Problem 1

$\boxed{1}$ $u''(x) = f(x)$ on $(0,1)$

$u'(0)=0$, $u(1)=0$

For $x < \bar{x}$:

$u''(x) = 0 \Rightarrow \int_0^x u''(t)\,dt = 0 \Rightarrow$

$\Rightarrow u'(x) - \underset{\substack{\shortparallel \\ 0}}{u'(0)} = 0 \Rightarrow u'(x) = 0 \Rightarrow u(x) = c$

from b.c.

for $x > \bar{x}$:

$u''(x) = 0 \Rightarrow \int u''(t)\,dt = c_1 \Rightarrow$

$\Rightarrow u'(x) = c_1 \Rightarrow \int_x^1 u'(t)\,dt = c_1 \int_x^1 dt \Rightarrow$

$\Rightarrow \underset{\substack{\shortparallel \\ 0}}{u(1)} - u(x) = c_1(1-x) \Rightarrow u(x) = c_1(x-1)$

from b.c.

Since $u''(x) = \delta(x-\bar{x}) \Rightarrow \int_{\bar{x}-\epsilon}^{\bar{x}+\epsilon} u''(x)\,dx = 1$

$u'(\bar{x}+\epsilon) - u'(\bar{x}-\epsilon) = 1$

We can make ϵ infinitely small, so that

$u'(\bar{x}-\epsilon) = u'(\bar{x}_-) = (c)' = 0$

$\Rightarrow c_1 - 0 = 1 \Rightarrow c_1 = 1$

$u'(x+\epsilon) = u'(\bar{x}_+) = c_1$

Since $\lim_{x\to\bar{x}^-} u(x) = \lim_{x\to\bar{x}^+} u(x) \Rightarrow c = c_1(\bar{x}-1) \Rightarrow$

$\Rightarrow c = \bar{x} - 1$

Hence: $G(x,\bar{x}) = \begin{cases} \bar{x}-1 & \text{for } x < \bar{x} \\ x-1 & \text{for } x > \bar{x} \end{cases}$

$u''(x) = 0$ on $(0,1)$

$u'(0) = 1$, $u(1) = 0$

Itegrating twice : $u(x) = ax + b$

$u'(0) = a = 1$, $u(1) = a + b = 0 \Rightarrow b = -1$

$\Rightarrow G_0(x) = x - 1$

$u''(x) = 0$ on $(0,1)$

$u'(0) = 0$, $u(1) = 1$

$u'(0) = a = 0$, $u(1) = a + b = b = 1$

$\Rightarrow G_1(x) = 1$

$\boxed{2.}$ $A = \frac{1}{0.0625} \cdot \begin{bmatrix} -0.25 & 0.25 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0.0625 \end{bmatrix}$

Let $B = A^{-1}$. From the textbook:

$B_{i0} = G_0(x_i) = x_i - 1$

$B_{i,4} = G_1(x_i) = 1$

$B_{i,j} = h\, G(x_i, x_j) = \begin{cases} 0.25(x_j - 1) & i = 1, 2, \ldots j \\ 0.25(x_i - 1) & i = j, j+1, \ldots, 3 \end{cases}$

Since $h = 0.25$: $x_0 = 0$, $x_1 = 0.25$, $x_2 = 0.5$, $x_3 = 0.75$, $x_4 = 1$

$B_{00} = x_0 - 1 = -1$, $B_{10} = x_1 - 1 = -0.75$, $B_{20} = x_2 - 1 = -0.5$,

$B_{30} = x_3 - 1 = -0.25$, $B_{40} = x_4 - 1 = 0$

$B_{04} = B_{14} = B_{24} = B_{34} = B_{44} = 1$

$B_{01} = B_{11} = -0.1875$ $\quad B_{21} = -0.125$

$B_{31} = B_{32} = -0.0625$, $\quad B_{41} = B_{42} = B_{43} = 0$

$B_{02} = B_{12} = B_{22} = -0.125$, $B_{03} = B_{13} = B_{23} = B_{33} = -0.0625$

Hence $A^{-1} = \begin{bmatrix} -1 & -0.1875 & -0.125 & -0.0625 & 1 \\ -0.75 & -0.1875 & -0.125 & -0.0625 & 1 \\ -0.5 & -0.125 & -0.125 & -0.0625 & 1 \\ -0.25 & -0.0625 & -0.0625 & -0.0625 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$

$\boxed{\text{Problem 2}}$

See the attachments,

$\boxed{\text{Problem 3}}$

$\boxed{1}$ From Matlab: $E(h) = 5.95256 \cdot h^{1.97716} \approx O(h^2)$

You can find the corresponding outputs at the end of the pdf.

$\boxed{2}$ $LTE = \frac{1}{3}(h_{i-1} - h_i)\, u'''(x_i) + O(h^2)$

$$h_{i-1} = x_i - x_{i-1} \approx h\, X'(z_i)$$

$$h_i = x_{i+1} - x_i \approx h\, X'(z_{i+1})$$

$$\frac{1}{3}(h_{i-1} - h_i)\, u'''(x_i) = \frac{1}{3} h\,(X'(z_i) - X'(z_{i+1}))\, u'''(x_i)$$

$$X'(z_{i+1}) = X'(z_i + ih) \approx X'(z_i) + X''(z_i)\, ih + O(h^2)$$

Hence if X is at least twice differentiable :

$$\begin{aligned} LTE &= \frac{1}{3}(h_{i-1} - h_i)\, u'''(x_i) + O(h^2) \\ &= \frac{1}{3} h\,(X'(z_i) - X'(z_{i+1}))\, u'''(x_i) + O(h^2) \\ &= \frac{1}{3} h\,(X''(z_i)\, ih + O(h^2))\, u'''(x_i) + O(h^2) \\ &= \frac{1}{3} i h^2\, X''(z_i)\, u'''(x_i) + O(h^2) = O(h^2) \end{aligned}$$

$\boxed{3}$ Average order of accuracy is $h^{1.74939}$

You can find the corresponding outputs at the end of the pdf.

$\boxed{\text{Problem 4}}$

$\boxed{1}$ $u''(x) + u(x) = 0$ in (a,b)

$$u(a) = \alpha, \quad u(b) = \beta$$

$u(x) = c_2 \sin(x) + c_1 \cos(x)$ → general solution

$$u(a) = c_2 \sin(a) + c_1 \cos(a) = \alpha \Rightarrow c_1 = \frac{\alpha - c_2 \sin(a)}{\cos(a)}$$

$$u(b) = c_2 \sin(b) + c_1 \cos(b) = \beta$$

$$c_2 \sin(b) + \frac{\alpha - c_2 \sin(a)}{\cos(a)} \cos(b) = \beta$$

$$c_2 \left(\sin(b) - \frac{\sin(a)\cos(b)}{\cos(a)} \right) = \beta - \alpha \frac{\cos(b)}{\cos(a)}$$

$$c_2 = \frac{\beta - \alpha \cos(b)/\cos(a)}{\sin(b) - \sin(a)\cos(b)/\cos(a)}$$

You can find the corresponding outputs at the end of the pdf. See the attachments for the code.

$\boxed{2}$

$$\left.\begin{aligned} u(0) &= c_2 \sin 0 + c_1 \cos 0 = c_1 = \alpha \\ u(\pi) &= c_2 \sin\pi + c_1 \cos\pi = -c_1 = \beta \end{aligned}\right\} \quad \alpha = -\beta$$

For $\alpha = -\beta = c_1$, BVP has solutions:

$u(x) = c_2 \sin(x) + \alpha \cos(x)$ is a family of solutions

You can find the corresponding sketch for $\alpha = 2$ and $c_2 = -10:2:10$ at the end of the pdf.

$\boxed{3}$ You can find the corresponding outputs at the end of the pdf. See the attachments for the code.

For $\beta = -1$, $u_n(x) \to u(x) = \cos(x)$

Because, bvp2.m makes $c_2 = 0$.

For $\beta = 1$, numerical solution just approximates $u(x) = 0$ and error does not change, because $\alpha \neq -\beta$.

$\boxed{4}$

$$A = \frac{1}{h^2} \cdot \begin{bmatrix} h^2 & 0 & & & & \\ 1 & -2+h^2 & 1 & & & \\ & 1 & -2+h^2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2+h^2 & 1 \\ & & & & 0 & h^2 \end{bmatrix}$$

$|A-\lambda I| = (1-\lambda)\det A_1 - \frac{1}{h^2}\det A_2$

Where A_1 is $(A-\lambda I)$ without 1^{st} row and 1^{st} column.

and A_2 is $(A-\lambda I)$ without 2^{nd} row and 1^{st} column.

Notice that A_2 has a row of all zeros, hence $\det(A_2)=0$.

$|A-\lambda I| = (1-\lambda)\det A_1 = (1-\lambda)\left((1-\lambda)\det A_{11} + \frac{1}{h^2}\cdot\right.$

$\left.\cdot \det A_{12}\right)$

Where A_{11} is A_1 without last row and last column.

and A_{12} is A_1 without penultimate row and last column.

Notice that A_{12} has a row of all zeros, hence $\det(A_{12})=0$.

$|A-\lambda I| = (1-\lambda)^2 \det A_{11}$

Using the textbook as a guidance we can find that $\lambda_p = \dfrac{2\cos(ph) - 2+h^2}{h^2}$ for $p = 1, 2, \ldots, m$

$u_j^p = \sin(pjh)$ — notice that in our case we have $\cos(ph)$ instead of $\cos(p\pi h)$ because our $h = \frac{\pi}{m}$ instead of $h = \frac{1}{m}$

$u_0^p = u_{m+1}^p = 0$

$$(A_{11}u^p)_j = \frac{1}{h^2}\left(u_{j-1}^p - (2-h^2)u_j^p + u_{j+1}^p\right)$$

$$= \frac{1}{h^2}\left(\sin(p(j-1)h) - (2-h^2)\sin(pjh) + \sin(p(j+1)h)\right)$$

$$= \frac{1}{h^2}\left(\sin(pjh)\cos(ph) - (2-h^2)\sin(pjh) + \sin(pjh)\cos(ph)\right)$$

$$= \lambda_p u_j^p \quad \text{for } j = 1, \ldots, m$$

$$\frac{1}{h^2}\cdot h^2 \cdot u_j^p = 0 = \lambda_p u_j^p \quad \text{for } j = 0, m+1$$

$$\lim_{h\to 0}\lambda_p = \lim_{h\to 0}\frac{\overbrace{2\cos(ph) - 2+h^2}^{\to 0}}{\underbrace{h^2}_{\to 0}} \overset{\text{L'Hopital}}{\approx} \lim_{h\to 0}\frac{\overbrace{-2p\sin(ph) + 2h}^{\to 0}}{\underbrace{2h}_{\to 0}} \overset{\text{L'Hopital}}{\approx}$$

$$\approx \lim_{h\to 0} -(p)^2\cos(ph) + 1 = -p^2+1$$

So for $p=1$: $\lim_{h\to 0}\lambda_1 = -1+1 = 0$

Since A^{-1} is also symmetric:

$$\|A^{-1}\|_2 = \rho(A^{-1}) = \max_{1 \le p \le m} |(\lambda_p)^{-1}| = \frac{1}{\min_{1 \le p \le m} |\lambda_p|}$$

Since as $h \to 0$ $\min|\lambda_p| \to 0$: $\|A^{-1}\|_2 \to \infty$

$\boxed{\text{Problem 5}}$

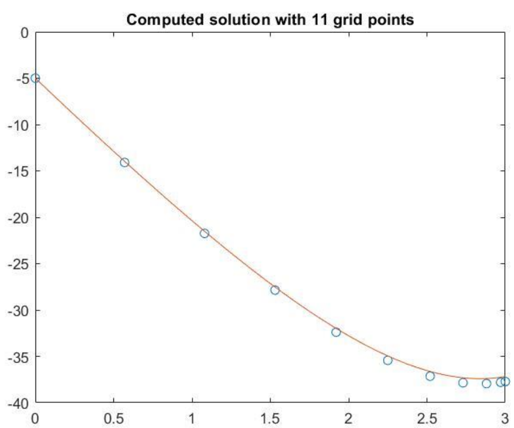
You can find the corresponding outputs at the end of the pdf. See the attachments for the code.

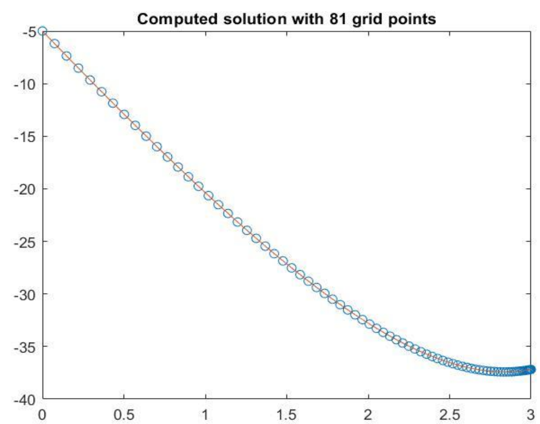
Problem 2.

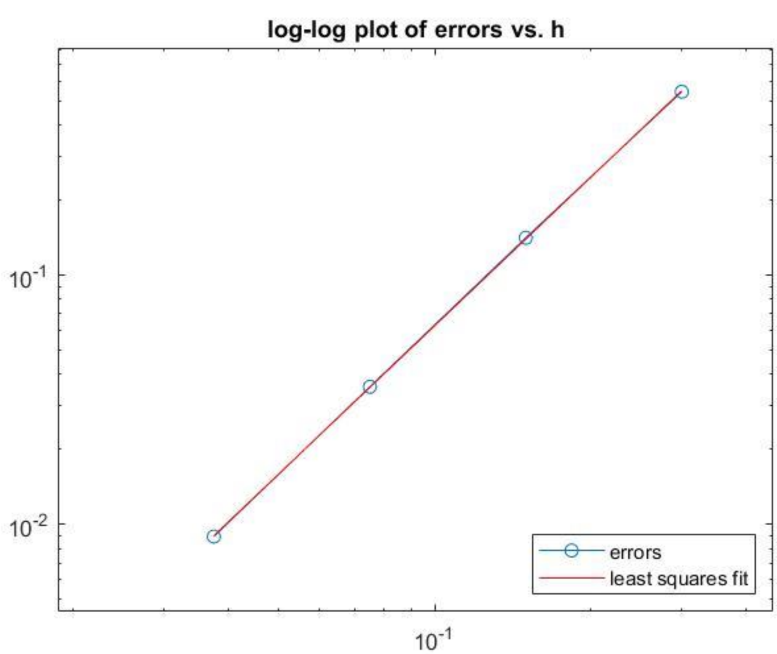
1. Modifications made:

```
% first row for Dirichlet BC at ax:
A(1,1:3) = fdcoeffF(0, x(1), x(1:3));

% interior rows:
for i=2:m1
   A(i,i-1:i+1) = fdcoeffF(2, x(i), x((i-1):(i+1)));
end

% last row for Neumann BC at bx:
A(m2,m:m2) = fdcoeffF(1,x(m2),x(m:m2));
```

Outputs:

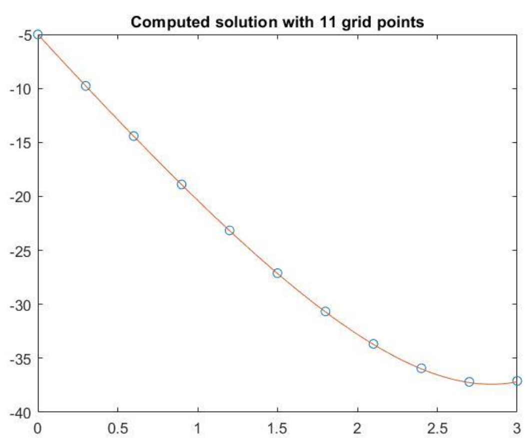


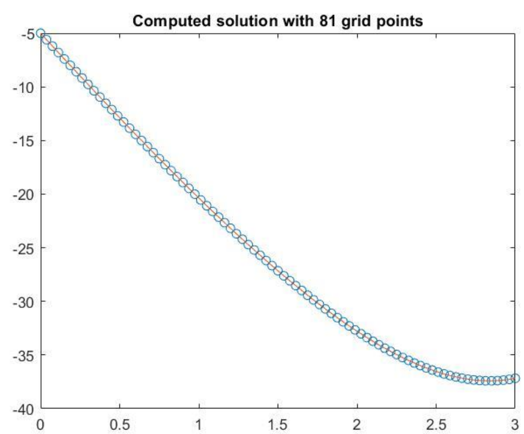


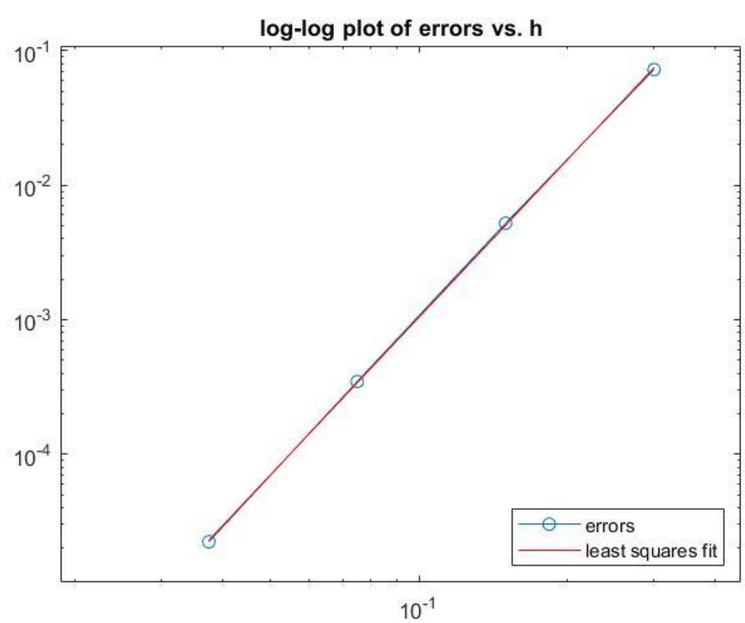


2. Modifications made:

```
% first row for Dirichlet BC on u'(x(1))
A(1,1:5) = fdcoeffF(0, x(1), x(1:5));
% second row for u''(x(2))
A(2,1:6) = fdcoeffF(2, x(2), x(1:6));

% interior rows:
for i=3:m
   A(i,i-2:i+2) = fdcoeffF(2, x(i), x((i-2):(i+2)));
   end

% next to last row for u''(x(m+1))
A(m1,m-3:m2) = fdcoeffF(2,x(m1),x(m-3:m2));
% last row for Neumann BC on u(x(m+2))
A(m2,m-2:m2) = fdcoeffF(1,x(m2),x(m-2:m2));
```





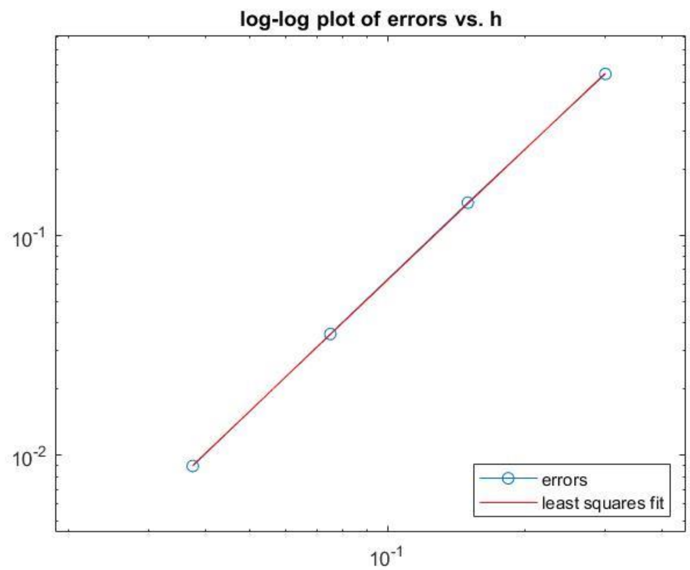


Problem 3.

1. rtlayer results:

h	error	ratio	observed order
0.30000	5.45012e-01	NaN	NaN
0.15000	1.41393e-01	3.85460	1.94658
0.07500	3.56757e-02	3.96328	1.98670
0.03750	8.93948e-03	3.99080	1.99668

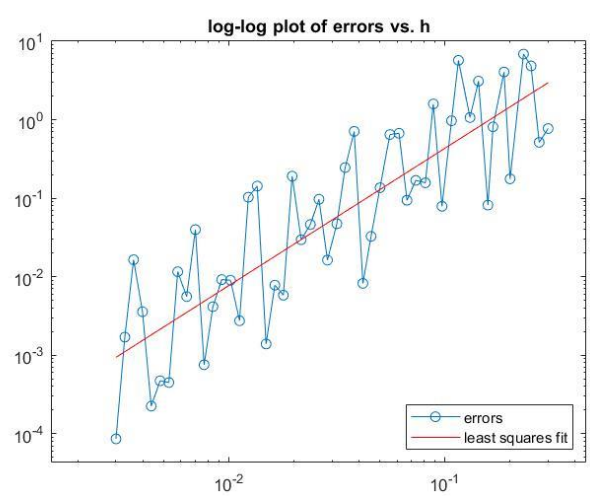
```
Least squares fit gives E(h) = 5.95256 * h^1.97766
```



   Indeed since 1.97766 ~ 2, second-order accuracy is observed.

2. Check the handwritten proof.
3. random results with m from 10 to 1000:

h	error	ratio	observed order
0.30000	7.71268e-01	NaN	NaN
0.27273	5.14687e-01	1.49852	4.24380
0.25000	4.84159e+00	0.10631	-25.76032
0.23077	6.85630e+00	0.70615	-4.34672
0.20000	1.76170e-01	38.91873	25.58668
0.18750	4.04848e+00	0.04352	-48.57020
0.16667	8.11682e-01	4.98776	13.64362
0.15789	8.15744e-02	9.95020	42.49512
0.14286	3.10441e+00	0.02628	-36.36027
0.13043	1.06393e+00	2.91787	11.77128
0.11538	5.70502e+00	0.18649	-13.69777
0.10714	9.68836e-01	5.88853	23.92463
0.09677	7.88057e-02	12.29399	24.65165
0.08824	1.58603e+00	0.04969	-32.49863
0.08108	1.57003e-01	10.10193	27.35097
0.07317	1.68526e-01	0.93162	-0.68997
0.06667	9.41136e-02	1.79067	6.25832
0.06122	6.70772e-01	0.14031	-23.06219
0.05556	6.46659e-01	1.03729	0.37678
0.05000	1.35806e-01	4.76164	14.81192
0.04545	3.26210e-02	4.16315	14.96453
0.04167	8.21861e-03	3.96916	15.84338
0.03797	7.09451e-01	0.01158	-48.04923
0.03448	2.45521e-01	2.88957	11.00048
0.03158	4.72927e-02	5.19153	18.72287
0.02857	1.62595e-02	2.90862	10.66789
0.02609	9.69222e-02	0.16776	-19.62403
0.02381	4.61798e-02	2.09880	8.11570



```
Least squares fit gives E(h) = 24.3375 * h^1.74939
```

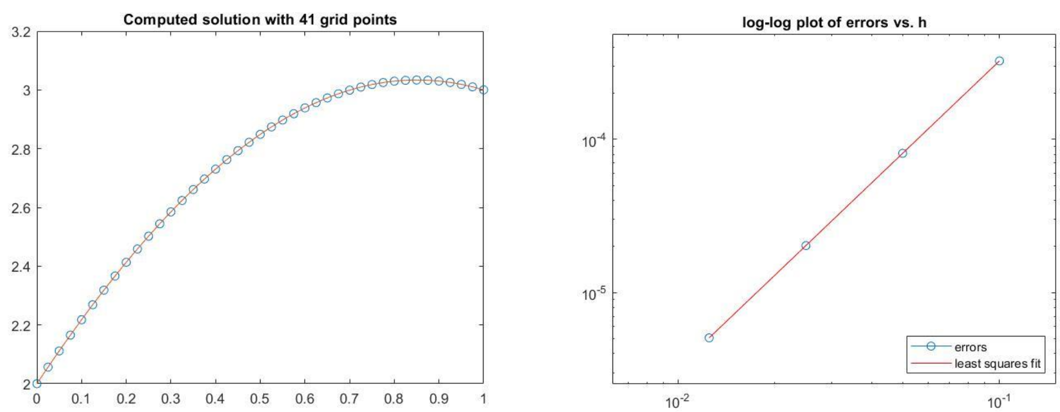
Problem 4.

1. Modifications:

```
f = @(x) 0*x;  % right hand side function
c2 = (beta - sigma*cos(bx)/cos(ax))/(sin(bx)-sin(ax)*cos(bx)/cos(ax));
c1 = (sigma - c2*sin(ax))/cos(ax);
utrue = @(x) c2*sin(x) + c1*cos(x);  % true soln
```

```
A = A + eye(size(A));
A(1,1) = A(1,1)-1;
A(m2,m2) = A(m2,m2)-1;
% solve linear system:
U = A\F;
```

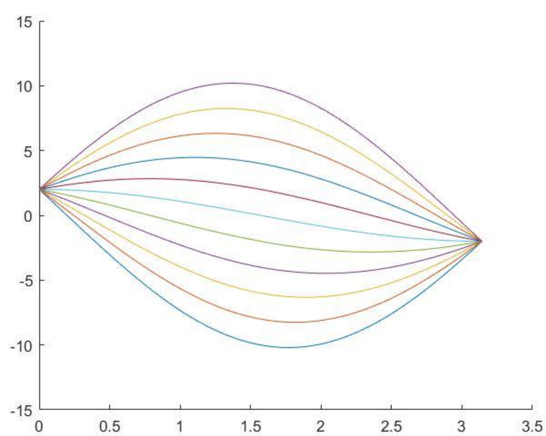
Outputs:



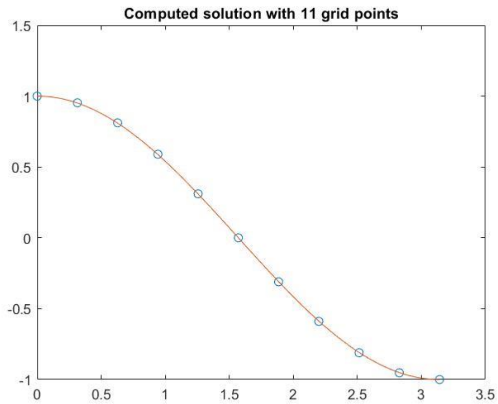
h	error	ratio	observed order
0.10000	3.24687e-04	NaN	NaN
0.05000	8.10848e-05	4.00430	2.00155
0.02500	2.02705e-05	4.00014	2.00005
0.01250	5.06999e-06	3.99812	1.99932

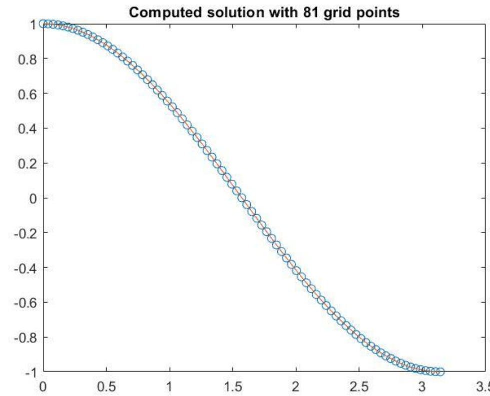
```
Least squares fit gives E(h) = 0.0324764 * h^2.00028
```

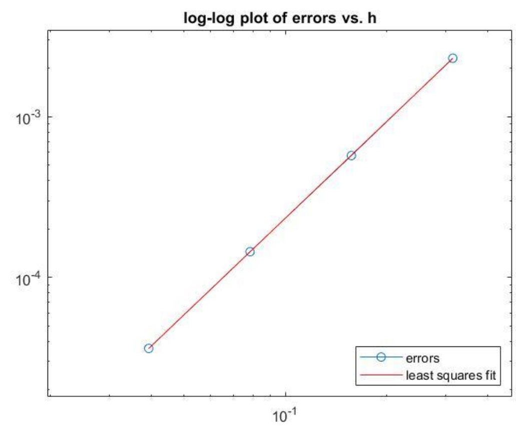
2. Family of solutions:



3. beta = 1:







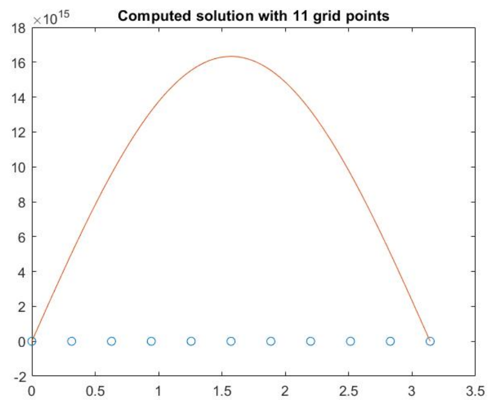
h	error	ratio	observed order
0.31416	2.31489e-03	NaN	NaN
0.15708	5.73244e-04	4.03823	2.01372
0.07854	1.44354e-04	3.97111	1.98954
0.03927	3.60617e-05	4.00296	2.00107

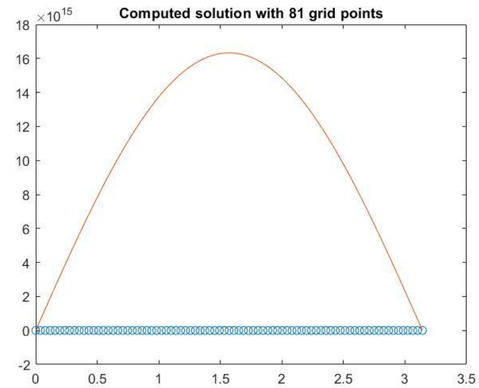
Least squares fit gives E(h) = 0.0233813 * h^2.00026

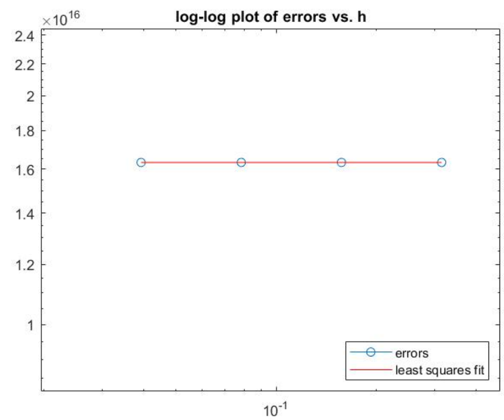
beta = -1:

h	error	ratio	observed order
0.31416	1.63312e+16	NaN	NaN
0.15708	1.63312e+16	1.00000	-0.00000
0.07854	1.63312e+16	1.00000	-0.00000
0.03927	1.63312e+16	1.00000	-0.00000

Least squares fit gives E(h) = 1.63312e+16 * h^-2.70735e-13



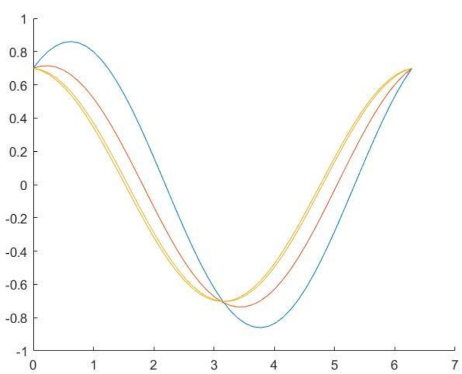


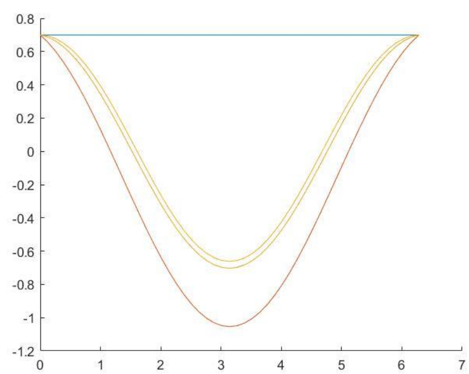


Problem 5.

1. Implementation is in file p5.m.

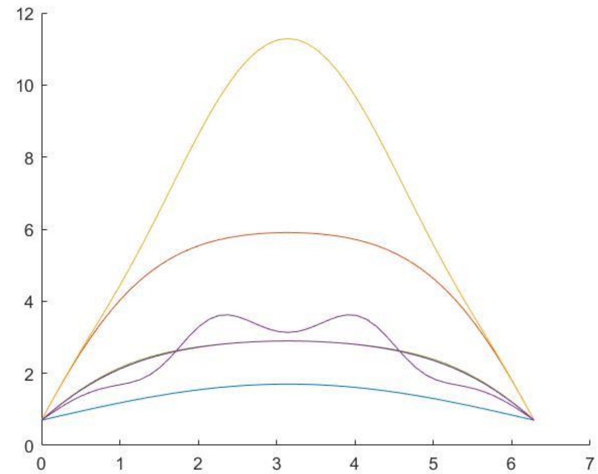
   Testing using figure 2.4 boundary conditions:





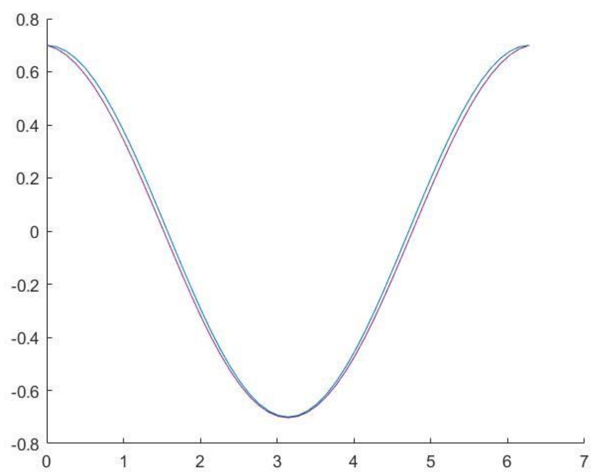
As it can be seen plots are identical to figure 2.4.

Testing using figure 2.5 boundary conditions:

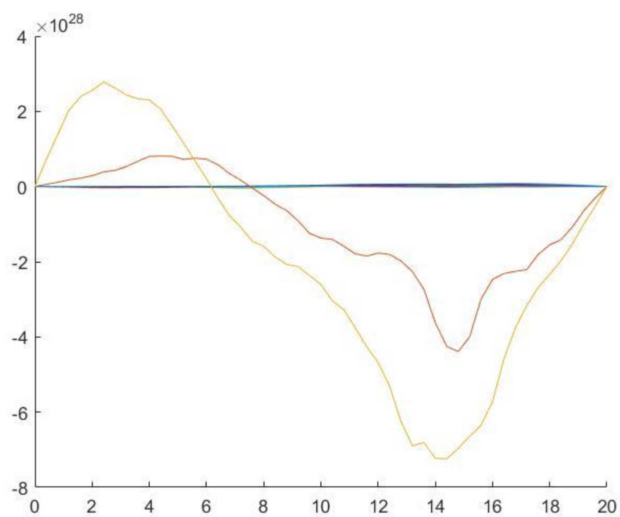


As it can be seen plots are identical to figure 2.5.

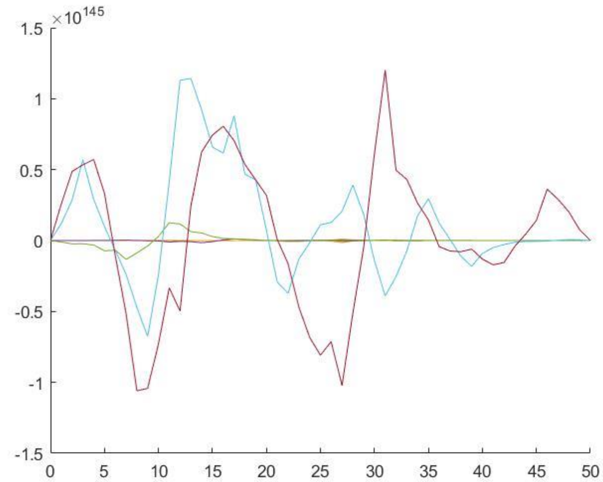
For u_0 (t) = 0.7*cos(t), we can get this plot:



2. T = 20 and iterations = 100:



T = 50 and iterations = 300:



Notice that max value is increasing as T increases, hence max_i theta_i should approach infinity as T increases.

Implementation:

```
T = 2*pi;
ax = 0;
bx = T;
sigma = 0.7;    % Dirichlet boundary condition at ax
beta = 0.7;      % Dirichlet boundary condtion at bx

m1 = 50;
m2 = m1 + 1;
m = m1 - 1;                        % number of interior grid points
h = (bx-ax)/m1;  % average grid spacing, for convergence tests

% set grid points:
gridchoice = 'uniform';           % see xgrid.m for other choices
% gridchoice = 'random';
t = xgrid(ax,bx,m,gridchoice);

% set up matrix A (using sparse matrix storage):
A = spalloc(m2,m2,3*m2);    % initialize to zero matrix

% first row for Dirichlet BC at ax:
A(1,1:3) = fdcoeffF(0, t(1), t(1:3));

% interior rows:
for i=2:m1
    A(i,i-1:i+1) = fdcoeffF(2, t(i), t((i-1):(i+1)));
end
```

```
% last row for Dirichlet BC at bx:
A(m2,m:m2) = fdcoeffF(0,t(m2),t(m:m2));
G = @(x) A*x + sin(x);  % true soln

% set up matrix JM (using sparse matrix storage):
JM = spalloc(m2,m2,3*m2);   % initialize to zero matrix

JM(1,1:3) = fdcoeffF(2, t(1), t(1:3));

% interior rows:
for i=2:m1
    JM(i,i-1:i+1) = fdcoeffF(2, t(i), t((i-1):(i+1)));
end

JM(m2,m:m2) = fdcoeffF(2,t(m2),t(m:m2));
J = @(x) JM + diag(cos(x));  % true soln

theta = 0.7*cos(t);
% theta = 0.7 + sin(t/2);
% theta = 0.7*cos(t) + 0.5*sin(t);
% theta = 0.7*ones(size(t));
hold on
plot(t,theta)  % plot true solution
hold off
for k=1:10
    % solve linear system:
    J1 = J(theta);
    J1 = J1(2:m2-1,2:m2-1);
    G1 = G(theta);
    G1 = G1(2:m2-1);
    G1 = -G1;
    dtheta = J1\G1;
    theta(2:m2-1) = theta(2:m2-1) + dtheta;
    hold on
    plot(t,theta)  % plot true solution
    hold off
end
```