Problem P7: Suppose this table of 'data' give samples of the function $Z(h)$:

h	1.0	0.5	0.1	0.05	0.01	0.005
Z(h)	28.43	10.747	0.5540	0.5555	0.04849	0.005521

This data may be fitted by a function $f(h) = Ch^p$ for some values C and p. Find these values by fitting a straight line to the logarithms of the data; in Matlab you may use polyfit. Then graph the data and show the fitted line on the same axes, using Matlab's loglog or similar.

Solution:
Let $ln(f(x))$ be the linear regression fitted to the log transformed data,

$$ln(f(x)) = x_1(ln(h)) + x_0$$

Solving $f(x)$ we get the following,

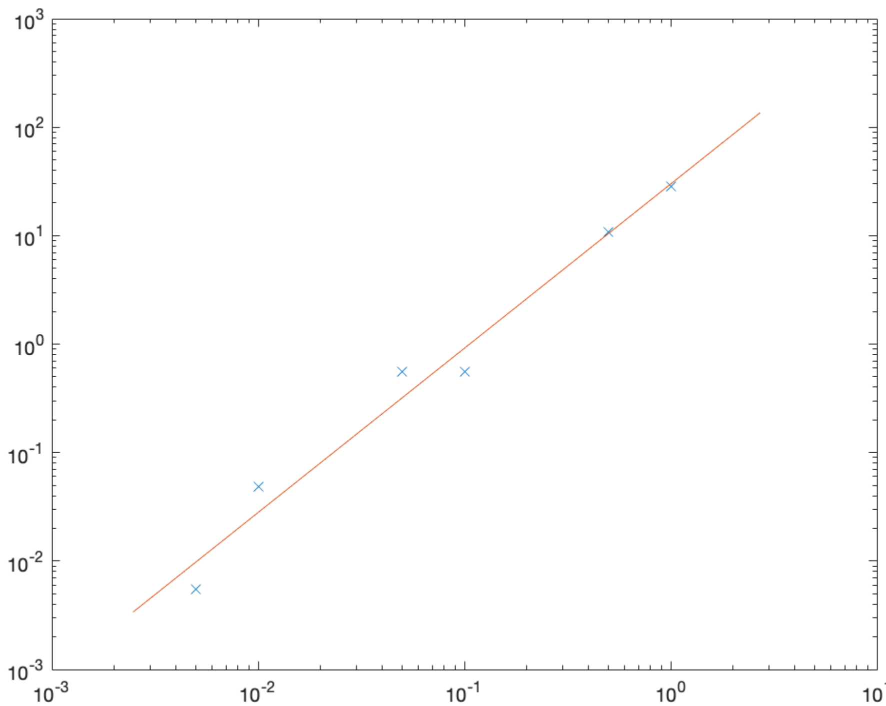
$$\begin{aligned} ln(f(x)) &= x_1(ln(h)) + x_0, \\ ln(f(x)) &= ln(h^{x_1}) + x_0, \\ e^{ln(f(x))} &= e^{ln(h^{x_1})+x_0}, \\ f(x) &= e^{x_0}e^{ln(h^{x_1})}, \\ f(x) &= e^{x_0}h^{x_1}. \end{aligned}$$

So $C = e^{x_0}$ and $p = x_1$. The following code fits the linear regression to the log transformed data and generates a loglog plot with the original data. From the code we see that $C \approx 29.8892$ and $p \approx 1.5143$.
Console:

```
% Importing 'data'
>> Zh = [28.43   10.747   0.5540   0.5555   0.04849    0.005521]
>> h = [1.0     0.5     0.1     0.05     0.01      0.005]

% Log Transformed linear regression
>> [P, S] = polyfit(log(h), log(Zh), 1)
P =
     1.5143      3.3978

% Polyfit returns coefficients in descending order.
>> p = P(1)
     1.5143

>> C = exp(P(2))
    29.8991

% Pulling range for plot
>> log(h)
     0     -0.6931     -2.3026     -2.9957     -4.6052     -5.2983

% Evaluating Regression
>> x1 = linspace(-6,1,100);
```

```
>> y1 = polyval(P,x1);

% Generating loglog plot
>> loglog(h, Zh)
>> hold on
% We have to untransform these values to
% plot them against the original data.
>> loglog(exp(x1), exp(y1))
>> hold off
```

Figure 1: Loglog plot of $Z(h)$ with $f(h)$ in red.



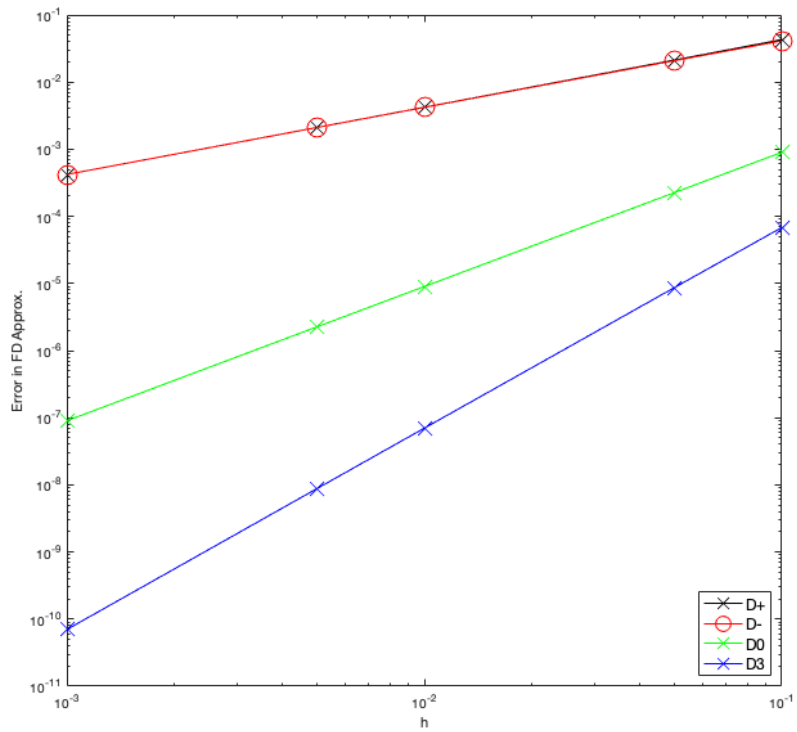
Problem P8: Reproduce Figure 1.2 on page 6 of the textbook. In particular, write a code which generates the data show in Table 1.1, by doing the calculations described by Example 1.1, with $u(x) = \sin(x)$ and $\overline{x} = 1$. Then generate the Figure, which has logarithmic scaling on both axes.

Solution:

Code:

```
function [table] = figure12(u,du,x,h0,n)
% This function takes in a function u(x), it's derivative du(x)
% a point x where we want to approximate the derivative
% an initial h0 spacing and an n number of iterations
% for adjusting the spacing h by the following recurrence relation.
% h_n = h_{n - 2}/10 h_0 = h0, h_1 = h0/2

% Very hacky way of putting together recurrence for h values.
h = [h0 h0/2];
for i = 2:n-1
 h = [h h(i - 1)/10];
end
x = ones(1,n)*x;


DP = (u(x+h) - u(x))./h - du(x);
DM = (u(x) - u(x-h))./h - du(x);
D0 = (u(x+h) - u(x-h))./(2.*h) - du(x);
D3 = (2.*u(x+h) + 3.*u(x) - 6.*u(x-h) + u(x-2.*h))./(6.*h) - du(x);


table = [h' DP' DM' D0' D3'];


hold off
loglog(table(:, 1), abs(table(:, 2)), 'k-x', 'LineWidth',.75, 'MarkerSize', 15)
hold on
loglog(table(:, 1), abs(table(:, 3)), 'r-o','LineWidth',.75, 'MarkerSize', 15)
loglog(table(:, 1), abs(table(:, 4)), 'g-x','LineWidth',.75, 'MarkerSize', 15)
loglog(table(:, 1), abs(table(:, 5)), 'b-x','LineWidth',.75, 'MarkerSize', 15)
legend('D+', 'D-', 'D0', 'D3', 'Location','southeast','FontSize', 12)
ylabel('Error in FD Approx.')
xlabel('h')

end
```

Figure 2: Loglog plot of Error in Various FD Approx.



Problem P9: (a) Use the method of undetermined coefficient to set up a 5x5 linear system that determine the fourth-order centered finite difference approximation to $u''(x)$ based on 5 equally spaced points, namely

$$u''(x) = c_{-2}u(x-2h) + c_{-1}u(x-h) + c_0u(x) + c_1u(x+h) + c_2u(x+2h) + O(h^4)$$

In particular, expand $u(x-2h), u(x-h), u(x+h), u(x+2h)$ in Taylor series. Then collect terms on the right side of the above equation to generate a square linear system $Ac = g$ in unknowns $c_{-2}, c_{-1}, c_0, c_1, c_2$. This system will have numerical entries in the matrix A, but the entries of vector G will depend on h.

Solution:

Applying Taylor's Theorem we can expand each function $u(x-2h), u(x-h), u(x+h)$ and $u(x+2h)$ in terms of $u(x)$ and it's derivatives. Doing so to the fourth order we get the following,

$$
\begin{aligned}
u(x-2h) &= u(x) - 2hu'(x) + \frac{1}{2}(2h)^2u''(x) - \frac{1}{6}(2h)^3u'''(x) + \frac{1}{24}(2h)^4u''''(x) + O(h^5),\\
u(x-h) &= u(x) - hu'(x) + \frac{1}{2}h^2u''(x) - \frac{1}{6}h^3u'''(x) + \frac{1}{24}h^4u''''(x) + O(h^5),\\
u(x+h) &= u(x) + hu'(x) + \frac{1}{2}h^2u''(x) + \frac{1}{6}h^3u'''(x) + \frac{1}{24}h^4u''''(x) + O(h^5),\\
u(x+2h) &= u(x) + 2hu'(x) + \frac{1}{2}(2h)^2u''(x) + \frac{1}{6}(2h)^3u'''(x) + \frac{1}{24}(2h)^4u''''(x) + O(h^5).
\end{aligned}
$$

By substitution and collecting like terms we get the following,

$$
\begin{aligned}
D_4^2(x) = &(c_{-2} + c_{-1} + c_0 + c_1 + c_2)u(x)\\
&+ (-2c_{-2} - c_{-1} + c_1 + 2c_2)hu'(x)\\
&+ (2c_{-2} + \frac{1}{2}c_{-1} + \frac{1}{2}c_1 + 2c_2)h^2u''(x)\\
&+ \left(-\frac{4}{3}c_{-2} - \frac{1}{6}c_{-1} + \frac{1}{6}c_1 + \frac{4}{3}c_2\right)h^3u'''(x)\\
&+ \left(\frac{1}{6} + \frac{1}{24} + 0 + \frac{1}{24} + \frac{1}{6}\right)h^4u''''(x)\\
&+ O(h^5).
\end{aligned}
$$

By method of undetermined coefficient we get the following system of equations.

$$
\begin{bmatrix}
1 & 1 & 1 & 1 & 1\\
-2 & -1 & 0 & 1 & 2\\
2 & \frac{1}{2} & 0 & \frac{1}{2} & 2\\
-\frac{4}{3} & -\frac{1}{6} & 0 & \frac{1}{6} & \frac{4}{3}\\
\frac{2}{3} & \frac{1}{24} & 0 & \frac{1}{24} & \frac{2}{3}
\end{bmatrix}
\begin{bmatrix}
c_{-2}\\ c_{-1}\\ c_0\\ c_1\\ c_2
\end{bmatrix}
=
\begin{bmatrix}
0\\ 0\\ \frac{1}{h^2}\\ 0\\ 0
\end{bmatrix}
$$

(b) Use Matlab to solve the linear system from part (a). A recommended way to do this is to use $h = 1$ in the vector g and solve the system numerically using the 'backslash' method. Then write down the answer in the form like (1.11), inserting the correct power of h. Use $h = .5$ to confirm that that you've captured the correct powers.

Solution:
Solving the system numerically we get the following finite difference approximation,

$$
D_5^2(x) = \frac{1}{h^2}\left[-\frac{1}{12}u(x-2h) + \frac{4}{3}u(x-h) - \frac{5}{2}u(x) + \frac{4}{3}u(x+h) - \frac{1}{12}u(x+2h)\right].
$$

As suggested we solved the system numerically with $h = 1$, put the approximation in the form of (1.11) then solved with $h = .5$, comparing the results to see that we must divide the coefficients by a factor of $1/h^2$.

Console:

```
>> A = [1      1      1      1      1  ;
       -2     -1      0      1      2;
        2    (1/2)    0    (1/2)  2;
      -(4/3)    -(1/6)  0  (1/6)      (4/3);
       (2/3)     (1/24)     0     (1/24)  (2/3)];

>> g = [0 0 1 0 0]'

>> rats(A\g)

     -1/12
      4/3
     -5/2
      4/3
     -1/12

>> fdcoeffh1 = A\g;

>> g = [0 0 1/(.5^2) 0 0]';

>> fdcoeffh2 = A\g;

>> (fdcoeffh2) - ((1/(.5^2)).*fdcoeffh1)

      0
      0
      0
      0
      0
```

Problem P10: In Section 2.4 the textbook uses finite differences to convert the boundary value problem,

$$u''(x) = f(x), \qquad u(0) = \alpha, \qquad u(1) = \beta$$

into matrix equation $AU = F$, with A and F given in (2.10). For any integer $m \geq 1$, this method is based on a grid with $h = 1/(m+1)$ and $x_j = jh$. There are m unknowns $U_1, U_2, \ldots, U_m$ located at the interior nodes $x_1, \ldots, x_m$. Note that finite difference approximation D^2 from equation (1.13) is used for the u'' term.

Assume q, x_L, x_R are real numbers with $x_L < x_R$. Similar to the method in Section 2.4, create a finite difference approximation for the problem

$$u''(x) + qu(x) = f(x), \qquad u(x_L) = \alpha, \qquad u(x_R) = \beta$$

Use the same approximation D^2 for u''. Use the same grid indexing with m unknowns

$U_1, \dots, U_m$ and give the new formulas for x_j and the mesh width h. State, in detail, A and F in $AU = F$

Solution:
First note that a grid with $m + 2$ indices across an interval from x_L to x_R will have $m - 1$ spaces, each with size $h = (x_R - x_L)/(m-1)$. Our formula for each x_j is given by $x_j = jh + x_L$. Recall that the finite difference approximation for D^2 used in (1.13) goes as follows,

$$D^2 U_j = \frac{1}{h^2}\left(U_{j-1} - 2U_j + U_{j+1}\right).$$

By substitution we get the following set of equations,

$$\frac{1}{h^2}\left(U_{j-1} - 2U_j + U_{j+1}\right) + qU_j = f(x_j) \qquad \text{for } j = 1, 2, \dots, m.$$

Written out with matrix notation we get the following,

$$A = \frac{1}{h^2}\begin{bmatrix} (qh^2-2) & 1 & & & & \\ 1 & (qh^2-2) & 1 & & & \\ & 1 & (qh^2-2) & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & (qh^2-2) & 1 \\ & & & & 1 & (qh^2-2) \end{bmatrix}, \qquad F = \begin{bmatrix} f(x_1) - \alpha/h^2 \\ f(x_2) \\ f(x_3) \\ \vdots \\ f(x_{m-1}) \\ f(x_m) - \beta/h^2 \end{bmatrix}.$$

Problem P11: Continuing along the lines of $P10$, setup a finite difference method for the most general linear, second-order Dirichlet boundary value problems:

$$u''(x) + p(x)u'(x) + q(x)u(x) = f(x), \qquad u(x_L) = \alpha, \qquad u(x_R) = \beta.$$

Compared to $P10$, now $p(x)$, $q(x)$ are arbitrary function like $f(x)$. Use approximation(1.3), namely the centered finite difference D_0, for the u' term. State A and F in the linear system $AU = F$.

Solution:
First note that our grid scheme is the same as $P10$ with $h = (x_R - x_L)/(m-1)$ for spacing and $x_j = jh + x_L$ for each point. Recall the centered finite difference scheme approximation for u',

$$D_0 U_j = \frac{1}{2h}\left(U_{j+1} - U_{j-1}\right).$$

Using the same finite difference approximation for D^2 as in the last problem, by substitution we get the following set of equations,

$$\frac{1}{h^2}\left(U_{j-1} - 2U_j + U_{j+1}\right) + p(x_j)\frac{1}{2h}\left(U_{j+1} - U_{j-1}\right) + q(x_j)U_j = f(x_j), \quad \text{for } j = 1, 2, \dots, m.$$

Through some algebra and combining like terms we get,

$$\begin{aligned}
\frac{1}{h^2}\left(U_{j-1} - 2U_j + U_{j+1}\right) + p(x_j)\frac{1}{2h}\left(U_{j+1} - U_{j-1}\right) + q(x_j)U_j &= f(x_j) \\
\frac{1}{h^2}\left(U_{j-1} - 2U_j + U_{j+1} + \frac{p(x_j)h}{2}\left(U_{j+1} - U_{j-1}\right) + q(x_j)h^2U_j\right) &= f(x_j) \\
\frac{1}{h^2}\left(\left(1 - \frac{p(x_j)h}{2}\right)U_{j-1} + \left(q(x_j)h^2 - 2\right)U_j + \left(1 + \frac{p(x_j)h}{2}\right)U_{j+1}\right) &= f(x_j).
\end{aligned}$$

Written out with matrix notation we get,

$$A = \frac{1}{h^2}\begin{bmatrix}
(q(x_1)h^2 - 2) & \left(1 + \frac{p(x_1)h}{2}\right) & & & \\
\left(1 - \frac{p(x_2)h}{2}\right) & (q(x_2)h^2 - 2) & \left(1 + \frac{p(x_2)h}{2}\right) & & \\
 & \ddots & \ddots & \ddots & \\
 & & \left(1 - \frac{p(x_{m-1})h}{2}\right) & (q(x_{m-1})h^2 - 2) & \left(1 + \frac{p(x_{m-1})h}{2}\right) \\
 & & & \left(1 - \frac{p(x_m)h}{2}\right) & (q(x_m)h^2 - 2)
\end{bmatrix}$$

$$F = \begin{bmatrix}
f(x_1) - \frac{1}{h^2}\left(1 - \frac{p(x_L)h}{2}\right)\alpha \\
f(x_2) \\
f(x_3) \\
\vdots \\
f(x_{m-1}) \\
f(x_m) - \frac{1}{h^2}\left(1 + \frac{p(x_R)h}{2}\right)\beta
\end{bmatrix}.$$