k=1:N+1
    T_old(k)=100*cos(0.5*pi*(k-1)*dx);   %  Initial condition at t=0
end

while ( t <= t_final)
    [p,q,r,s] = matrix2(N,a,T_old); % calling matrix function
    T_new = matrix_solver(p,q,r,s,N); % calling matrix_solver function

    T_old = T_new;    % give new value of temperature for next iteration
    %T_new is T-approximate
    c=c+1;
    t= t + dt;

    T(c,:)=T_new;

    if(c==c_max)
        plot(x,T_new,'b',x,T_a,'or') %plot the graph of T-exact & T-apprpximate at
t_final
        xlabel('x'), ylabel('Temperature');
    end

    % to caluclate RMSE
    braket=zeros(N+1,1);
    for s = 1:N+1
        braket(s) = (T_a(s)-T_new(s))^2;
    end
    RMSE = sqrt(sum(braket,1)/N);
end
```

for dt/dx^2 = 0.25 graph of T_approximate vs x

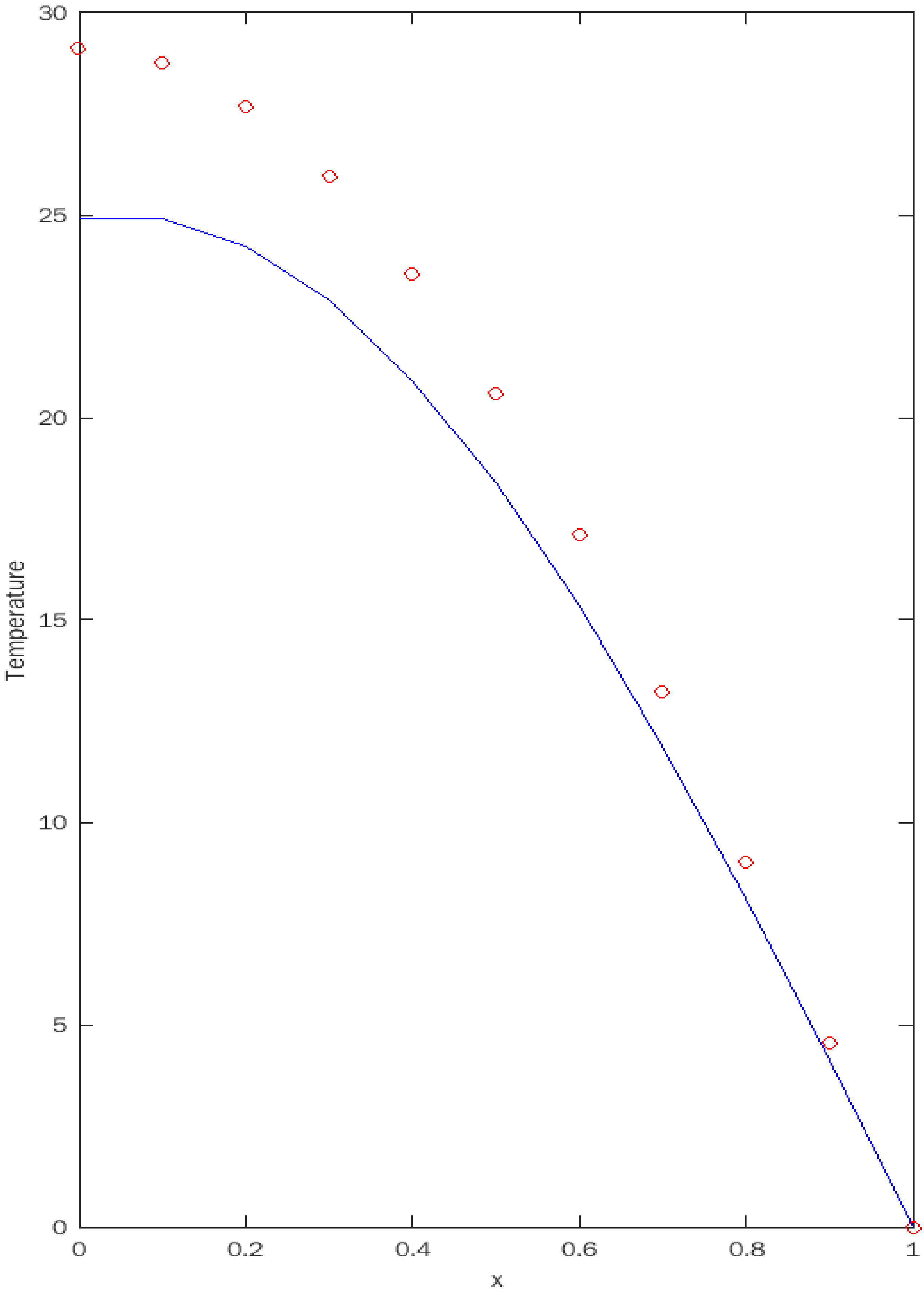


for dt/dx^2 = 0.75 graph of T_approximate vs x

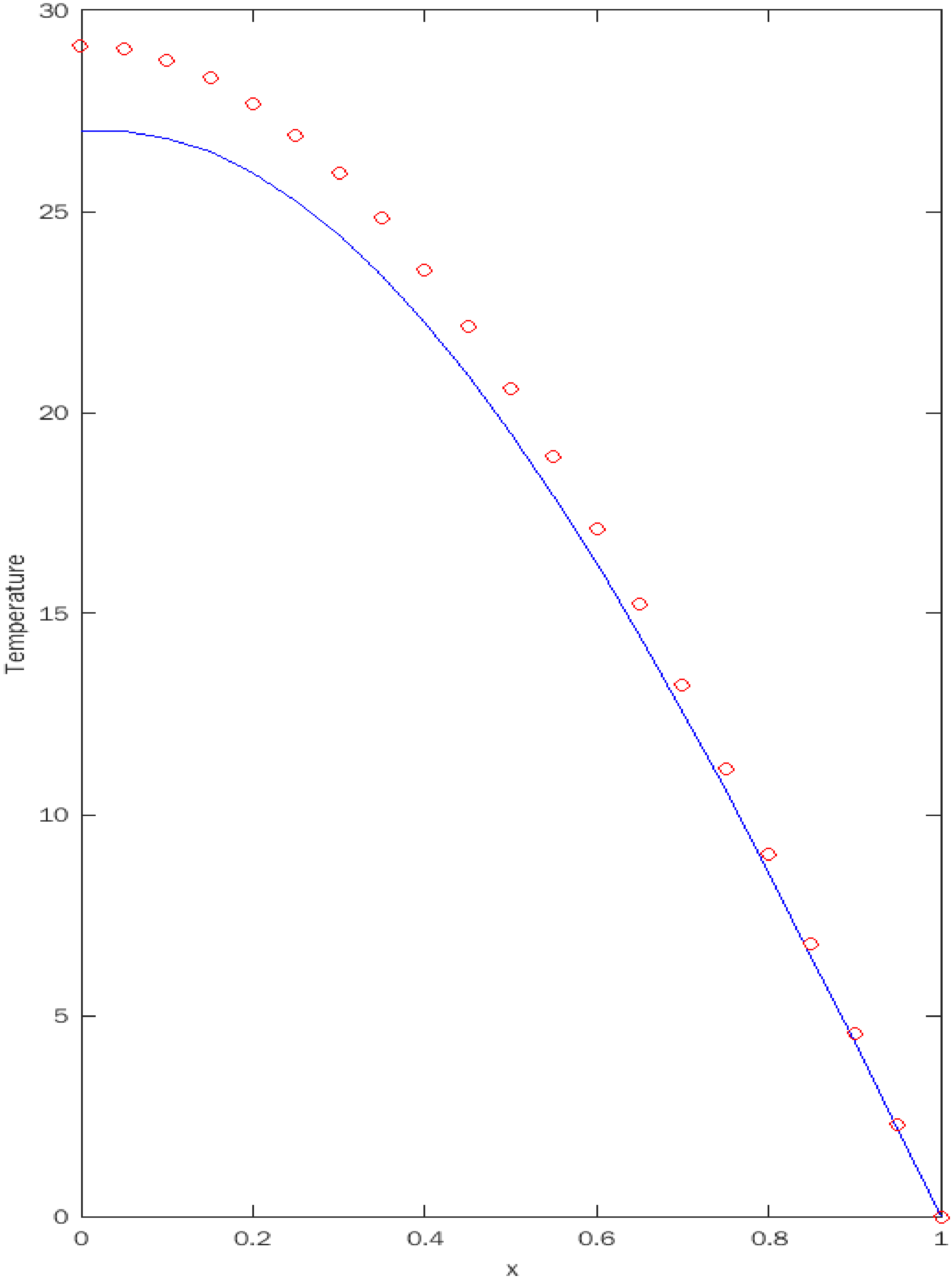


Question 2 partB:-by using above codes

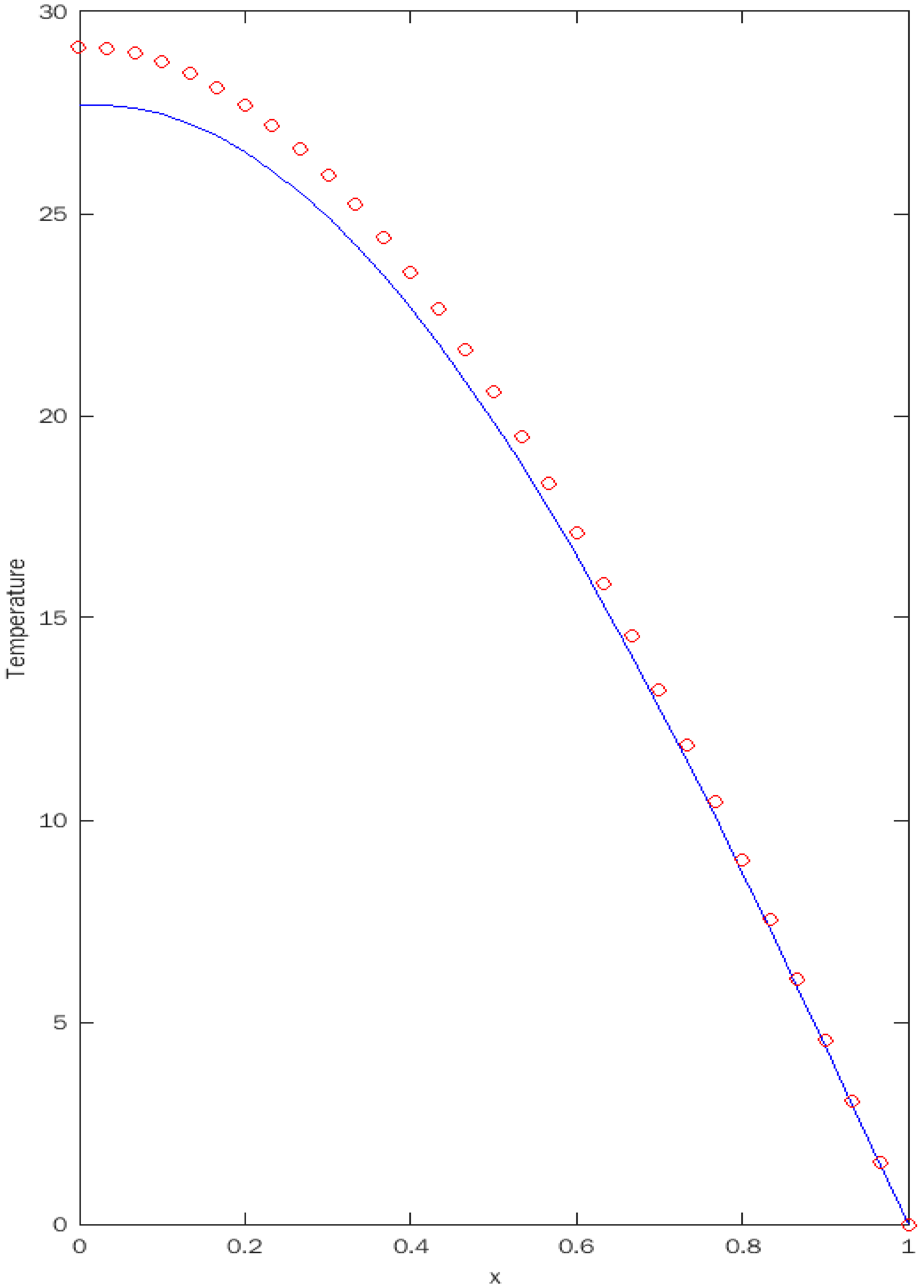
FOR N =10, RMSE = 2.6729



FOR N =20, RMSE = 1.3418



FOR N =30, RMSE = 0.8807



FOR N =40, RMSE = 0.6473