Midterm Exam 2
Instructor: Prof. Raja Sengupta

April 17, 2019
25 questions, 25 points, 50 minutes
14 pages

Name: ____________________________

Student ID: ____________________________

Statement of Academic Integrity
UC Berkeley Honor Code: "As a member of the UC Berkeley community, I act with honesty, integrity, and respect for others."
On my honor, I will neither give nor receive any assistance in taking this exam. I will not use any program other than MATLAB on my computer and have turned off all of my internet connections.

Signed: ____

Instructions

1. Write your full name and SID in the blanks above and on the top of the bubble sheet.
2. Read and sign the above statement of academic integrity.
3. Bring your Cal ID to the exam room.
4. Mark your answers on the bubble sheet with pen or pencil. There is one and only one correct choice for each question. Multiple bubbles, incomplete bubbles, or stray marks will be marked incorrect.
5. You may bring notes for this exam provided they are bundled together.
6. Only one laptop is allowed per student and only MATLAB should be running on that computer. The use of any other program or electronic device during the exam constitutes cheating and subjects the offender to immediate dismissal from the exam.
7. Please do not get up to leave until the exam is over.
8. You will be photographed during the exam for the record.
9. At the end of the exam, hand in the completed bubble sheet AND the exam to your primary section GSI.

Do not open the exam book until instructed to do so.

1. Suppose we have a set of data points:

$$x_{data} = \begin{bmatrix} x_4 & x_3 & x_2 & x_1 \end{bmatrix}^T$$

$$y_{data} = \begin{bmatrix} y_4 & y_3 & y_2 & y_1 \end{bmatrix}^T$$

to which we would like to fit a curve represented by a cubic polynomial:

$$y = a_3 x^3 + a_2 x^2 + a_1 x + a_0$$

Given the data and model above, we may solve for the polynomial coefficients:

$$a = \begin{bmatrix} a_3 & a_2 & a_1 & a_0 \end{bmatrix}^T = (X^T X)^{-1} X^T y_{data}$$

What is X?

(a)

$$X = \begin{bmatrix} 1 & x_1 & x_1^2 & x_1^3 \\ 1 & x_2 & x_2^2 & x_2^3 \\ 1 & x_3 & x_3^2 & x_3^3 \\ 1 & x_4 & x_4^2 & x_4^3 \end{bmatrix}$$

(b)

$$X = \begin{bmatrix} 1 & x_4 & x_4^2 & x_4^3 \\ 1 & x_3 & x_3^2 & x_3^3 \\ 1 & x_2 & x_2^2 & x_2^3 \\ 1 & x_1 & x_1^2 & x_1^3 \end{bmatrix}$$

(c)

$$X = \begin{bmatrix} x_1^3 & x_1^2 & x_1 & 1 \\ x_2^3 & x_2^2 & x_2 & 1 \\ x_3^3 & x_3^2 & x_3 & 1 \\ x_4^3 & x_4^2 & x_4 & 1 \end{bmatrix}$$

(d)

$$X = \begin{bmatrix} x_4^3 & x_4^2 & x_4 & 1 \\ x_3^3 & x_3^2 & x_3 & 1 \\ x_2^3 & x_2^2 & x_2 & 1 \\ x_1^3 & x_1^2 & x_1 & 1 \end{bmatrix}$$

(e)

$$X = \begin{bmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ x_1^2 & x_2^2 & x_3^2 & x_4^2 \\ x_1^3 & x_2^3 & x_3^3 & x_4^3 \end{bmatrix}$$

2. In this class we learn two methods to compute roots of a function: Bisection method and Newton-Raphson method. Which of the following is **incorrect** about the aforementioned methods?
   (a) Bisection method will always converge given appropriate inputs
   (b) Newton-Raphson method usually converges faster than Bisection method
   (c) Newton-Raphson method will always converge
   (d) Bisection method does not require the derivative of the function whose roots we are trying to find
   (e) All of the above statements are correct, concerning the bisection and Newton-Rasphson methods.

3. The floating ball in the figure below has a radius of R. You are asked to find the depth to which the ball is submerged when floating in water. The equation that gives the depth x to which the ball is submerged under water is given by $x^3 - 12x^2 + 4 = 0$.

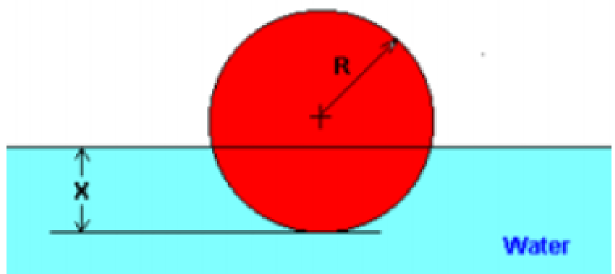


Figure 1: Floating ball problem

Use the bisection method of finding roots of equations to find the depth x to which the ball is submerged under water. First assume x is in range of [0, 10]. What is the lower and higher bound of x after computing one iteration?

   (a) 0, 10
   (b) 0, 5
   (c) 5, 10
   (d) 5, 7.5
   (e) None of the above

4. Consider the graph of a function $f(x)$ below. What are the values of the **forward**, **backward**, and **central difference** approximations of $f'(x)$ at $x = 4$, respectively? (Hint: the analytical derivative of $f(x)$ at $x = 4$ is 4).

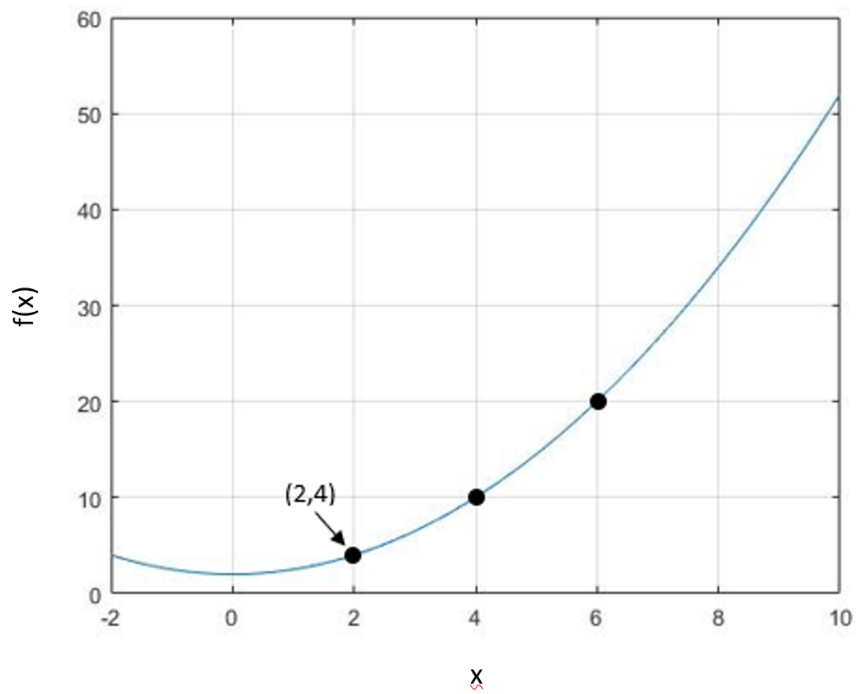


Figure 2: Graph of $f(x)$

(a) 5, -3, 4
(b) 5, 3, 4
(c) 5, 3, 8
(d) 10, 5, 8
(e) 3, 5, 4

5. Using Newton-Raphson's Method with an initial guess of $x_0 = 5$, what is the result after the second iteration to 2 decimal points?

$$f(x) = (x-2)^3$$

Hint: the equation for Newton-Raphson's Method: $x_{n+1} = x_n - f(x_n)/df(x_n)$

(a) $x_0 = 3.00$
(b) $x_0 = 3.33$
(c) $x_0 = 3.50$
(d) $x_0 = 4.00$
(e) $x_0 = 4.33$

6. Consider the differential equation

$$\frac{dy}{dx} = x + 1 \tag{1}$$

where y is a function of x. Using a step size of $h = 2$ and the initial condition $y(0) = 2$, find the value of $y(2)$ using the forward Euler's method.

   (a) 1
   (b) 2
   (c) 3
   (d) 4
   (e) 5

7. Consider the following code written by our GSI Julia, which implements a finite difference method to approximate the derivative of a function. The input `myFun` is a function handle to a function that is differentiable between the endpoints `a` and `b`, and `N` is the number of evaluation points:

```
function [der] = myFiniteDiff(myFun, a, b, N)
    x = linspace(a,b,N);
    dx = (b - a) / (N - 1);
    der = [];

    for i = 2:length(x)
        der = [der, (myFun(x(i)) - myFun(x(i-1))) / dx];
    end
end
```

The estimates of the derivative stored in `der` are an example of which of the following finite difference methods for finding the derivative at the given x values.

   (a) This could only be the backward difference for $f'(x)$ at $x = \{a, a + \Delta x, ..., b - \Delta x, b\}$
   (b) This could only be the backward difference for $f'(x)$ at $x = \{a + \Delta x, a + 2\Delta x, ..., b - \Delta x, b\}$
   (c) This could only be the forward difference for $f'(x)$ at $x = \{a, a + \Delta x, ..., b - 2\Delta x, b - \Delta x\}$
   (d) This could only be the central difference for $f'(x)$ at $x = \{a + \frac{1}{2}\Delta x, a + \frac{3}{2}\Delta x, ..., b - \frac{3}{2}\Delta x, b - \frac{1}{2}\Delta x\}$
   (e) This could be any of the options described in (b), (c), or (d)

8. Shelley is taking E7 at UC Berkeley, and Professor Sengupta has asked her to write a function that interpolates an arbitrary set of data using Lagrange Polynomials. Shelley wrote the function below:

```
function [yy, p] = myLagrange(x, y, xx)
    M = length(x);
    for i = 1 : length(x)
        if i == 1
            powers(i) = M - 1;
        else if i == M
            powers(i) = 0;
        else
            powers(i) = i - 1;
        end
    end
```

```
    A = x' .^ powers
    p = (A \ y');
    yy = (xx' .^ powers) * p;
    p = p';
    yy = yy';
end
```

Prof. Sengupta gave her a set of data: `x = [1 2 3 4 5 6]`, `y = [10 9 8 7 6 5]`, `xx = [20 30]`. After running the function and storing both outputs, which term of the Lagrange polynomial does `p(3)` correspond with?

(a) Constant

(b) Linear

(c) Quadratic

(d) Cubic

(e) Quartic

9. You run various program scripts with a sufficiently large number of various inputs of sufficiently varied size. You plot their execution times below. Match the execution time curves, A, B, C, D, E, F and G, with their corresponding functions' big-O time complexity, which can be: $O(1)$, $O(n^n)$, $O(\log n)$, $O(n)$, $O(\sqrt{n})$, $O(n^3)$ and $O(n^2)$

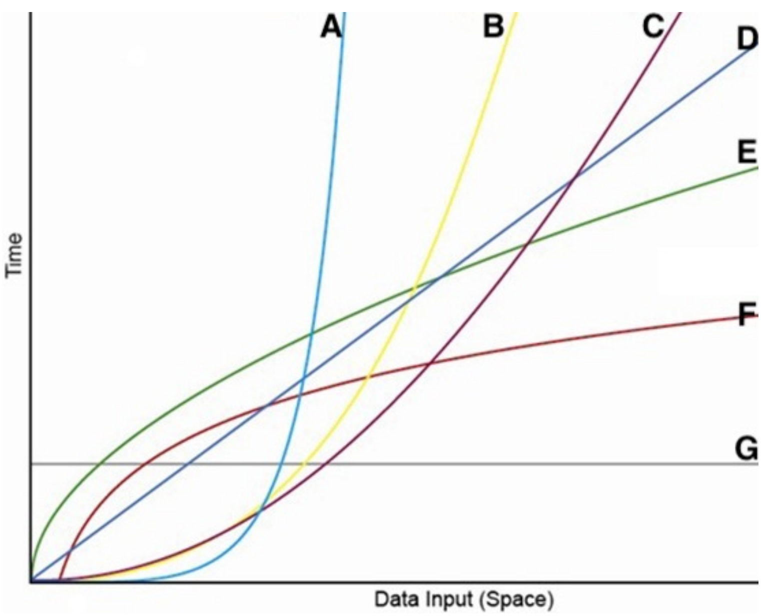


Figure 3: Graph of $f(x)$

(a) G is $O(1)$, A is $O(n^n)$, F is $O(\log n)$, D is $O(n)$, E is $O(\sqrt{n})$, B is $O(n^3)$, and C is $O(n^2)$

(b) G is $O(1)$, B is $O(n^n)$, F is $O(\log n)$, D is $O(n)$, E is $O(\sqrt{n})$, C is $O(n^3)$, and A is $O(n^2)$

(c) G is $O(1)$, C is $O(n^n)$, E is $O(\log n)$, D is $O(n)$, F is $O(\sqrt{n})$, B is $O(n^3)$, and A is $O(n^2)$

(d) G is $O(1)$, A is $O(n^n)$, E is $O(\log n)$, D is $O(n)$, F is $O(\sqrt{n})$, B is $O(n^3)$, and C is $O(n^2)$

(e) None of the above is correct

10. The time derivative of location is velocity, the time derivative of velocity is acceleration, the time derivative of acceleration is jerk, the time derivative of jerk is snap, the time derivative of snap is crackle, and the time derivative of crackle is pop. Given `N` data points of the location history of an object (in other words, the location of an object at various points in time), you are to use the forward difference method for all derivative approximations. How many values of pop will you obtain?

    (a) `2N`
    (b) `N + 5`
    (c) `N`
    (d) `N - 5`
    (e) `N - 6`

11. Which of the following lines of code will **not** correctly solve a linear system `A * x = b` for `x`, assuming `A` is an n-by-n matrix and a unique solution does exist?

    (a) `x = A \ b`
    (b) `x = b / A`
    (c) `x = inv(A) * b`
    (d) `x = pinv(A) * b`
    (e) `x = A^(-1) * b`

12. A common mathematical operation implemented in computer programming is matrix multiplication; multiplying two matrices of compatible dimensions. For instance, we can compute a matrix $C = A*B$ using the following function where all matrices A, B, C are of size n-by-n. What is the computational complexity of the function?

```
function C = MatrixMultiplication(A, B)
    C = zeros(n);
    for i = 1 : n
        for j = 1 : n
            for k = 1 : n
                C(i, j) = C(i, j) + A(i, k) * B(k, j);
            end
        end
    end
end
```

    (a) $O(n^3)$
    (b) $O(n^2)$
    (c) $O(n)$
    (d) $O(n \log n)$
    (e) $O(\log n)$

13. Assume that you have already converted the linear system of equations below into the matrix form $Ax = b$. Which of the following would produce the solution x?

$$3x_1 + 2x_2 = 3$$
$$6x_1 - 4x_2 = -6$$

(a) `pinv(A)*b` only
(b) `A\b` only
(c) `inv(A)*b` only
(d) Both a and b, but not c
(e) All a, b and c

14. Out of the following choices below, which will correctly return the roots of $\Psi(x)$ when typed in MATLAB's command window, given that

$$\Psi(x) + 21x - 15 + 18x^2 = -2\Psi(x) + 60x^8 + 75x^6 + 9x^5 - 261x^4$$

Hint: What is the correct line of code which will return all values of x that satisfy $\Psi(x)$=0?

(a) `roots([60, 75, 9, -261, -18, -21, 15])`
(b) `roots([20, 0, 25, 3, -87, 0, -6, -7, 5])`
(c) `roots([-15, -21, -18, -261, 9, 75, 60])`
(d) `roots([5, -7, -6, 0, -87, 3, 25, 0, 20])`
(e) `roots([-60, -75, -9, 261, 18, 21, -15])`

15. Which of the following $f(x)$ is nonlinear with respect to x for all $a > 0$?

(a) $x + a^2$
(b) a^3x
(c) ax^2
(d) πx
(e) None of the above

16. Consider the following (poorly-named) MATLAB function and output from the command line, where the variables `data_x` and `data_y` contain (x, y) data from an experiment and are already defined in the workspace:

```
function [a,b,c] = hmm(x,y,flag)

    if flag
        A = [ zeros(length(x),1), x'];
    else
        A = [ ones(length(x),1), x'];
    end

    tmp = A \ y';

    a = tmp(1);
    b = tmp(2);

    c = sum( (A * tmp - y').^2 ) / length(y);
end
```

```
>> size(data_x)
ans =
     1     5

>> size(data_y)
ans =
     1     5

>> [a,b,c] = hmm(data_x,data_y,0)
a =
    6.8853
b =
    3.7786
c =
    0.4939
```

Which of the following statements is correct based on the information above?

(a) A model with the form $y = cx$ fitted to `[data_x,data_y]` has a mean-squared-error (MSE) of ≈ 6.9

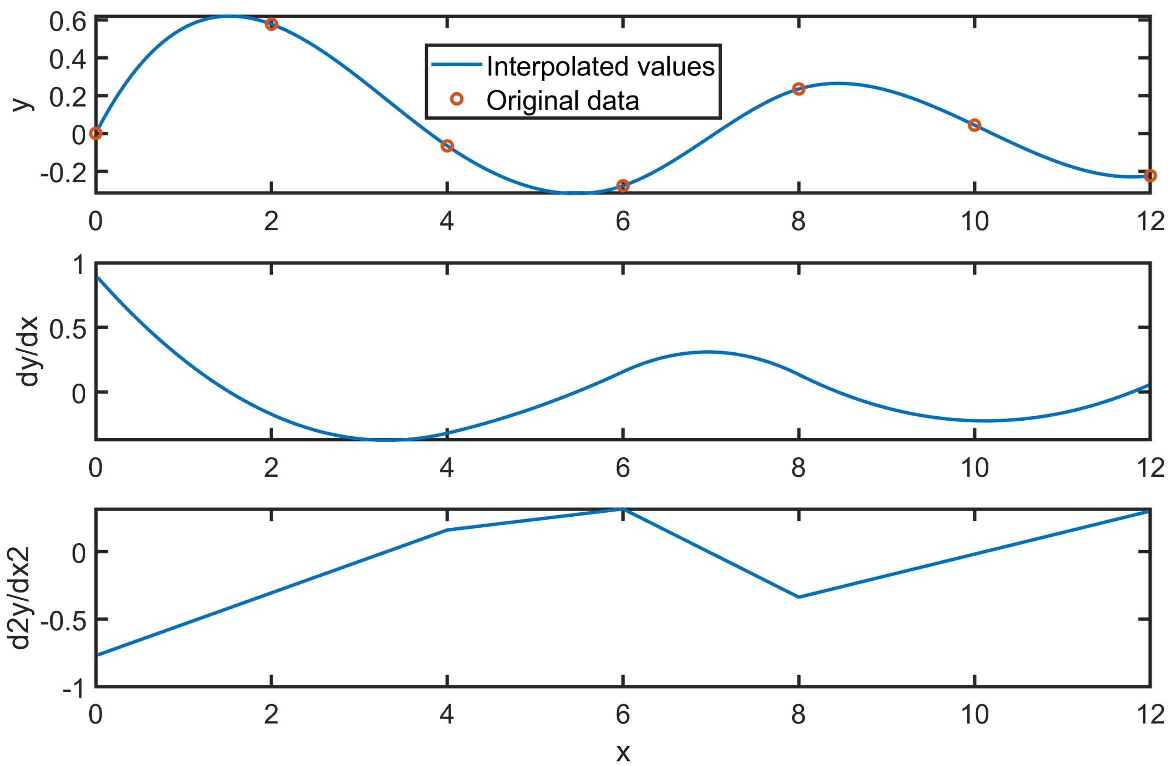
(b) A model with the form $y = cx$ fitted to `[data_x,data_y]` has a MSE of ≈ 0.49

(c) A model with the form $y = cx$ fitted to `[data_x,data_y]` has a slope of ≈ 3.8

(d) A model with the form $y = cx$ fitted to `[data_x,data_y]` has a slope of ≈ 6.9

(e) A model with the form $y = c_1x + c_2$ fitted to `[data_x,data_y]` has a slope of ≈ 3.8

17. You perform interpolation on a set of 7 data points and plot the interpolated curve, along with the first and second derivative of the interpolated values, as shown below:



Given these curves, which of the following could be the interpolation method used?

(a) Linear interpolation

(b) Spline interpolation

(c) Lagrange interpolation

(d) Both (b) and (c)

(e) None of these interpolation methods would produce these results

18. Professor Sengupta asked Maribel to model function $y = a_1 + a_2 \log(x)$. You collected n pairs of data points $\{x_1, y_1\}, \{x_2, y_2\}...\{x_n, y_n\}$. You seek to fit a linear model to this data using least squares. The model is defined as such:

$$\theta = (X^T X)^{-1} X^T y$$

where $y = [y_1, y_2, ..., y_n]^T$. You need to help her formulate the X matrix in the above equation such that the parameters can be solved for in the form $\theta = \begin{bmatrix} a_1 & a_2 \end{bmatrix}^T$. What is the correct X matrix formulation?

(a)
$$X = \begin{bmatrix} 1 & \log x_1 \\ 1 & \log x_2 \\ \vdots & \vdots \\ 1 & \log x_n \end{bmatrix}$$

(b)
$$X = \begin{bmatrix} 1 & e^{x_1} \\ 1 & e^{x_2} \\ \vdots & \vdots \\ 1 & e^{x_n} \end{bmatrix}$$

(c)
$$X = \begin{bmatrix} 1 & \log x_1 \\ 1 & \log x_2 \\ \vdots & \vdots \\ 1 & \log x_n \end{bmatrix}^T$$

(d)
$$X = \begin{bmatrix} e^{x_1} & 1 \\ e^{x_2} & 1 \\ \vdots & \vdots \\ e^{x_n} & 1 \end{bmatrix}$$

(e)
$$X = \begin{bmatrix} 1 & e^{x_1} \\ 1 & e^{x_2} \\ \vdots & \vdots \\ 1 & e^{x_n} \end{bmatrix}^T$$

19. Consider the following system of linear equations:

$$\begin{aligned} 3a - 2b + 2c &= 0 \\ 2a - 1b + 5c &= 1 \\ -c + 6a &= 1 \\ a + b &= 4, \end{aligned}$$

where the solution `x` is defined as:

$$\mathrm{x} = \begin{bmatrix} a \\ b \\ c \end{bmatrix}.$$

Which of the following definitions of `A` and `x` will give a least-squares best-fit solution to the above system using MATLAB?

(a) `A = [3 -2 2; 2 -1 5; 6 0 -1; 1 1 0];`
`x = pinv(A)*[0 1 1 4];`

(b) `A = [3 -2 2; 2 -1 5; 6 0 -1; 1 1 0];`
`x = pinv(A)*[0;1;1;4];`

(c) `A = [3 -2 2; 2 -1 5; 6 0 -1; 1 1 0];`
`x = inv(A)*[0;1;1;4];`

(d) `A = [3 -2 2; 2 -1 5; -1 6; 1 1];`
`x = pinv(A)*[0;1;1;4];`

(e) `A = [3 -2 2; 2 -1 5; -1 6; 1 1];`
`x = inv(A)*[0;1;1;4];`

20. A unique polynomial is generated to pass through all $n + 1$ data points. What is the degree of this polynomial function?

(a) exactly n

(b) exactly $n + 1$

(c) $n + 1$ or less

(d) n or less

(e) exactly $n + 2$

21. In this class we have coded multiple ways to numerically calculate the integral of a function. Which of the following functions `INT` will result in the most accurate approximation to the exact definite integral $\int_a^b f(x)dx$?

Assume the inputs to be:

```
f = @(x) x^2
interval = [0 4]
N = 4
```

(a)

```
function integral = INT(f, interval, N)
    dx = (interval(2) - interval(1)) / N;
    x = interval(1) + 0:dx:interval(2) - dx;
    integral = sum(f(x)) * dx;
end
```

(b)

```
function integral = INT(f, interval, N)
    dx = (interval(2) - interval(1)) / N;
    x = interval(1) + dx:dx:interval(2);
    integral = sum(f(x)) * dx;
end
```

(c)

```
function integral = INT(f, interval, N)
    dx = (interval(2) - interval(1)) / N;
     x = interval(1) + dx/2:dx:interval(2) - dx/2;
    integral = sum(f(x)) * dx;
end
```

(d)

```
function integral = INT(f, interval, N)
    dx = (interval(2) - interval(1)) / N;
    x= interval(1):dx:interval(2);
    integral = trapz(x,f(x));
end
```

(e) (a) and (b) have the same accuracy

22. Lagrange interpolation is one of the simplest and commonly used interpolation methods. Which of the following is **incorrect** about Lagrange interpolation?

    (a) For `N` number of data points, the interpolation will have the largest polynomial of degree `N-1`
    (b) This method is good for smooth data with small `N`
    (c) This method is good for data with sharp jumps and large `N`
    (d) Currently, there is no built-in MATLAB function to interpolate using this method
    (e) All of the above statements are correct about Lagrange interpolation.

23. Consider the function $f(x) = 2x^3 - 9x^2 + 7$. Which of the following numerical integration methods will estimate $\int_a^b f(x)dx$ the most accurately, where the accuracy is defined as how close the approximation is to the analytical solution. You may assume that you perform each method with the same step size, and that a and b are real numbers.

    (a) Trapezoidal rule
    (b) Simpson's rule
    (c) Right-Riemann Sum
    (d) Midpoint-Riemann Sum
    (e) Left-Riemann Sum

24. The code below is computing the derivative of a function $f(x)$ (using MATLAB function handle) at a specific x with step size h. What is the order of accuracy of this method?

```
function derivative = myDerivative[f, x, h]
    f_neg = f(x - h);
    f_pos = f(x + h);
    derivative = (f_pos - f_neg) / (2 * h);
end
```

    (a) 0
    (b) 1
    (c) 2
    (d) 3
    (e) 4

25. Which of the following does **not** affect the accuracy of numerical integration methods?

    (a) Step size
    (b) Length of interval of integration
    (c) Magnitude of higher order derivatives
    (d) Methods of numerical integration
    (e) All of the above affect the accuracy of numerical integration methods.