1.2

Programming Task

A program to implement the three methods can be found on page 10.

Question 1

From the table, we can see that $\dfrac{\log E_n - \log E_{n-1}}{h}$ tends to a number which is approximately 3.12245832. If we assume it to be γ, then $\frac{\log E_n - \log E_{n-1}}{h} \approx \gamma$ so $\frac{E_n}{E_{n-1}} \approx e^{\gamma h}$. Multiplying from 2 to n, we get $E_n \propto e^{\gamma h n} = e^{\gamma x}$. So the error grows exponentially with 'growth rate' $\gamma = 3.12245832$.

We run the code for $h = 0.2, 0.1$ and 0.005 [i.e for $n = 50, 100$ 200]. The table below shows the results at the point $x = 5$ and 10.

We can see that in both cases of $x = 5$ and 10, the error increases as we take $h = 0.2$ and 0.1 but decreases when $h = 0.05$. The growth rate keeps increasing and gets colser to 4.

x_n	Y_n	$y(x_n)$	E_n	$\frac{\log E_n - \log E_{n-1}}{h}$
0.40000000	1.20000000	0.46842353	0.73157647	
0.80000000	-2.23123189	0.40856676	-2.63979865	3.20814042
1.20000000	9.41833156	0.29296446	9.12536710	3.10088871
1.60000000	-31.64702677	0.20023496	-31.84726173	3.12473319
2.00000000	111.17336888	0.13499982	111.03836906	3.12231101
2.40000000	-387.07700251	0.09065022	-387.16765274	3.12245500
2.80000000	1350.03750000	0.06079639	1349.97670361	3.12246201
3.20000000	-4707.05105837	0.04075944	-4707.09181781	3.12245733
3.60000000	16412.69871608	0.02732317	16412.67139291	3.12245853
4.00000000	-57227.62137289	0.01831553	-57227.63968842	3.12245828
4.40000000	199541.13106686	0.01227732	199541.11878954	3.12245833
4.80000000	-695759.21132122	0.00822974	-695759.21955096	3.12245832
5.20000000	2425970.62704616	0.00551656	2425970.62152959	3.12245832
5.60000000	-8458865.20462916	0.00369786	-8458865.20832703	3.12245832
6.00000000	29494339.29073435	0.00247875	29494339.28825560	3.12245832
6.40000000	-102840750.92903011	0.00166156	-102840750.93069166	3.12245832
6.80000000	358584742.26761848	0.00111378	358584742.26650470	3.12245832
7.20000000	-1250311926.18273616	0.00074659	-1250311926.18348265	3.12245832
7.60000000	4359582906.05416679	0.00050045	4359582906.05366611	3.12245832
8.00000000	-15200977225.55486679	0.00033546	-15200977225.55520248	3.12245832
8.40000000	53002710027.83055115	0.00022487	53002710027.83032990	3.12245832
8.80000000	-184809649314.61212158	0.00015073	-184809649314.61227417	3.12245832
9.20000000	644393587834.58972168	0.00010104	644393587834.58959961	3.12245832
9.60000000	-2246869130385.29882813	0.00006773	-2246869130385.29882813	3.12245832
10.00000000	7834374805067.54687500	0.00004540	7834374805067.54687500	3.12245832

Table 1: LF method with $h = 0.4$ from $x = 0$ to $x = 10$

x_n	Y_n	$y(x_n)$	E_n	$\frac{\log E_n - \log E_{n-1}}{h}$
$x = 5, h = 0.2$	9188936.9479	0.00673794	9188936.9412	3.66334128
$x = 5, h = 0.1$	-9862748.7013	0.00673794	-9862748.7081	3.90035320
$x = 5, h = 0.05$	-3872083.3445	0.00673794	-3872083.3512	3.97380221
$x = 10, h = 0.2$	-828151490544095.7500	0.00004540	-828151490544095.7500	3.66334128
$x = 10, h = 0.1$	-2907416990859673.0000	0.00004540	-2907416990859673.0000	3.90035320
$x = 10, h = 0.05$	-1647959562443503.2500	0.00004540	-1647959562443503.2500	3.97380221

Table 2: LF method with $h = 0.2, 0.1, 0.005$ for $x = 5, 10$

Question 2

(i) We have $Y_{n+1} = Y_{n-1} + 2h(-4Y_n + 3e^{-hn})$. It has particular solution

$$p(n) = \frac{6he^{-hn}}{8h + e^{-h} - e^h} = p_h e^{-hn} \text{ where } p_h = \frac{6h}{8h + e^{-h} - e^h} = \frac{3h}{4h - \sinh h}$$

Let z_n be the complimentary solution. So $z_{n+1} + 8hz_n - z_{n-1} = 0$, which has solution in the form $z_n = A\lambda^n + B\mu^n$ where A, B are constants and λ, μ are roots of the equation $z^2 + 8hz - 1 = 0$. So $(\lambda, \mu) = (-4h + \sqrt{16h^2+1}, -4h - \sqrt{16h^2+1})$. Now

$$\begin{aligned} Y_0 =& 0 = z_0 + p = A + B + p(o) \\ Y_1 =& 3h = z_1 + p = A\lambda + B\mu + p(1) \end{aligned}$$

Solving them, we get $(A, B) = \left(\frac{3h + p_h(\mu - e^{-h})}{\lambda - \mu}, -\frac{3h + p(\lambda - e^{-h})}{\lambda - \mu}\right)$. So $Y_n = A\lambda^n + B\mu^n + p(n)$ where A, B, λ, μ, p are specified above.

(ii) We have $(\lambda, \mu) = (-4h + \sqrt{16h^2+1}, -4h - \sqrt{16h^2+1})$, so $|\lambda| < 1, |\mu| > 1$. Hence $|\lambda^n| \to 0$ and $|\mu^n| \to \infty$ as $n \to \infty$. Also $p(n) = p_h e^{-hn} \to 0$ as $n \to \infty$, which implies $|Y_n| = |A\lambda^n + B\mu^n + p(n)| \to \infty$ as $n \to \infty$. But $y(x) = e^{-x} - e^{-4x} \to \infty$ as $x \to \infty$. That's why instability occurs.

Now

$$\begin{aligned} \lim_{n\to\infty} |\frac{E_{n+1}}{E_N}| &= \lim_{n\to\infty} \left| \frac{A\lambda^{n+1} + B\mu^{n+1} + p - e^{-(n+1)h} + e^{-4(n+1)h}}{A\lambda^n + B\mu^n + p - e^{-nh} + e^{-4nh}} \right| \\ &= \lim_{n\to\infty} \left| \frac{B\mu^{n+1} + p}{B\mu^n + p} \right| \\ &= |\mu| \\ &= 4h + \sqrt{16h^2+1} \end{aligned}$$

We also know $\lim_{n\to\infty} |\frac{E_{n+1}}{E_N}| = e^{\gamma h}$, hence $\gamma = \frac{1}{h}\log(4h + \sqrt{16h^2+1})$.

(iii) Suppose $x = nh$ is fixed, so $h = x/n$. The constants A, B, λ, μ depends only on h, so instead we will write $A_h, B_h, \lambda_h, \mu_h$.

Now

$$\lim_{h\to 0} p_h = \lim_{h\to 0} \frac{3h}{4h - \sinh h} = \lim_{h\to 0} \frac{3}{4 - \cosh h} = 1$$

Also

$$A_h = \frac{3h + p_h(\mu_h - e^{-h})}{\lambda_h - \mu_h} \to \frac{0 + 1(-1-1)}{1 - (-1)} = -1 \text{ as } h \to 0$$

As $A_h + B_h + p_h = 0$, we have $B_h \to 0$ as $h \to 0$.

Also

$$\frac{1}{1+4h} > \frac{1}{4h+\sqrt{16h^2+1}} = \lambda_h = -4h + \sqrt{16h^2+1} > 1-4h$$
$$\Rightarrow \frac{1}{\left(1+\frac{4x}{n}\right)^n} > \lambda_h^n > \left(1-\frac{4x}{n}\right)^n$$

Similarly for $|\mu_h| = 4h + \sqrt{16h^2+1}$, we can show that

$$\frac{1}{1-4h} > |\mu_h|^n > 1+4h$$
$$\frac{1}{\left(1-\frac{4x}{n}\right)^n} > |\mu_h|^n > \left(1+\frac{4x}{n}\right)^n$$

Now using the fact that $\left(1+\frac{4x}{n}\right)^n \to e^{4x}$ and $\left(1-\frac{4x}{n}\right)^n \to e^{-4x}$ as $n \to \infty$ and applying Sandwitch theorem, we can conclude that $\lambda_h^n \to e^{-4x}$ and $|\mu_h^n| \to e^{4x}$ as $n \to \infty$.

Hence $Y_n = A_h\lambda_h^n + B_h\mu_h^n + p_h e^{-hn} \to (-1)\times e^{-4x} + 0\times e^{4x} + 1\times e^{-x} = e^{-x} - e^{-4x} = y(x)$. So the solution of LF-difference-equation converges to the analytic solution.

Question 3

Numerical solution of ODE (1), (10) using Euler and RK4 method can be found in Table 3. Figure 1 shows the numerical solutions with the exact solution superimposed.

x_n	$y(x_n)$	Y_n(Euler method)	Y_n(RK4 method)
0.40	0.46842353	1.20000000	0.40585600
0.80	0.40856676	0.08438406	0.38179688
1.20	0.29296446	0.48856432	0.28560073
1.60	0.20023496	0.06829446	0.19946792
2.00	0.13499982	0.20129915	0.13587704
2.40	0.09065022	0.04162285	0.09166779
2.80	0.06079639	0.08388783	0.06160540
3.20	0.04075944	0.02263938	0.04133823
3.60	0.02732317	0.03533102	0.02772144
4.00	0.01831553	0.01158986	0.01858537

Table 3: Euler and RK4 method with $h = 0.4$ from $x = 0$ to $x = 4$

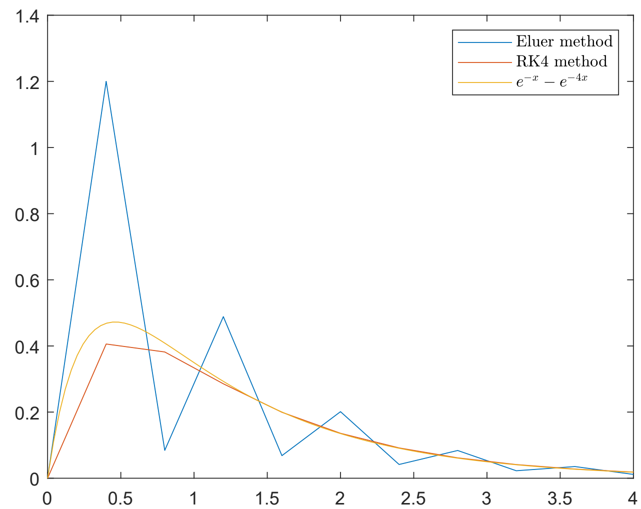

Figure 1: Euler and RK4 method with $h = 0.4$ from $x = 0$ to $x = 4$

Question 4

Table 4 demostrates the global error E_n of the Euler, LF, RK4 method at $x_n = 0.4$ against $h = \frac{0.4}{n}$ for $n = 2^k$, $k = 0, 1, 2, \ldots, 15$. Figure 2 shows a log-log graph of $|E_n|$ against h ove this range.

	E_n		
k	Euler	LF	RK4
0	0.73157647	0.73157647	0.0625675255534157
1	0.14281492	0.44594662	0.0019232773566684
2	0.06371591	0.16016334	0.0000842011524771
3	0.03003888	0.04479847	0.0000044133616700
4	0.01459857	0.01153981	0.0000002526839234
5	0.00719827	0.00290699	0.0000000151165496
6	0.00357439	0.00072814	0.0000000009243527
7	0.00178107	0.00018212	0.0000000000571440
8	0.00088901	0.00004554	0.0000000000035523
9	0.00044413	0.00001138	0.0000000000002225
10	0.00022197	0.00000285	0.0000000000000128
11	0.00011096	0.00000071	0.0000000000000042
12	0.00005547	0.00000018	0.0000000000000041
13	0.00002774	0.00000004	0.0000000000000202
14	0.00001387	0.00000001	0.0000000000000182
15	0.00000693	0.00000000	0.0000000000000746

Table 4: Global error of Euler, LF and RK4 method at $x_n = 0.4$

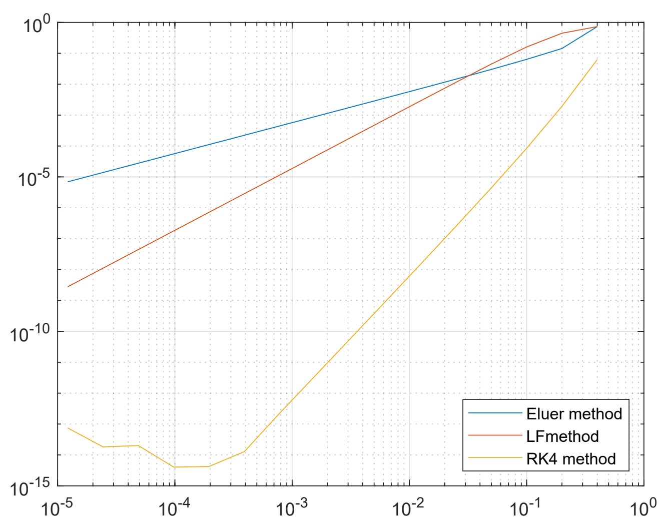

Figure 2: log-log graph of $|E_n|$ against h

Question 5

Let y_c and y_p be the complimentary and particular solutions. Now y_c has characteristic equation $x^2 + \gamma x + \Omega^2 x = 0$, which has complex roots because $0 < \gamma < 2\Omega$. So $y_c = e^{\rho t}(P\cos\sigma t + Q\sin\sigma t)$ where P, Q are constants and $(\rho, \sigma) = (-\frac{\gamma}{2}, \sqrt{\Omega^2 - (\frac{\gamma}{2})^2})$. Now $(P\cos\sigma t + Q\sin\sigma t)$ is bounded and $e^{\rho t} \to 0$ as $\rho < 0$. Hence $y_c \to 0$ as $t \to \infty$.

The forced term of the ODE is $a\sin(\omega t)$, so the particular solution is in the form

$$y_p = U\sin(\omega t) + V\cos(\omega t)$$

Substituting it in the original ODE, we get

$$(U, V) = (\frac{a(\Omega^2 - \omega^2)}{(\Omega^2 - \omega^2)^2 + \gamma^2\omega^2}, \frac{-a\gamma\omega}{(\Omega^2 - \omega^2)^2 + \gamma^2\omega^2})$$

Hence the analytics olution is $y(t) = y_c + y_p = e^{\rho t}(P\cos\sigma t + Q\sin\sigma t) + U\sin(\omega t) + V\cos(\omega t)$ where P, Q, U, V are specified above.

Now $y = y_c + y_p \to y_p = U\sin(\omega t) + V\cos(\omega t) = A_s\sin(\omega t - \phi_s)$ as $t \to \infty$ where

$$(A_s, \phi_s) = (U^2 + V^2, \arctan\frac{-V}{U}) = (\frac{a}{\sqrt{(\gamma\omega)^2 + (\Omega^2 - \omega^2)^2}}, \arctan(\frac{\gamma\omega}{\Omega^2 - \omega^2}))$$

Programming task

A program to solve equation (15) with initial conditions (20) using the RK4 method can be found on page 12.

Question 6

For equation (21), we have $\Omega = 1, a = 1$. Solving for $y = \frac{dy}{dt} = 0$, we get $(P, Q) = (-V, \frac{V\rho - U\omega}{\sigma})$. Hence the analytic solution is

$$\begin{aligned} y(t) &= e^{\rho t}(P\cos\sigma t + Q\sin\sigma t) + U\sin\omega t + V\cos\omega t \\ &= \frac{1}{(1-\omega^2)^2 + \gamma^2\omega^2}\left(\gamma\omega\cos\sigma t - \frac{\omega(\gamma\rho + 1 - \omega^2)}{\sigma}\sin\sigma t + (1-\omega^2)\sin\omega t - \gamma\omega\cos\omega t\right) \end{aligned}$$

where ρ, σ are specified above. A program to find the analytic solution in terms of γ and ω can be found in page 14.

Numerical solution of ODE (15) with $\gamma = 1, \omega = \sqrt{3}$ and initial conditions (20) for t up to 10 with $h = 0.4$ can be found in table 5. Table 6 shows some results of the same ODE for $h = 0.2$ and $h = 0.1$. Program to generate the tables are in page 14

t	Y_n	$y(t_n)$	$E_n = Y_n - y(t_n)$
0.00	0.00000000	0.00000000	0.00000000
0.40	0.01629712	0.01622784	0.00006929
0.80	0.10709631	0.10705223	0.00004409
1.20	0.27735052	0.27735670	−0.00000619
1.60	0.46317589	0.46319336	−0.00001747
2.00	0.57001842	0.56998226	0.00003616
2.40	0.52514946	0.52501709	0.00013237
2.80	0.31906494	0.31884920	0.00021575
3.20	0.01601561	0.01578634	0.00022927
3.60	−0.27142181	−0.27156798	0.00014617
4.00	−0.43191070	−0.43189732	−0.00001339
4.40	−0.40589676	−0.40570782	−0.00018894
4.80	−0.21341441	−0.21310784	−0.00030658
5.20	0.05408533	0.05439858	−0.00031325
5.60	0.27449342	0.27469586	−0.00020244
6.00	0.34989073	0.34990951	−0.00001878
6.40	0.25039991	0.25023906	0.00016086
6.80	0.02693791	0.02667660	0.00026131
7.20	−0.21296849	−0.21321127	0.00024278
7.60	−0.35516526	−0.35528375	0.00011850
8.00	−0.33165293	−0.33160140	−0.00005153
8.40	−0.15189534	−0.15170730	−0.00018804
8.80	0.10171222	0.10194097	−0.00022875
9.20	0.31206394	0.31222048	−0.00015654
9.60	0.38157993	0.38158683	−0.00000689
10.00	0.27741533	0.27726642	0.00014891

Table 5: RK4 method with $\gamma = 1, \omega = \sqrt{3}$ for $h = 0.4$

t	$Y_n[h = 0.2]$	$Y_n[h = 0.1]$	$y(t_n)$	$E_n[h = 0.2]$	$E_n[h = 0.1]$
2.00	0.56998817	0.56998274	0.56998226	0.00000591	0.00000048
4.00	−0.43190295	−0.43189781	−0.43189732	−0.00000564	−0.00000049
6.00	0.34991316	0.34990987	0.34990951	0.00000365	0.00000036
8.00	−0.33160834	−0.33160193	−0.33160140	−0.00000694	−0.00000053
10.00	0.27727812	0.27726720	0.27726642	0.00001170	0.00000078

Table 6: RK4 method with $\gamma = 1, \omega = \sqrt{3}$ for $h = 0.2, 0.1$ at $x = 2, 4, 6, 8, 10$

Question 7

Figure 3-6 show numerical solution of (20)-(21) up to $t = 40$ for $h = 1$ and $(\omega, \gamma) = (1, 0.25)$, $(1, 1.0)$, $(2, 0.5)$, $(2, 1.9)$ respectively. We see that the numerical solutions are very close to the analytic solutions.

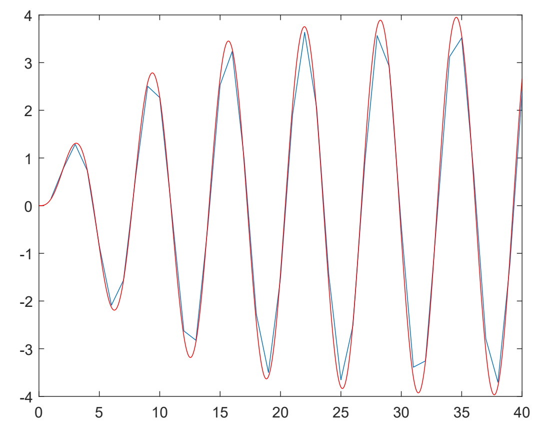

Figure 3: RK4 program with $\omega = 1, \gamma = 0.25$

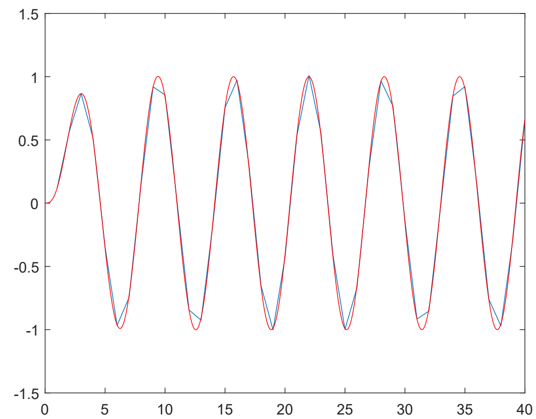

Figure 4: RK4 program with $\omega = 1, \gamma = 1.0$

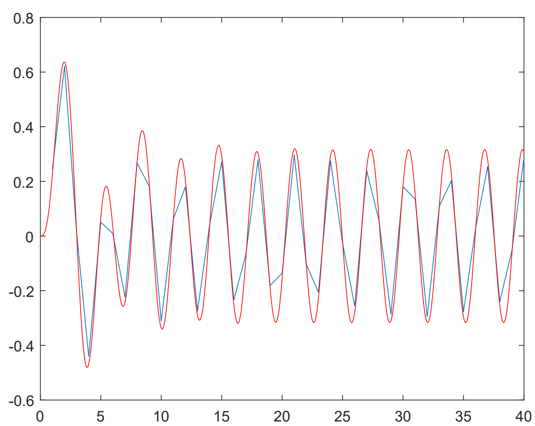

Figure 5: RK4 program with $\omega = 2, \gamma = 0.25$

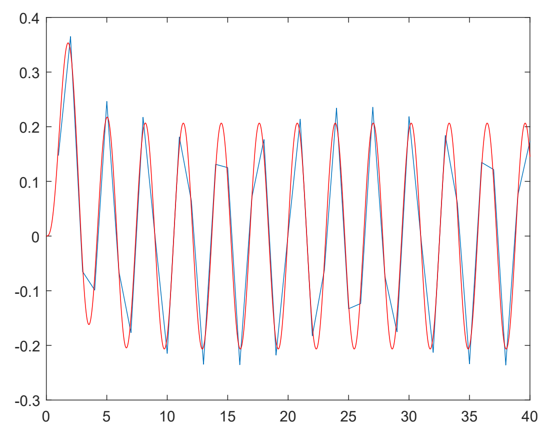

Figure 6: RK4 program with $\omega = 2, \gamma = 1.9$

Program for Euler, LF and RK4 method

(i) EulerMethod.m

```
function [E,Yarray] = EulerMethod (xmax,nmax)

f = @(x,y) -4*y+3*exp(-x);
y = @(x) exp(-x)-exp(-4*x);
h = xmax/nmax;
x = 0;
Yexact = 0;
Y = 0;
Yarray(1) = 0;

for i = 1:nmax
    Yexact = y(x+h);
    Y = Y + h*f(x,Y);
    E = Y - Yexact;
    x = x + h;

    fprintf('%.8f %.8f %.8f %.8f \n' ,x, Y, Yexact, E );

    Yarray(i+1) = Y;

end

end
```

(ii) LFmethod.m

```
function [E] = LFmethod (xmax,nmax)

f = @(x,y) -4*y+3*exp(-x);
y = @(x) exp(-x)-exp(-4*x);
h = xmax/nmax;
Yp = 0;
Ep = 0;
Y = Yp + h*f(0,Yp);
Yexact = y(h);
E = Y - Yexact;
x = h;

fprintf('%.8f %.8f %.8f %.8f %.8f \n' ,x, Y, Yexact, E, (log(abs(E))-log(abs(Ep)))/h );


for i=1:nmax-1
```

```
    Yexact = y(x+h);
    storeme = Y;
    Y = Yp + 2*h*f(x,Y);
    Yp = storeme;
    Ep = E;
    E = Y - Yexact;
    x = x + h;

    fprintf('%.8f %.8f %.8f %.8f %.8f \n' ,x, Y, Yexact, E, (log(abs(E))-log(abs(Ep)))/h );

end

end
```

(iii) RK4method.m

```
function [E,Yarray] = RK4method (xmax,nmax)

f = @(x,y) -4*y+3*exp(-x);
y = @(x) exp(-x)-exp(-4*x);
h = xmax/nmax;
x = 0;
Yexact = 0;
Y = 0;
Yarray(1) = 0;
for i = 1:nmax
    Yexact = y(x+h);
    k1 = h*f(x,Y);
    k2 = h*f(x+h/2,Y+(k1)/2);
    k3 = h*f(x+h/2,Y+(k2)/2);
    k4 = h*f(x+h,Y+k3);

    Y = Y + (k1+2*k2+2*k3+k4)/6;
    E = Y - Yexact;
    x = x + h;

    fprintf('%.8f %.8f %.8f %.8f \n' ,x, Y, Yexact, E );

    Yarray(i+1) = Y;
end

end
```

Program to produce figure 1

```
q3plot.m

figure
x = linspace(0,4,11);

[E1,Yarray1] = EulerMethod (4,10);
[E2,Yarray2] = RK4method(4,10);

plot(x,Yarray1,x,Yarray2)

hold on
x = linspace(0,4,100);
y3 = exp(-x)-exp(-4*x);
plot(x,y3)
hold off

legend({'Eluer method','RK4 method','$e^{-x}-e^{-4x}$'},'Interpreter','latex')
```

Program to produce figure 2

```
q4plot.m

for k=0:15
    E1(k+1) = abs(EulerMethod(0.4,2^k));
    E2(k+1) = abs(LFmethod(0.4,2^k));
    E3(k+1) = abs(RK4method(0.4,2^k));
    h(k+1)  = (0.4)/(2^k);

end

figure
loglog(h,E1,h,E2,h,E3)
grid on
legend('Eluer method','LFmethod','RK4 method','Location','southeast')
```

Program for solving 2nd order ODE using RK4

```
RK4method2.m

function [Y] = RK4method2(gamma,delta,Omega,omega,a,tmax,nmax)


f1 = @(t,y1,y2) (y2);
f2 = @(t,y1,y2) -(gamma)*(y2) - (delta)^3*(y1)^2*(y2) - (Omega)^2*(y1) + a*sin((omega)*t);

h = tmax/nmax;
t = 0;
Y1 = 0;
Y2 = 0;

for i=1:nmax


    k1 = h*f1(t,Y1,Y2);
    m1 = h*f2(t,Y1,Y2);

    k2 = h*f1(t+h/2,Y1+(k1)/2,Y2+(m1)/2);
    m2 = h*f2(t+h/2,Y1+(k1)/2,Y2+(m1)/2);

    k3 = h*f1(t+h/2,Y1+(k2)/2,Y2+(m2)/2);
    m3 = h*f2(t+h/2,Y1+(k2)/2,Y2+(m2)/2);

    k4 = h*f1(t+h,Y1+k3,Y2+m3);
    m4 = h*f2(t+h,Y1+k3,Y2+m3);

    Y1 = Y1 + (k1+2*k2+2*k3+k4)/6;
    Y2 = Y2 + (m1+2*m2+2*m3+m4)/6;

    Y(i) = Y1;

    t = t + h;

end

end
```

Program for finding analytic solution of (21) for general γ and ω

```
function [y] = analyticsol (gamma,omega)

delta = 0;
Omega = 1;
a = 1;

rho = -(gamma)/2;
sigma = sqrt((Omega)^2-(rho)^2);
U = (a*(Omega^2-omega^2))/((Omega^2-omega^2)^2+(gamma*omega)^2);
V = (-a*gamma*omega)/((Omega^2-omega^2)^2+(gamma*omega)^2);
P = -V;
Q = (V*rho-U*omega)/(sigma);

y = @(t) exp(t*rho)*(P*cos(t*sigma)+Q*sin(t*sigma)) + U*sin(t*omega) + V*cos(t*omega);
```

Program for question 6

```
q6.m

h1 = 0.4;
h2 = 0.2;
h3 = 0.1;

Z1 = RK4method2(1,0,1,sqrt(3),1,10,25);
Z2 = RK4method2(1,0,1,sqrt(3),1,10,50);
Z3 = RK4method2(1,0,1,sqrt(3),1,10,100);

y = analyticsol (1,sqrt(3))


for i=1:25
    E1(i) = Z1(i) - y(i*h1);
    fprintf('%.2f %.8f %.8f %.8f \n' ,i*h1, Z1(i), y(i*h1), E1(i) );
end

for i=1:50
    E2(i) = Z2(i) - y(i*h2);
    fprintf('%.2f %.8f %.8f %.8f \n' ,i*h2, Z2(i), y(i*h2), E2(i) );
end

for i=1:100
    E3(i) = Z3(i) - y(i*h3);
```

```
    fprintf('%.2f %.8f %.8f %.8f \n' ,i*h3, Z3(i), y(i*h3), E3(i) );
end
```

Program to produce figure 2

```
q7plot.m

y11 = analyticsol (0.25,1);
y13 = analyticsol (1.00,1);

y22 = analyticsol (0.50,2);
y24 = analyticsol (1.90,2);

Y11 = RK4method2( 0.25,0,1,1,1,40,40);
Y13 = RK4method2( 1.00,0,1,1,1,40,40);

Y22 = RK4method2( 0.50,0,1,2,1,40,40);
Y24 = RK4method2( 1.90,0,1,2,1,40,40);

x = linspace(1.0,40,40);

figure(11)

plot(x,Y11)
hold on
fplot(@(y) y11(y),[0 40],'r')
hold off


figure(13)

plot(x,Y13)
hold on
fplot(@(y) y13(y),[0 40],'r')
hold off


figure(22)

plot(x,Y22)
hold on
fplot(@(y) y22(y),[0 40],'r')
hold off


figure(24)
```

```
plot(x,Y24)
hold on
fplot(@(y) y24(y),[0 40],'r')
hold off
```