ME630 Assignment 1

Aman Parekh (180073)

August 27, 2021

Problem 1

Exact Solution

$$\frac{d\bar{y}}{dx} = \bar{y}$$
$$\implies \int \frac{d\bar{y}}{\bar{y}} = \int dx$$
$$\implies ln\bar{y} = x + c$$

Using Boundary Condition: $\bar{y}(0) = 1$, we get $c = 0$,

$$\boxed{\therefore \bar{y} = e^x}$$

Moving further, we will refer to the exact solution as y instead of $\bar{y}$.

Set up of the FEM Solution

Let $\boxed{\bar{y} = 1 + \sum_{j=1}^{N} a_j x^j}$, where $\bar{y}$ is the approximate solution, 1 refers to the solution at the boundary condition $x = 0$, N refers to the discretization parameter, x^j refers to the polynomial functions that will serve as the trial functions, and a_j are the unknown constants to be determined.
The residue is given by:

$$R = \frac{d\bar{y}}{dx} - \bar{y}$$
$$\implies R = \sum_{j=1}^{N} a_j j x^{j-1} - \sum_{j=1}^{N} a_j x^j - 1$$
$$\implies R = -1 + \sum_{j=1}^{N} a_j (j x^{j-1} - x^j)$$

Now, applying the weighted residual method, where our weights will also belong to the same family of polynomials as the trial functions (Galerkin) as $w_m = x^{m-1}$:

$$\int_0^1 w_m R dx = 0$$
$$\boxed{\int_0^1 x^{m-1} \left(-1 + \sum_{j=1}^{N} a_j (j x^{j-1} - x^j) \right) dx = 0}$$

Now, depending upon our value of N, the solution (values of $a_j s$) will vary as follows:

N=1 (m will only take the value of 1)

$$\bar{y} = 1 + a_1 x$$

For $m = 1$:

$$\int_0^1 x^{m-1}\left(-1+\sum_{j=1}^{N} a_j(jx^{j-1}-x^j)\right)dx = 0$$

$$\implies \int_0^1 x^{1-1}\left(-1+\sum_{j=1}^{1} a_j(jx^{j-1}-x^j)\right)dx = 0$$

$$\implies \int_0^1 (-1 + a_1(1-x))\,dx = 0$$

$$\implies \int_0^1 ((a_1-1) - a_1 x)\,dx = 0$$

$$\implies (a_1-1)\,[x]_0^1 - a_1\left[\frac{x^2}{2}\right]_0^1 = 0$$

$$\implies a_1 - 1 - \frac{a_1}{2} = 0$$

$$\implies a_1 = 2$$

$$\boxed{\textbf{Solution for N=1: } \bar{y} = 1 + 2x}$$

N=2 (m will vary from 1-2)

$$\bar{y} = 1 + a_1 x + a_2 x^2$$

For $m = 1$:

$$\int_0^1 x^{m-1}\left(-1+\sum_{j=1}^{N} a_j(jx^{j-1}-x^j)\right)dx = 0$$

$$\implies \int_0^1 x^{1-1}\left(-1+\sum_{j=1}^{2} a_j(jx^{j-1}-x^j)\right)dx = 0$$

$$\implies \int_0^1 \left(-1 + a_1(1-x) + a_2(2x-x^2)\right)dx = 0$$

$$\implies \int_0^1 \left((a_1-1) + (2a_2-a_1)x - a_2x^2\right)dx = 0$$

$$\implies (a_1-1)\,[x]_0^1 + (2a_2-a_1)\left[\frac{x^2}{2}\right]_0^1 - a_2\left[\frac{x^3}{3}\right]_0^1 = 0$$

$$\implies a_1 - 1 + a_2 - \frac{a_1}{2} - \frac{a_2}{3} = 0$$

$$\implies 3a_1 + 4a_2 = 6\textbf{; eq. (A)}$$

For $m = 2$:

$$\int_0^1 x^{m-1} \left(-1 + \sum_{j=1}^{N} a_j (jx^{j-1} - x^j) \right) dx = 0$$

$$\implies \int_0^1 x^{2-1} \left(-1 + \sum_{j=1}^{2} a_j (jx^{j-1} - x^j) \right) dx = 0$$

$$\implies \int_0^1 x \left(-1 + a_1(1-x) + a_2(2x - x^2) \right) dx = 0$$

$$\implies \int_0^1 \left((a_1 - 1)x + (2a_2 - a_1)x^2 - a_2 x^3 \right) dx = 0$$

$$\implies (a_1 - 1) \left[\frac{x^2}{2} \right]_0^1 + (2a_2 - a_1) \left[\frac{x^3}{3} \right]_0^1 - a_2 \left[\frac{x^4}{4} \right]_0^1 = 0$$

$$\implies \frac{a_1 - 1}{2} + \frac{2a_2 - a_1}{3} - \frac{a_2}{4} = 0$$

$$\implies 2a_1 + 5a_2 = 6\textbf{; eq. (B)}$$

From equation A and B, we get

$$\begin{bmatrix} 3 & 4 \\ 2 & 5 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 6 \\ 6 \end{bmatrix}$$

Solving this, we get $a_1 = 0.8571$ and $a_2 = 0.8571$.

$$\boxed{\textbf{Solution for N=2: } \bar{y} = 1 + 0.8571x + 0.8571x^2}$$

N=4 (m will vary from 1-4)

$$\bar{y} = 1 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4$$

For $m = 1$:

$$\int_0^1 x^{m-1} \left(-1 + \sum_{j=1}^{N} a_j (jx^{j-1} - x^j) \right) dx = 0$$

$$\implies \int_0^1 x^{1-1} \left(-1 + \sum_{j=1}^{2} a_j (jx^{j-1} - x^j) \right) dx = 0$$

$$\implies \int_0^1 \left(-1 + a_1(1-x) + a_2(2x - x^2) + a_3(3x^2 - x^3) + a_4(4x^3 - x^4) \right) dx = 0$$

$$\implies \int_0^1 \left((a_1 - 1) + (2a_2 - a_1)x + (3a_3 - a_2)x^2 + (4a_4 - a_3)x^3 - a_4 x^4 \right) dx = 0$$

$$\implies (a_1 - 1) [x]_0^1 + (2a_2 - a_1) \left[\frac{x^2}{2} \right]_0^1 + (3a_3 - a_2) \left[\frac{x^3}{3} \right]_0^1 + (4a_4 - a_3) \left[\frac{x^4}{4} \right]_0^1 - a_4 \left[\frac{x^5}{5} \right]_0^1 = 0$$

$$\implies a_1 - 1 + a_2 - \frac{a_1}{2} + a_3 - \frac{a_2}{3} + a_4 - \frac{a_3}{4} - \frac{a_4}{5} = 0$$

$$\implies \frac{a_1}{2} + \frac{2a_2}{3} + \frac{3a_3}{4} + \frac{4a_4}{5} = 1\textbf{; eq. (A)}$$

Going forward, the integral evaluation for individual cases is not done, but the final result is written.

$$\textbf{m=2:}\ \frac{a_1}{6}+\frac{5a_2}{12}+\frac{11a_3}{20}+\frac{19a_4}{30}=\frac{1}{2}\textbf{; eq. (B)}$$

$$\textbf{m=3:}\ \frac{a_1}{12}+\frac{3a_2}{10}+\frac{13a_3}{30}+\frac{11a_4}{21}=\frac{1}{3}\textbf{; eq. (C)}$$

$$\textbf{m=4:}\ \frac{a_1}{20}+\frac{7a_2}{30}+\frac{5a_3}{14}+\frac{25a_4}{56}=\frac{1}{4}\textbf{; eq. (D)}$$

From equation A, B, C and D we get

$$\begin{bmatrix} 1/2 & 2/3 & 3/4 & 4/5 \\ 1/6 & 5/12 & 11/20 & 19/30 \\ 1/12 & 3/10 & 13/30 & 11/21 \\ 1/20 & 7/30 & 5/14 & 25/56 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1/2 \\ 1/3 \\ 1/4 \end{bmatrix}$$

Solving this, we get $a_1 = 0.99$, $a_2 = 0.5095$, $a_3 = 0.1399$ and $a_4 = 0.0699$

$$\boxed{\textbf{Solution for N=4: } \bar{y} = 1 + 0.99x + 0.5095x^2 + 0.1399x^3 + 0.0699x^4}$$

<u>**N=5**</u> (m will vary from 1-5)

$$\bar{y} = 1 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5$$

After arriving at the linear equations in 5 unknowns, we get the following matrix:

$$\begin{bmatrix} 1/2 & 2/3 & 3/4 & 4/5 & 5/6 \\ 1/6 & 5/12 & 11/20 & 19/30 & 29/42 \\ 1/12 & 3/10 & 13/30 & 11/21 & 33/56 \\ 1/20 & 7/30 & 5/14 & 25/56 & 37/72 \\ 1/30 & 4/21 & 17/56 & 7/18 & 41/90 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} 1 \\ 1/2 \\ 1/3 \\ 1/4 \\ 1/5 \end{bmatrix}$$

Solving this, we get $a_1 = 1.001$, $a_2 = 0.4992$, $a_3 = 0.1703$, $a_4 = 0.0348$ and $a_5 = 0.0139$

$$\boxed{\textbf{Solution for N=5: } \bar{y} = 1 + 1.001x + 0.4992x^2 + 0.1703x^3 + 0.0348x^4 + 0.0139x^5}$$

<u>**N=7**</u> (m will vary from 1-7)

$$\bar{y} = 1 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5 + a_6x^6 + a_7x^7$$

After arriving at the linear equations in 7 unknowns, we get the following matrix:

$$\begin{bmatrix} 1/2 & 2/3 & 3/4 & 4/5 & 5/6 & 6/7 & 7/8 \\ 1/6 & 5/12 & 11/20 & 19/30 & 29/42 & 41/56 & 55/72 \\ 1/12 & 3/10 & 13/30 & 11/21 & 33/56 & 23/36 & 61/90 \\ 1/20 & 7/30 & 5/14 & 25/56 & 37/72 & 17/30 & 67/110 \\ 1/30 & 4/21 & 17/56 & 7/18 & 41/90 & 28/55 & 73/132 \\ 1/42 & 9/56 & 19/72 & 31/90 & 9/22 & 61/132 & 79/156 \\ 1/56 & 5/36 & 7/30 & 17/55 & 49/132 & 11/26 & 85/182 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \end{bmatrix} = \begin{bmatrix} 1 \\ 1/2 \\ 1/3 \\ 1/4 \\ 1/5 \\ 1/6 \\ 1/7 \end{bmatrix}$$

Solving this, we get $a_1 = 1$, $a_2 = 0.5$, $a_3 = 0.1667$, $a_4 = 0.0416$, $a_5 = 0.0085$, $a_6 = 0.0012$ and $a_7 = 0.0003$

$$\boxed{\textbf{Solution for N=7: } \bar{y} = 1 + x + 0.5x^2 + 0.1667x^3 + 0.0416x^4 + 0.0085x^5 + 0.0012x^6 + 0.0003x^7}$$

N	$\bar{y}$
1	$1 + 2x$
2	$1 + 0.8571x + 0.8571x^2$
4	$1 + 0.99x + 0.5095x^2 + 0.1399x^3 + 0.0699x^4$
5	$1 + 1.001x + 0.4992x^2 + 0.1703x^3 + 0.0348x^4 + 0.0139x^5$
7	$1 + x + 0.5x^2 + 0.1667x^3 + 0.0416x^4 + 0.0085x^5 + 0.0012x^6 + 0.0003x^7$

For the following plots, the blue line shows the plot where x steps are taken in 0.1 increment, but it was observed it did not capture the variation in error adequately, so the red line is also plotted where x steps are in increment of 0.01. (**In plotting these, the value of coefficients upto 16 decimal places was considered, rounding off leads to spurious results.**)

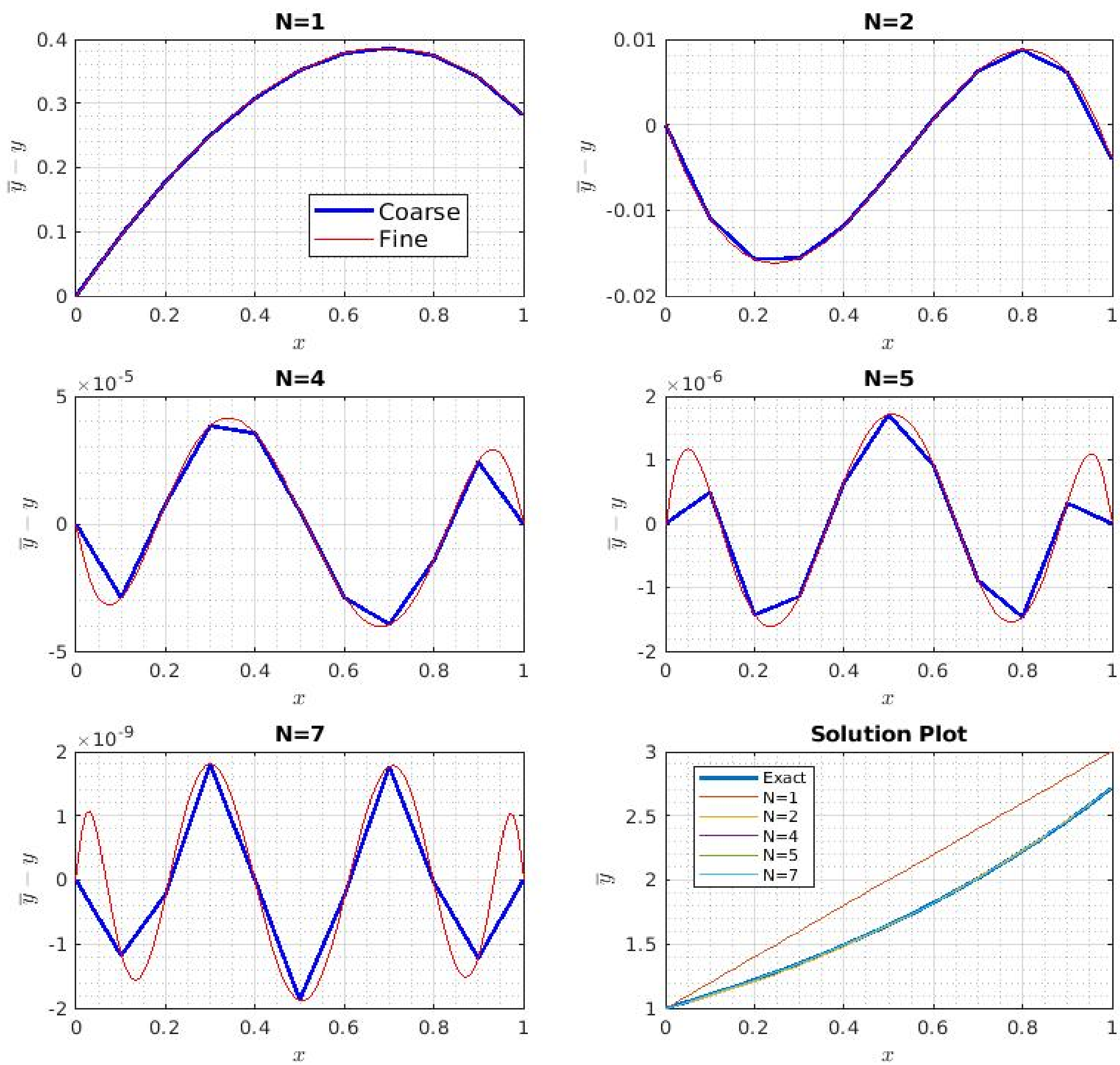

Figure 1: Plot of $\bar{y} - y$ vs x

Clearly, the error is least for $N = 7$. There is a massive increase in accuracy as we move from $N = 1$ to $N = 2$ as the solution almost overlaps with the exact solution. The reason to show the Solution Plot in Figure 1 is to portray that in the range $[0, 1]$, the solution has good accuracy for $N = 2$ and above.

Also, we know that e^x is an infinite series as follows:

$$\boxed{e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + ...}$$

It is the observed that as N increases, the coefficients of the polynomial terms closely resemble the coefficients in the infinite series.

Problem 2a

$$\boxed{\frac{d\phi}{dt} = -\phi} \quad ; \quad \phi(0) = 1$$

Exact Solution: $\phi = e^{-t}$

EXPLICIT EULER SOLUTION

From Taylor Series Expansion,

$$\phi_{n+1} = \phi_n + \Delta t \frac{d\phi}{dt} + \frac{\Delta t^2}{2!}\frac{d^2\phi}{dt^2} + O(\Delta t^3)$$

$$\implies \phi_{n+1} = \phi_n + \Delta t(\phi_n') + O(\Delta t^2)$$

$$\implies \phi_{n+1} = \phi_n + \Delta t(-\phi_n) + O(\Delta t^2)$$

$$\implies \boxed{\phi_{n+1} = (1 - \Delta t)\phi_n}$$

```
%% Explicit Euler for 2a
T = 15;

dt = 0.1 ;      % For timestep = 0.1
time = 0:dt:T;
n = T/dt;      % Number of timesteps

phi1 = zeros(ceil(n),1);       % Initializing Solution to 0
phi1(1) = 1;             % Assigning Initial Condition at x=0

for i = 1:n-1
    phi1(i+1) = phi1(i) - dt*phi1(i);
end

t = 0:0.1:T;
plot(t, exp(-t), 'r','LineWidth', 2);
hold on;
plot(time(1:length(phi1)), phi1, 'bo-', 'MarkerFaceColor', 'b');
xlabel('x');
xlim([0 10]);
ylabel('\phi');
legend('Exact Solution', 'Explicit Solution for dt=0.1')
grid on;
grid minor;
```

The above code is shown for $\Delta t = 0.1$. This can be changed to arrive at the plots for different timesteps, as shown in the following plots: (**Note: the x-axis in the following plots should actually be 't' instead of 'x'**)

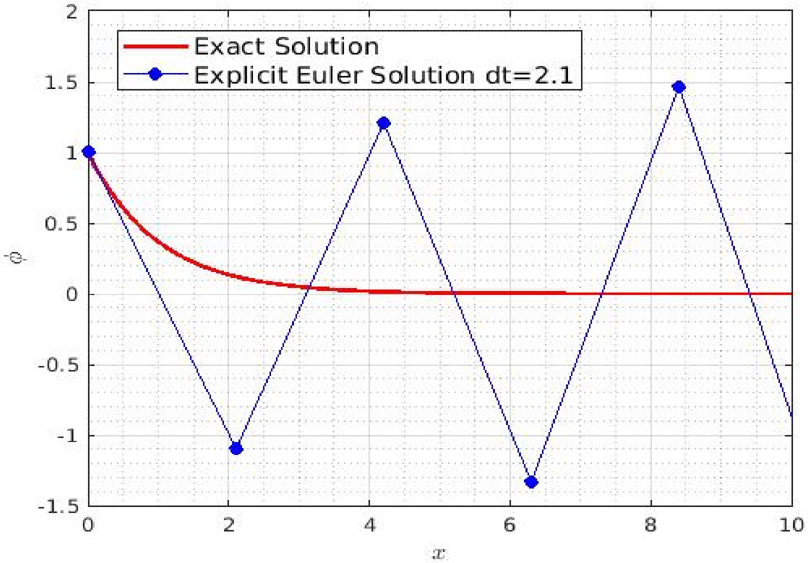

Figure 2: Explicit Euler for $\Delta t = 2.1$

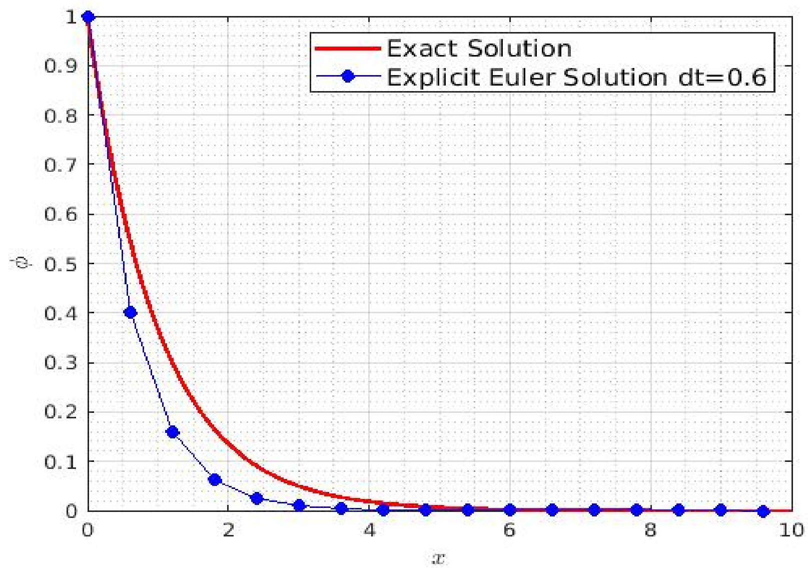

Figure 3: Explicit Euler for $\Delta t = 0.6$

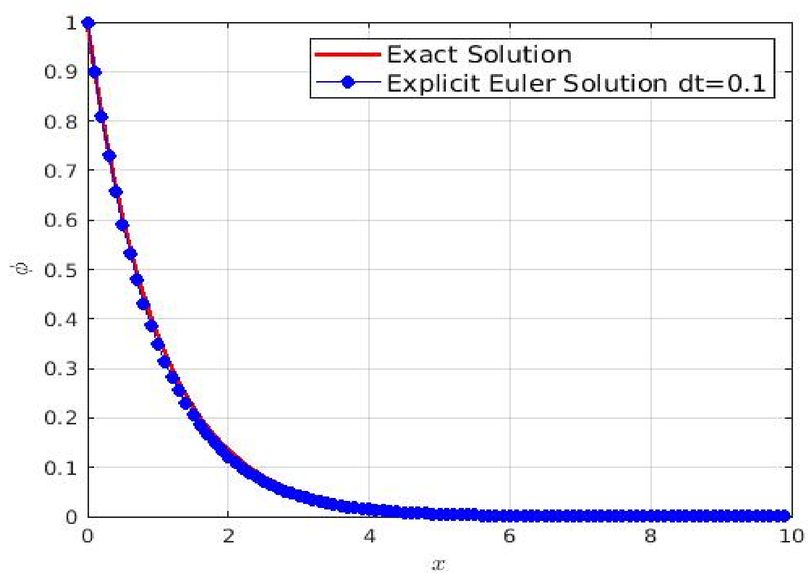

Figure 4: Explicit Euler for $\Delta t = 0.1$

OBSERVATIONS

Clearly, for $\Delta t = 2.1$, the solution is not stable, as explicit euler scheme is conditionally stable. For it to be stable, $\Delta t \leq {}^{-2}/_{-1}$. Since $\Delta t = 2.1$ is greater than the timestep required for conditional stability, the solution diverges.

Also, as the the timestep decreases, the accuracy increases.

IMPLICIT EULER SOLUTION

From Taylor Series Expansion,

$$\phi_n = \phi_{n+1} - \Delta t \frac{d\phi}{dt} + \frac{\Delta t^2}{2!}\frac{d^2\phi}{dt^2} + O(\Delta t^3)$$

$$\implies \phi_n = \phi_{n+1} - \Delta t(\phi'_{n+1}) + O(\Delta t^2)$$

$$\implies \phi_n = \phi_{n+1} - \Delta t(-\phi_{n+1}) + O(\Delta t^2)$$

$$\implies \boxed{\phi_{n+1} = \frac{1}{1+\Delta t}\phi_n}$$

```
%% Implicit Euler for 2a
T = 15;

dt = 0.1 ;      % For timestep = 0.1
time = 0:dt:T;
n = T/dt;     % Number of timesteps

phi1 = zeros(ceil(n),1);      % Initializing Solution to 0
phi1(1) = 1;            % Assigning Initial Condition at x=0

for i = 1:n-1
    phi1(i+1) = (1/(1+dt))*phi1(i);
end

t = 0:0.1:T;
plot(t, exp(-t), 'r','LineWidth', 2);
hold on;
plot(time(1:length(phi1)), phi1, 'bo-', 'MarkerFaceColor', 'b');
xlabel('x');
xlim([0 10]);
ylabel('\phi');
legend('Exact Solution', 'Implicit Solution for dt=0.1')
grid on;
grid minor;
```

The above code is shown for $\Delta t = 0.1$. This can be changed to arrive at the plots for different timesteps, as shown in the following plots: (**Note: the x-axis in the following plots should actually be 't' instead of 'x'**)

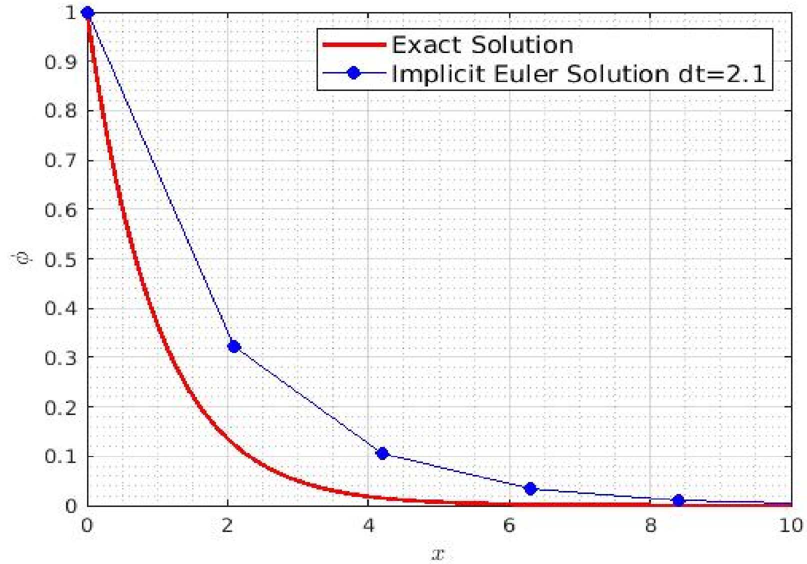

Figure 5: Implicit Euler for $\Delta t = 2.1$

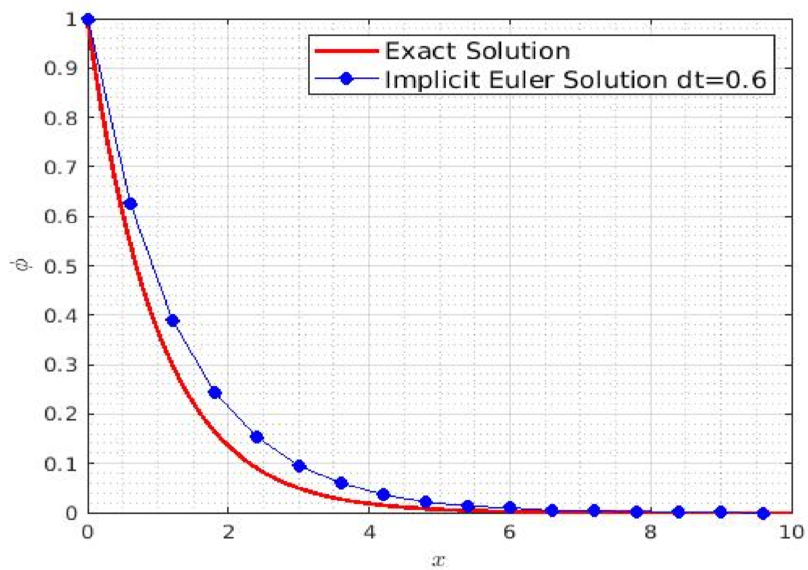

Figure 6: Implicit Euler for $\Delta t = 0.6$

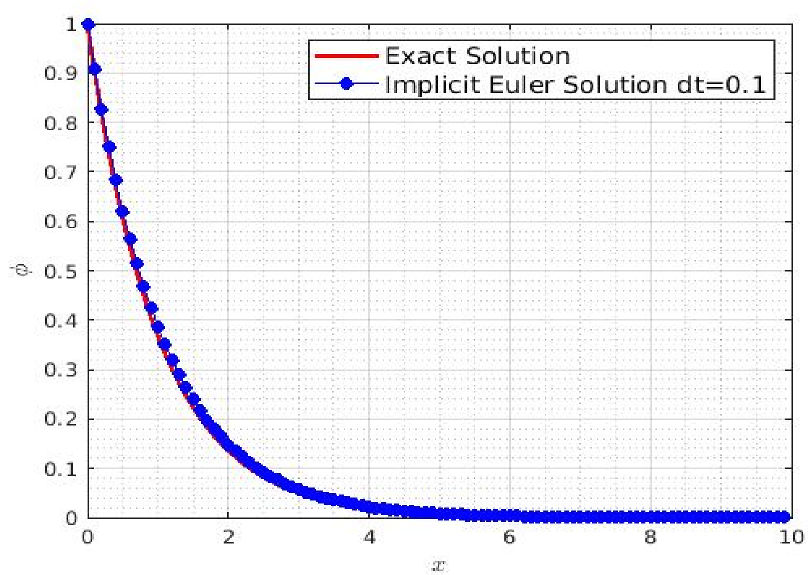

Figure 7: Implicit Euler for $\Delta t = 0.1$

OBSERVATIONS

The implicit euler scheme is unconditionally stable, therefore none of the above cases diverge. However, the solution becomes more accurate with decrease in timestep.

CRANK-NICOLSON SOLUTION

After applying Taylor Series expansion, we arrive at the following equation:

$$\phi_{n+1} = \phi_n + \frac{\Delta t}{2}(\phi'_n + \phi'_{n+1}) + O(\Delta t^3)$$

$$\implies \phi_{n+1} = \phi_n + \frac{\Delta t}{2}\left((-\phi_n) + (-\phi_{n+1})\right) + O(\Delta t^3)$$

$$\implies \boxed{\phi_{n+1} = \left(\frac{1 - \dfrac{\Delta t}{2}}{1 + \dfrac{\Delta t}{2}}\right)\phi_n}$$

```
%% Crank Nicolson for 2a
T = 15;

dt = 0.1 ;      % For timestep = 0.1
time = 0:dt:T;
n = T/dt;      % Number of timesteps

phi1 = zeros(ceil(n),1);      % Initializing Solution to 0
phi1(1) = 1;            % Assigning Initial Condition at x=0

for i = 1:n-1
    phi1(i+1) = ((1-(dt/2))/(1+(dt/2)))*phi1(i);
end

t = 0:0.1:T;
plot(t, exp(-t), 'r','LineWidth', 2);
hold on;
plot(time(1:length(phi1)), phi1, 'bo-', 'MarkerFaceColor', 'b');
xlabel('x');
xlim([0 10]);
ylabel('\phi');
legend('Exact Solution', 'Crank Nicolson Solution for dt=0.1')
grid on;
grid minor;
```

The above code is shown for $\Delta t = 0.1$. This can be changed to arrive at the plots for different timesteps, as shown in the following plots: (**Note: the x-axis in the following plots should actually be 't' instead of 'x'**)

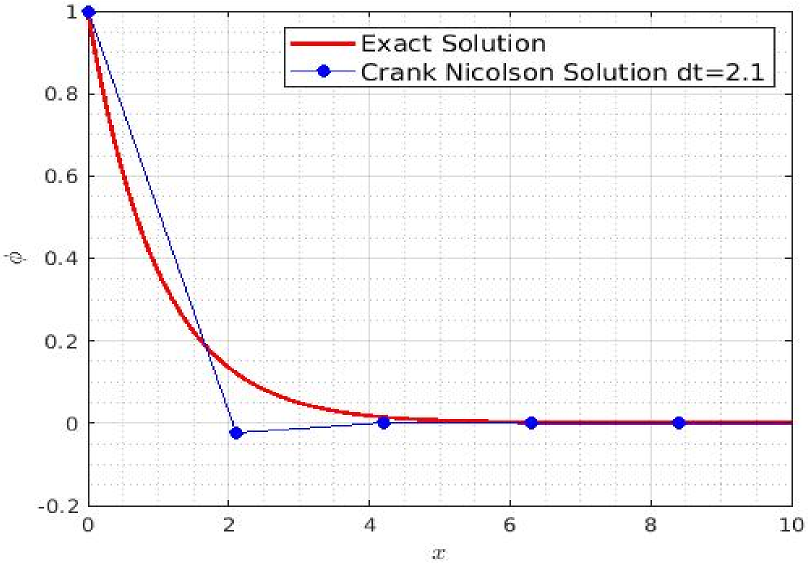

Figure 8: Crank Nicolson for $\Delta t = 2.1$

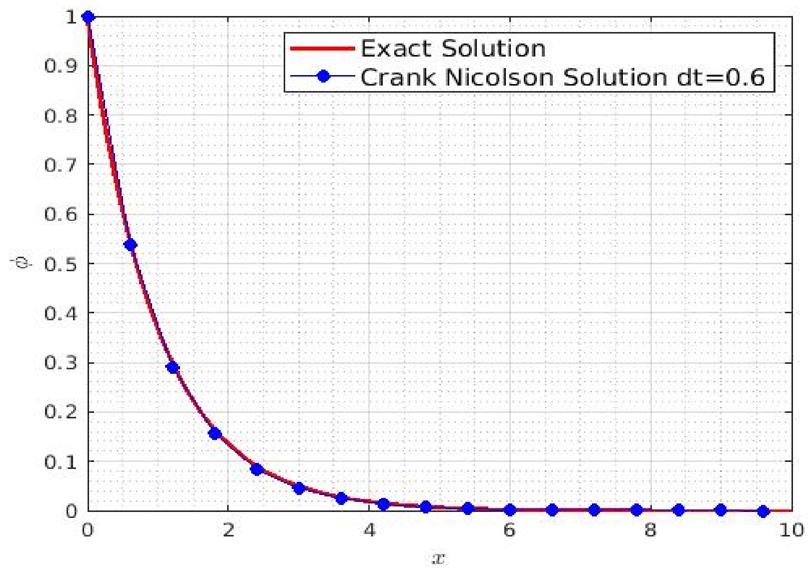

Figure 9: Crank Nicolson for $\Delta t = 0.6$

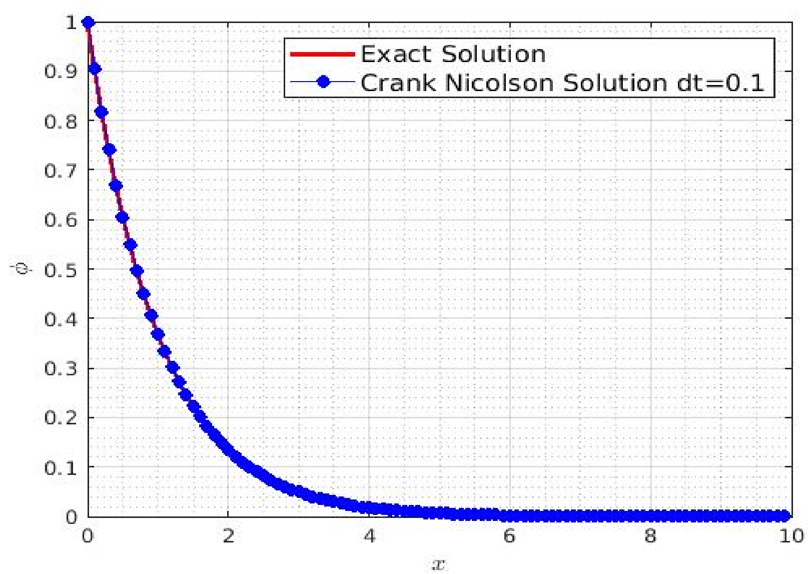

Figure 10: Crank Nicolson for $\Delta t = 0.1$

OBSERVATIONS

This scheme is unconditionally stable and as can be seen, performs better than the implicit scheme. With decrease in timstep, the accuracy increases.

RUNGE-KUTTA-3 SOLUTION

The RK-3 method follows these 3 steps to arrive at the solution. Variables are being re-used, therefore this scheme is the low storage scheme:

Step 1	**Step 2**	**Step 3**
$\phi^* = \phi_n$	$\phi^* = \phi^* + \frac{\Delta t}{3} K_1$	$\phi^* = \phi^* + \frac{15\Delta t}{16} K_1$
$K_1 = -\phi^*_n$	$K_1 = -\frac{5}{9} K_1 - \phi^*_n$	$K_1 = -\frac{153}{128} K_1 - \phi^*_n$

And finally: $\boxed{\phi_{n+1} = \phi^* + \frac{8\Delta t}{15} K_1}$

```
%% RK-3 (Low Storage) for 2a
T = 15;

dt = 0.1 ;      % For timestep = 0.1
time = 0:dt:T;
n = T/dt;      % Number of timesteps

phi1 = zeros(ceil(n),1);      % Initializing Solution to 0
phi1(1) = 1;           % Assigning Initial Condition at x=0

phis = 0;
k1 = 0;

for i=1:n-1
    phis = phi1(i);              % Step 1
    k1 = -phi1(i);              % Step 1
    phis = phis + (dt/3)*k1;              % Step 2
    k1 = (-5/9)*k1 - phis;           % Step 2
    phis = phis + (15/16)*dt*k1;              % Step 3
    k1 = (-153/128)*k1 - phis;             % Step 3
    phi1(i+1) = phis + (8/15)*dt*k1;             % Solution Step
end

t = 0:0.1:T;
plot(t, exp(-t), 'r','LineWidth', 2);
hold on;
plot(time(1:length(phi1)), phi1, 'bo-', 'MarkerFaceColor', 'b');
xlabel('x');
xlim([0 10]);
ylabel('\phi');
legend('Exact Solution', 'RK-3 for dt=0.1')
grid on;
grid minor;
```

The above code is shown for $\Delta t = 0.1$. This can be changed to arrive at the plots for different timesteps, as shown in the following plots: (**Note: the x-axis in the following plots should actually be 't' instead of 'x'**)

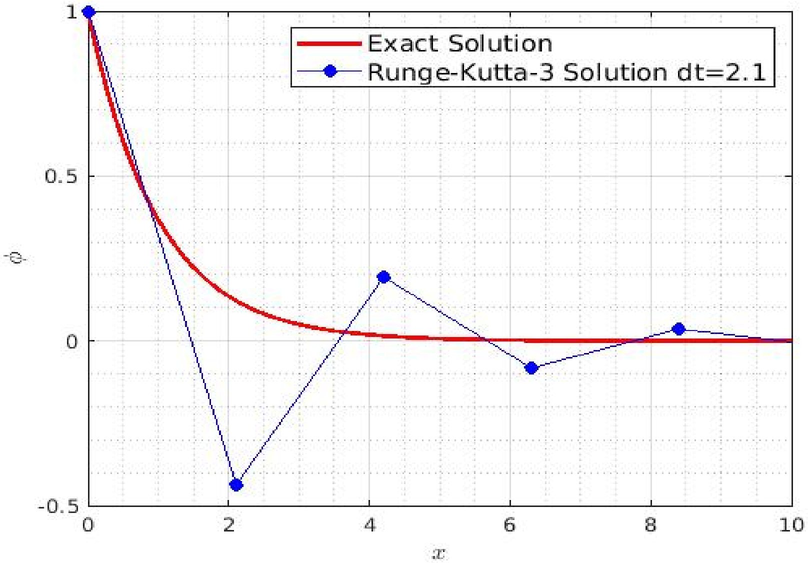

Figure 11: RK-3 for $\Delta t = 2.1$

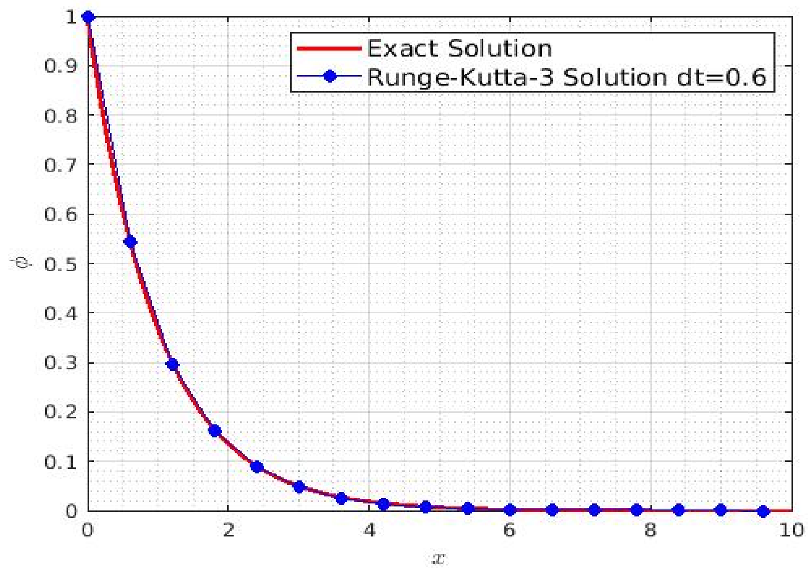

Figure 12: RK-3 for $\Delta t = 0.6$

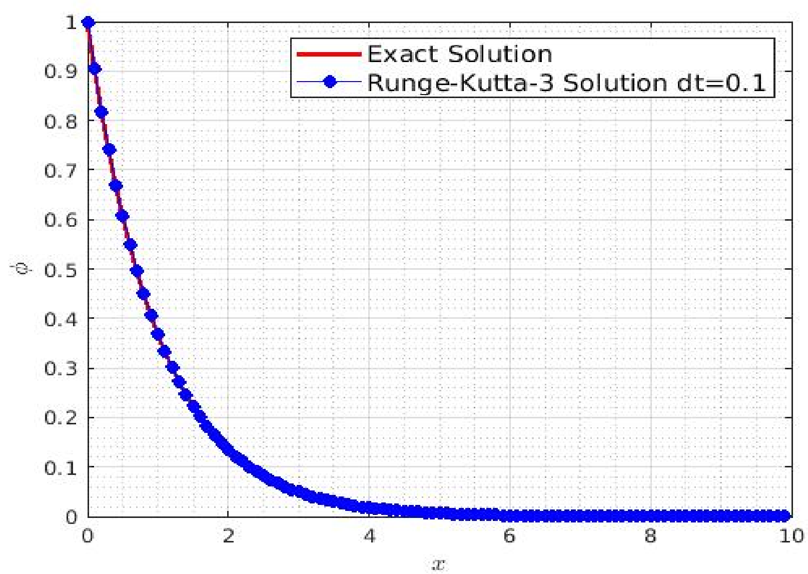

Figure 13: RK-3 for $\Delta t = 0.1$

OBSERVATIONS

The Runge-Kutta scheme is stable, but for $\Delta t = 2.1$, the solution is not accurate at all. With decrease in the timestep, the solution becomes increasingly accurate.

Problem 2b

$$\boxed{\frac{d^2\phi}{dt2} = -\phi} \quad ; \quad \phi(0) = 1 \quad ; \quad \phi'(0) = 0$$

Exact Solution: $\phi = cos(t)$

EXPLICIT EULER SOLUTION

From Taylor Series Expansion,

$$\phi_{n+1} = \phi_n + \Delta t \phi'_n$$

Differentiating this again,

$$\phi'_{n+1} = \phi'_n + \Delta t \phi''_n$$
$$\implies \phi'_{n+1} = \phi'_n - \Delta t \phi_n$$

Therefore, we can use the following two equations to march in time. We know $\phi(0)$ and $\phi'(0)$ to start off:

$$\boxed{\phi_{n+1} = \phi_n + \Delta t \phi'_n}$$
$$\boxed{\phi'_{n+1} = \phi'_n - \Delta t \phi_n}$$

```
%% Explicit Euler for 2b
T = 15;

dt = 0.1 ;      % For timestep = 0.1
time = 0:dt:T;
n = T/dt;      % Number of timesteps

phi1 = zeros(ceil(n),1);       % Initializing Solution to 1
phi1_prime = zeros(ceil(n),1);
phi1(1) = 1;            % Assigning Initial Condition
phi1_prime(1) = 0;

for i = 1:n-1
    phi1(i+1) = phi1(i) + dt*phi1_prime(i);
    phi1_prime(i+1) = phi1_prime(i) - dt*phi1(i);
end

t = 0:0.1:T;
plot(t, cos(t), 'r','LineWidth', 2);
hold on;
plot(time(1:length(phi1)), phi1, 'bo-', 'MarkerFaceColor', 'b');
xlabel('x');
xlim([0 10]);
ylabel('\phi');
legend('Exact Solution', 'Explicit Solution for dt=0.1')
grid on;
grid minor;
```

The above code is shown for $\Delta t = 0.1$. This can be changed to arrive at the plots for different timesteps, as shown in the following plots: (**Note: the x-axis in the following plots should actually be 't' instead of 'x'**)

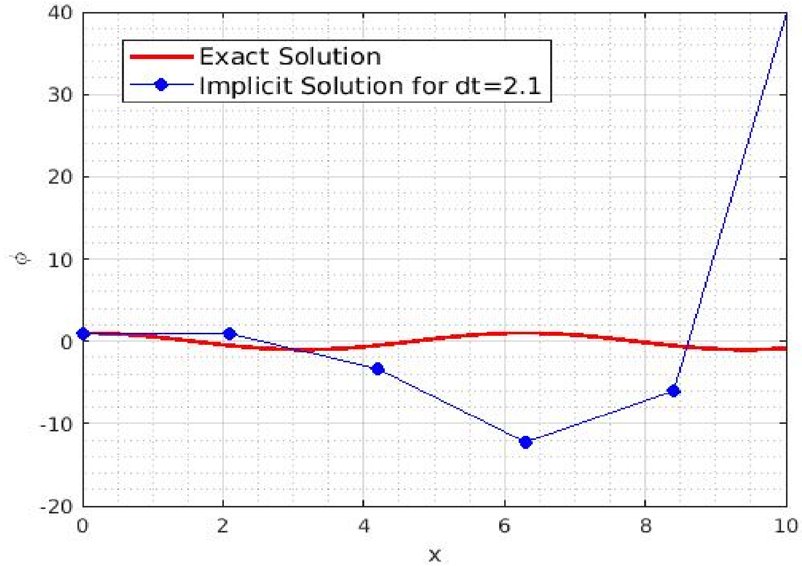

Figure 14: Explicit Euler for $\Delta t = 2.1$

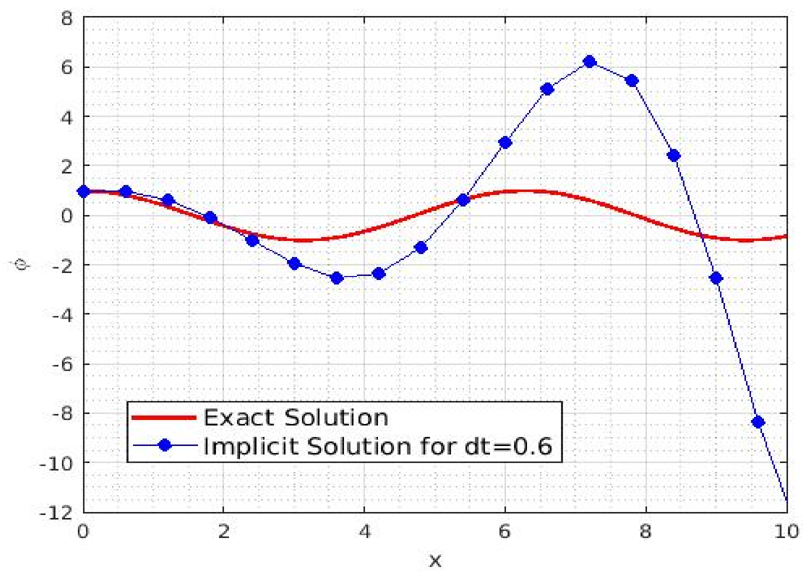

Figure 15: Explicit Euler for $\Delta t = 0.6$

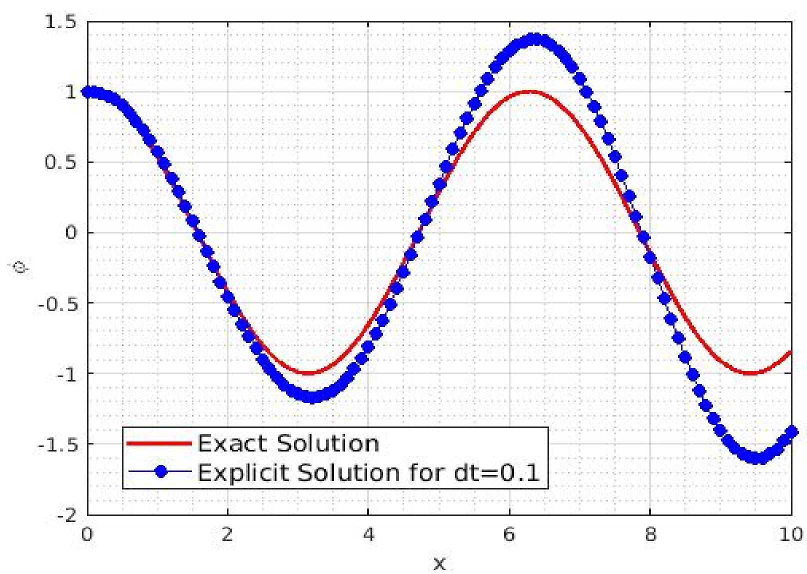

Figure 16: Explicit Euler for $\Delta t = 0.1$

OBSERVATION

It is observed that for all timesteps chosen, the amplitude of the oscillations is increasing with time. Therefore this explicit scheme is unstable for all timesteps. However, when the timestep is smaller, the growth of amplitude is slower.

IMPLICIT EULER SOLUTION

From Taylor Series Expansion,

$$\phi_{n+1} = \phi_n + \Delta t \phi'_{n+1}$$

Differentiating this again,

$$\phi'_{n+1} = \phi'_n + \Delta t \phi''_{n+1}$$
$$\implies \phi'_{n+1} = \phi'_n - \Delta t \phi_{n+1}$$

After manipulating the above equations in ϕ_{n+1} and ϕ'_{n+1}, we get the following two explicit equations:

$$\boxed{\phi_{n+1} = \frac{\phi_n + \Delta t \phi'_n}{1 + \Delta t^2}}$$

$$\boxed{\phi'_{n+1} = \frac{\phi'_n - \Delta t \phi_n}{1 + \Delta t^2}}$$

```
%% Implicit Euler for 2b
T = 15;

dt = 0.1 ;      % For timestep = 0.1
time = 0:dt:T;
n = T/dt;      % Number of timesteps

phi1 = zeros(ceil(n),1);      % Initializing Solution to 1
phi1_prime = zeros(ceil(n),1);
phi1(1) = 1;            % Assigning Initial Condition
phi1_prime(1) = 0;

for i = 1:n-1
    phi1(i+1) = (phi1(i) + dt*phi1_prime(i))/(1 + dt*dt);
    phi1_prime(i+1) = (phi1_prime(i) - dt*phi1(i))/(1 + dt*dt);
end


t = 0:0.1:T;
plot(t, cos(t), 'r','LineWidth', 2);
hold on;
plot(time(1:length(phi1)), phi1, 'bo-', 'MarkerFaceColor', 'b');
xlabel('x');
xlim([0 10]);
ylabel('\phi');
legend('Exact Solution', 'Implicit Solution for dt=0.1')
grid on;
grid minor;
```

The above code is shown for $\Delta t = 0.1$. This can be changed to arrive at the plots for different timesteps, as shown in the following plots: (**Note: the x-axis in the following plots should actually be 't' instead of 'x'**)

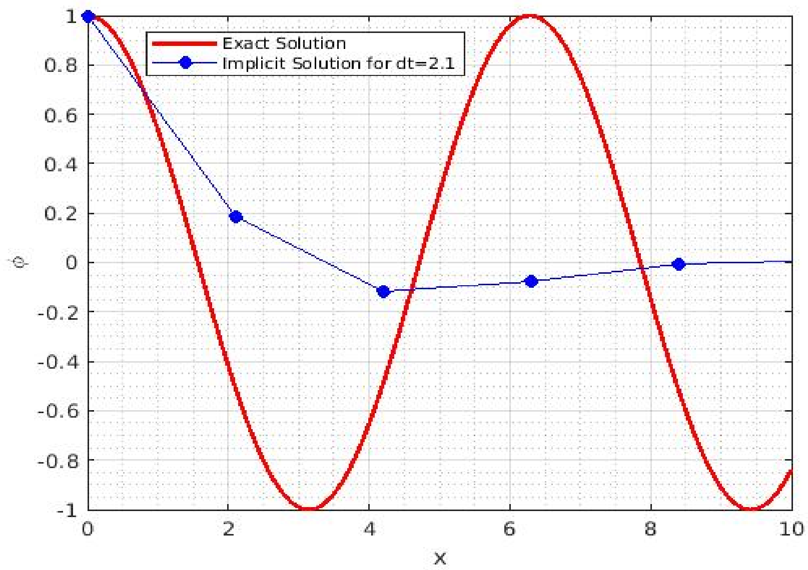

Figure 17: Implicit Euler for $\Delta t = 2.1$

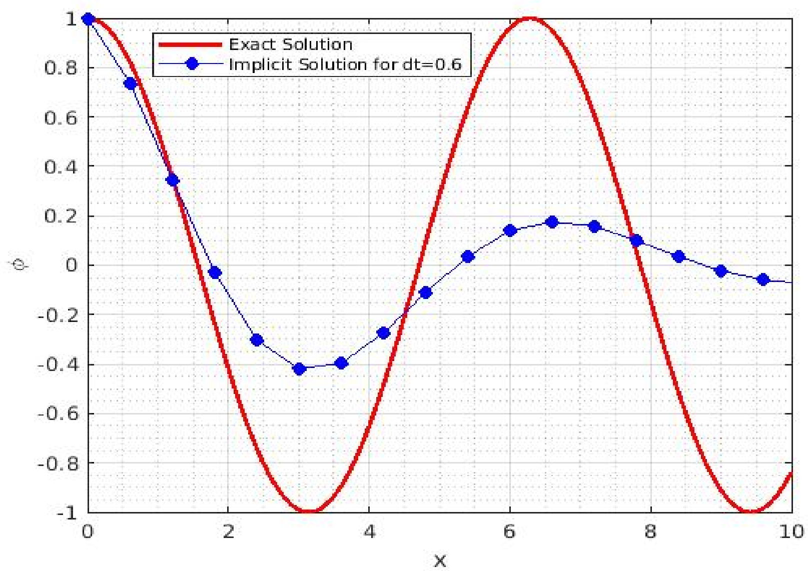

Figure 18: Implicit Euler for $\Delta t = 0.6$

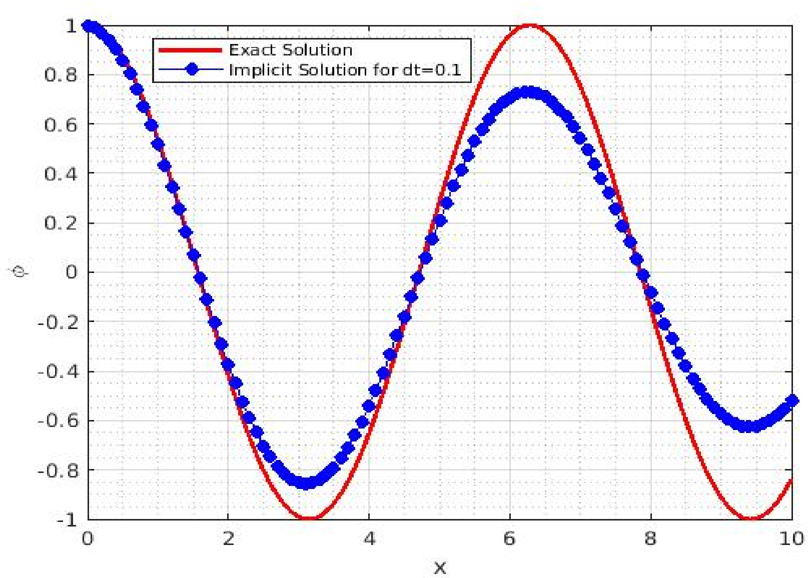

Figure 19: Implicit Euler for $\Delta t = 0.1$

OBSERVATION

For the implicit scheme applied, the amplitude in the case of all timesteps is decreasing with time. Thereby, this scheme is imparting artificial dissipation to the solution. With a smaller timestep, the decrease in the amplitude is slower.

CRANK-NICOLSON SOLUTION

Applying Taylor Series Expansion and arriving at the Crank Nicolson Scheme,

$$\phi_{n+1} = \phi_n + \frac{\Delta t}{2}(\phi_n' + \phi_{n+1}') + O(\Delta t^3)$$

Differentiating this again,

$$\phi_{n+1}' = \phi_n' + \frac{\Delta t}{2}(\phi_n'' + \phi_{n+1}'')$$

$$\phi_{n+1}' = \phi_n' + \frac{\Delta t}{2}(-\phi_n - \phi_{n+1})$$

After manipulating the above equations in ϕ_{n+1} and ϕ_{n+1}', we get the following two explicit equations:

$$\boxed{\phi_{n+1} = \frac{1}{1 + \dfrac{\Delta t^2}{4}} \left\{ \left(1 - \frac{\Delta t^2}{4}\right) \phi_n + \Delta t \phi_n' \right\}}$$

$$\boxed{\phi_{n+1}' = \frac{1}{1 + \dfrac{\Delta t^2}{4}} \left\{ \left(1 - \frac{\Delta t^2}{4}\right) \phi_n' - \Delta t \phi_n \right\}}$$

```
%% Crank Nicolson for 2b
T = 15;

dt = 0.1 ;      % For timestep = 0.1
time = 0:dt:T;
n = T/dt;     % Number of timesteps

phi1 = zeros(ceil(n),1);      % Initializing Solution to 1
phi1_prime = zeros(ceil(n),1);
phi1(1) = 1;             % Assigning Initial Condition
phi1_prime(1) = 0;

for i = 1:n-1
    phi1(i+1) = ((1-(dt/2)*(dt/2))*phi1(i) + dt*phi1_prime(i))/(1 + (dt/2)*(dt/2));
    phi1_prime(i+1) = ((1-(dt/2)*(dt/2))*phi1_prime(i) - dt*phi1(i))/(1 + (dt/2)*(dt/2));
end

t = 0:0.1:T;
plot(t, cos(t), 'r','LineWidth', 2);
hold on;
plot(time(1:length(phi1)), phi1, 'bo-', 'MarkerFaceColor', 'b');
xlabel('x');
xlim([0 10]);
ylabel('\phi');
legend('Exact Solution', 'Crank Nicolson for dt=0.1')
grid on;
grid minor;
```

The above code is shown for $\Delta t = 0.1$. This can be changed to arrive at the plots for different timesteps, as shown in the following plots: (**Note: the x-axis in the following plots should actually be 't' instead of 'x'**)

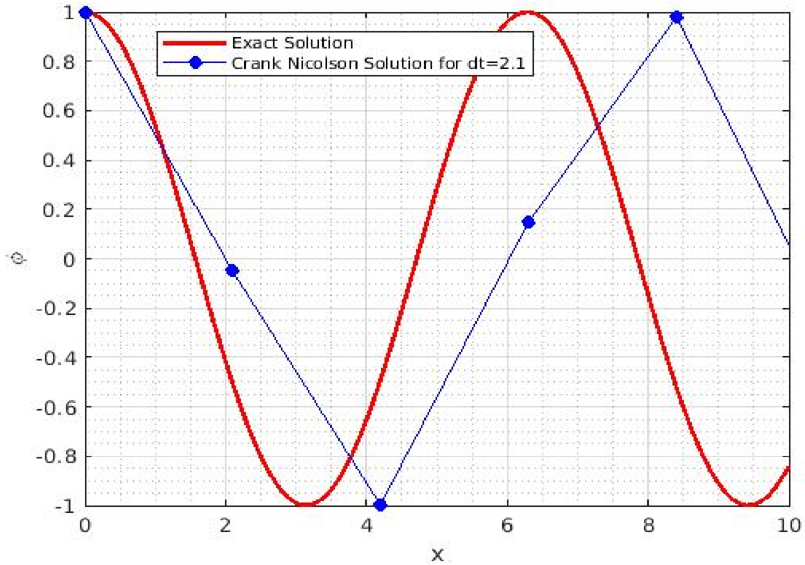

Figure 20: Crank Nicolson for $\Delta t = 2.1$

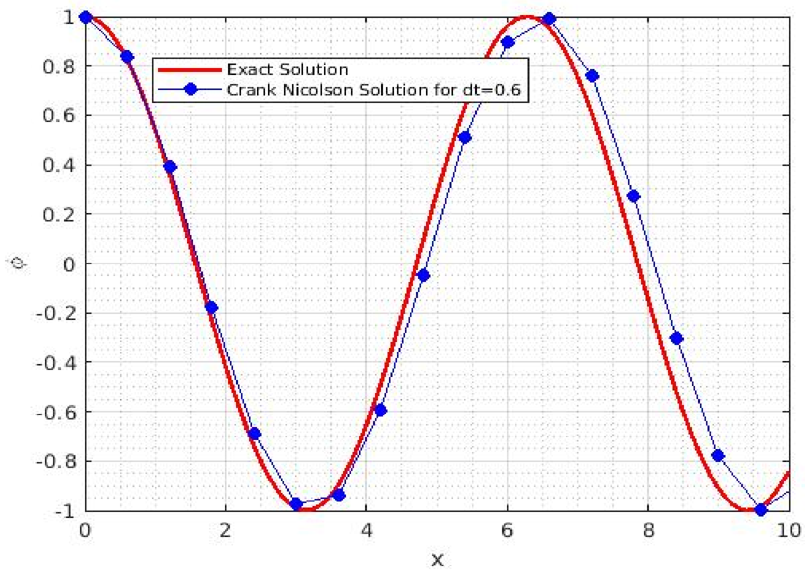

Figure 21: Crank Nicolson for $\Delta t = 0.6$

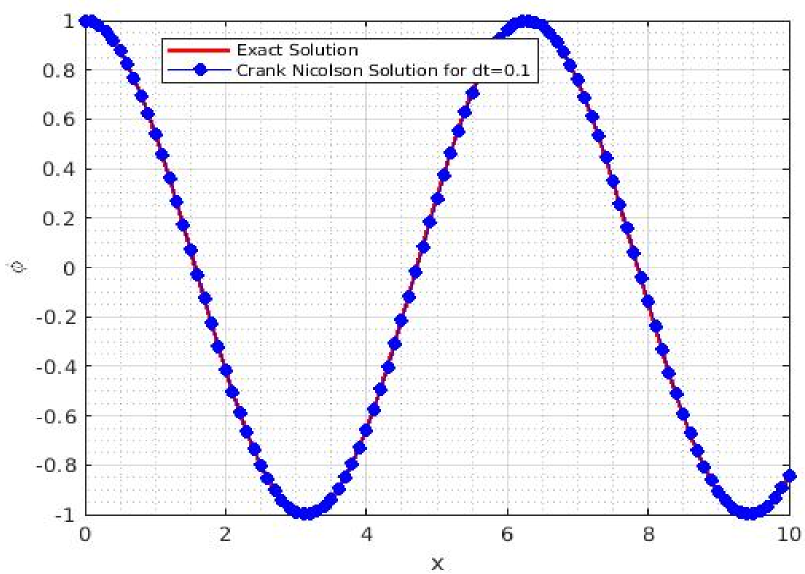

Figure 22: Crank Nicolson for $\Delta t = 0.1$

OBSERVATION

The Crank-Nicolson scheme is stable for all the timesteps chosen. However, it is observed that the scheme imparts a shift to the solution in a manner that the solution is convected. For a smaller timestep, this shift is small, and the numerical solution resembles the exact solution closely.