MATH 232 Homework 1

1. **Problem 1**

SOLUTION:

(a) Given two data points, we can derive a first order polynomial of the form:

$$y(x) = a_1 x + a_0.$$

Next, we can use the Lagrange interpolation equation to construct our $y(x)$ given the data points $(\overline{x}, u(\overline{x}))$ and $(\overline{x}+h, u(\overline{x}+h))$:

$$y(x) = u(\overline{x})\frac{(x-\overline{x}-h)}{(\overline{x}-\overline{x}-h)} + u(\overline{x}+h)\frac{(x-\overline{x})}{(\overline{x}+h-\overline{x})}$$

.
We can simplify the equation above to:

$$y(x) = \frac{u(\overline{x}+h)(x-\overline{x}) - u(\overline{x})(x-\overline{x}-h)}{h}$$

Now, we take the derivative of $y(x)$:

$$y'(x) = \frac{u(\overline{x}+h) - u(\overline{x})}{h}$$

We can see that our derivative, $y'(x)$, corresponds exactly to the first order forward difference, $D_+u(x)$.

(b) Again, we have two data points. So, we derive a first order polynomial of the form:

$$y(x) = a_1 x + a_0.$$

Again, we use the Lagrange interpolation equation to construct our $y(x)$ given the data points $(\overline{x}-h, u(\overline{x}-h))$ and $(\overline{x}, u(\overline{x}))$:

$$y(x) = u(\overline{x}-h)\frac{(x-\overline{x})}{(\overline{x}-h-\overline{x})} + u(\overline{x})\frac{(x-\overline{x}+h)}{(\overline{x}-\overline{x}+h)}$$

.
We can simplify the equation above to:

$$y(x) = \frac{u(\overline{x})(x-\overline{x}+h) - u(\overline{x}-h)(x-\overline{x})}{h}$$

Now, we take the derivative of $y(x)$:

$$y'(x) = \frac{u(\overline{x}) - u(\overline{x}-h)}{h}$$

We can see that our derivative, $y'(x)$, corresponds exactly to the first order backwards difference, $D_-u(x)$.

(c) Now that we have three data points, we can increase the order of our polynomial to the form

$$y(x) = a_2 x^2 + a_1 x + a_0.$$

Again, we use the Lagrange interpolation equation to construct our $y(x)$ given the data points $(\overline{x} - h, u(\overline{x} - h))$, $(\overline{x}, u(\overline{x}))$, and $(\overline{x} + h, u(\overline{x} + h))$:

$$\begin{aligned} y(x) = u(\overline{x} - h) &\frac{(x - \overline{x})(x - \overline{x} - h)}{(\overline{x} - h - \overline{x})(\overline{x} - \overline{x} - 2h)} \\ +u(\overline{x}) &\frac{(x - \overline{x} + h)(x - \overline{x} - h)}{(\overline{x} - \overline{x} + h)(\overline{x} - \overline{x} - h)} \\ +u(\overline{x} + h) &\frac{(x - \overline{x} + h)(x - \overline{x})}{(\overline{x} - \overline{x} + 2h)(\overline{x} - \overline{x} + h)}. \end{aligned}$$

We can simplify the equation above to:

$$\begin{aligned} y(x) = &\frac{1}{2h^2}\left(u(\overline{x} - h)(x - \overline{x})(x - \overline{x} + h)\right) \\ &-\frac{1}{h^2}\left(u(\overline{x})(x - \overline{x} + h)(x - \overline{x} - h)\right) \\ &+\frac{1}{2h^2}\left(+u(\overline{x} + h)(x - \overline{x} + h)(x - \overline{x})\right). \end{aligned}$$

Now, we take the derivative of $y(x)$:

$$\begin{aligned} y'(x) = &\frac{1}{2h^2}\left(2u(\overline{x} - h)(x - \overline{x} + h)\right) \\ &-\frac{1}{h^2}\left(2u(\overline{x})(x - \overline{x} + h)\right) \\ &+\frac{1}{2h^2}\left(+2u(\overline{x} + h)(x - \overline{x} + h)\right). \end{aligned}$$

Which simplifies to:

$$y'(x) = \frac{u(\overline{x} + h) - u(\overline{x} - h)}{2h}$$

We can see that our derivative, $y'(x)$, corresponds exactly to the first order centered difference, $D_0 u(x)$.

Discussion:

When using this alternative derivation for the finite difference formulas, we are missing the terms that provide the local truncation error. Our method recovers the correct coefficients, but with this method we cannot (alone) determine how accurate our solution is.

2. **Problem 2**

SOLUTION:

(a) Given the equation $u''(x) = c_{-2}u(x-2h) + c_{-1}u(x-h) + c_0u(x) + c_1u(x+h) + c_2u(x+2h)$, we see we have a system of 5 unknowns to solve for. This means we must Taylor expand each term to include 5 terms in order to set up our 5 x 5 Vandermonde system. By doing this we obtain:

$$
\begin{aligned}
&c_{-2}[u(x) - 2hu'(x) + \tfrac{(2h)^2}{2!}u''(x) - \tfrac{(2h)^3}{3!}u'''(x) + \tfrac{(2h)^4}{4!}u''''(x) + O(h^5)] \\
&+c_{-1}[u(x) - hu'(x) + \tfrac{(h)^2}{2!}u''(x) - \tfrac{(h)^3}{3!}u'''(x) + \tfrac{(h)^4}{4!}u''''(x) + O(h^5)] \\
&+c_0u(x) \\
&+c_1[u(x) + hu'(x) + \tfrac{(h)^2}{2!}u''(x) + \tfrac{(h)^3}{3!}u'''(x) + \tfrac{(h)^4}{4!}u''''(x) + O(h^5)] \\
&+c_2[u(x) + 2hu'(x) + \tfrac{(2h)^2}{2!}u''(x) + \tfrac{(2h)^3}{3!}u'''(x) + \tfrac{(2h)^4}{4!}u''''(x) + O(h^5)]
\end{aligned}
$$

Since we want to determine $h^2u''(x)$, we want to weight the coefficients such that the $O(1), O(h), O(h^3)$, and $O(h^4)$ terms vanish. Using the system of linear equations above, we then can form a system $Ax = b$, where A corresponds to the coefficients of each term, x is the vector of c weights, and b is our target vector to obtain the 2nd derivative.
Concretely, this means we solve the system $Ax = b$ with:

$$
\text{A} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ -2 & -1 & 0 & 1 & 2 \\ 2 & \frac{1}{2} & 0 & \frac{1}{2} & 2 \\ \frac{-8}{6} & \frac{-1}{6} & 0 & \frac{1}{6} & \frac{8}{6} \\ \frac{16}{24} & \frac{1}{24} & 0 & \frac{1}{24} & \frac{16}{24} \end{bmatrix}, \text{ b} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \texttt{x} = \begin{bmatrix} c_{-2} \\ c_{-1} \\ c_0 \\ c_1 \\ c_2 \end{bmatrix}
$$

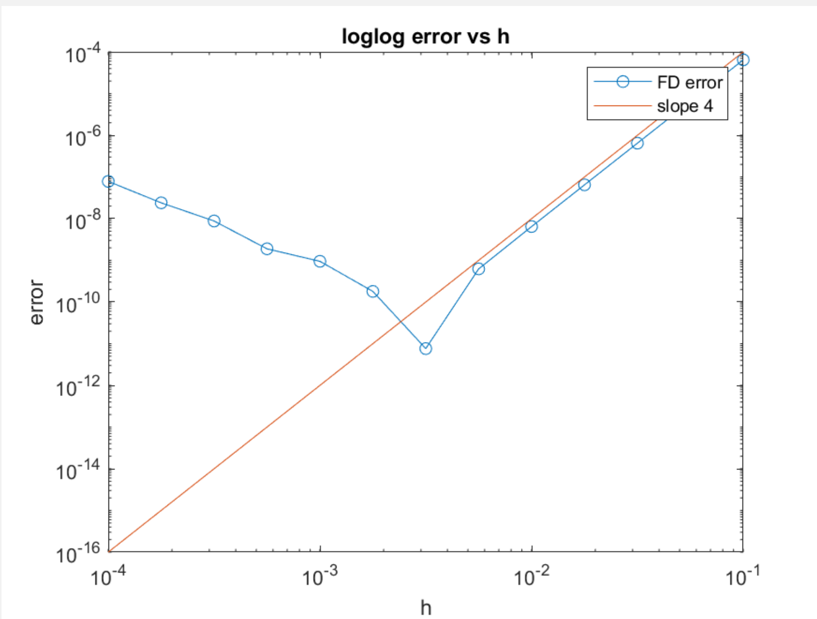
Using `MATLAB`'s `Ax = b` solver then yields:
$c_{-2} = -0.0833333$
$c_{-1} = 1.3333333$
$c_0 = -2.5$
$c_1 = 1.3333333$
$c_2 = -0.0833333$

(b) Using the script `fdstencil.m` provided to us, we use parameter `k` $= 2$ because we want the 2nd derivative of $u(x)$ and `j` $= [-2, -1, 0, 1, 2]$ as our indices of grid points. We find that we obtain the weights:
$c_{-2} = -0.0833333$
$c_{-1} = 1.3333333$
$c_0 = -2.5$
$c_1 = 1.3333333$
$c_2 = -0.0833333$
which exactly match our calculation from (a).

(c) Using the script `q2c.m` attached in the appendix, we test this finite difference formula to approximate $u''(1)$ for $u(x) = sin(2x)$ with values of h from `hvals = logspace(-1-4,13)`. We tabulate the results of the computed error for each value of h as well as the predicted error from the leading term of the expression from `fdstencil.m`, which is $-0.0111111h^4u^{(6)}(x)$, where $u^{(6)}$ is the 6th derivative of our function $u(x)$. Below is the table of errors:

h	err	pred err
1.0000e-01	6.4431e-05	6.4661e-05
5.6234e-02	6.4588e-06	6.4661e-06
3.1623e-02	6.4638e-07	6.4661e-07
1.7783e-02	6.4651e-08	6.4661e-08
1.0000e-02	6.4564e-09	6.4661e-09
5.6234e-03	6.2349e-10	6.4661e-10
3.1623e-03	7.6308e-12	6.4661e-11
1.7783e-03	1.7961e-10	6.4661e-12
1.0000e-03	9.4716e-10	6.4661e-13
5.6234e-04	1.8868e-09	6.4661e-14
3.1623e-04	8.7742e-09	6.4661e-15
1.7783e-04	2.3961e-08	6.4661e-16
1.0000e-04	7.8163e-08	6.4661e-17

We also produce a loglog plot of the absolute error vs h below.



We see that this is an order 4 method, hence we verify that after machine precision and round off error that we are parallel to a line with slope 4 in the loglog plot. From the table, we see that for larger values of h, our predicted error is very close to the actual error, matching it in terms of magnitude with slight numerical differences. We see that for small values h, namely as h approaches $1e{-}3$ we see that our error approaches machine precision, resulting in our computed error to deviate from the expected linear trend in the loglog plot.

Contributions:

Each group member completed the assignment individually, then the team we met together to discuss the answers complete the write up in Latex.

APPENDIX

q2c.m

```
hvals = logspace(-1, -4, 13);
u = @(x) sin(2*x);
exact = -4*sin(2);
errs = [];
disp(' ')
disp('     h                 err      pred err')
for i=1:length(hvals)
    approx = (-8.333333333333333e-02*u(1-2*hvals(i)) +
        1.333333333333333e+00*u(1-hvals(i)) + -2.500000000000000e+00*u(1) +
        1.333333333333333e+00 * u(1+hvals(i)) + -8.333333333333333e-02 *
        u(1+2*hvals(i)))/(hvals(i)^2);
    err = abs(approx-exact);
    errs = [errs err];
    fprintf('%13.4e %13.4e  %13.4e\n', hvals(i) ,err, -0.0111111 *
        hvals(i)^4*-64*sin(2))
end

loglog(hvals, errs,'-o')
hold on
sl4 = @(x) x.^(4);
ys4 = sl4(hvals);
loglog(hvals, ys4);
xlabel('h')
ylabel('error')
title('loglog error vs h')
legend('FD error', 'slope 4')
saveas(gcf,'2c.png')
```