
E7: Introduction to Computer Programming for Scientists and Engineers

University of California at Berkeley, Spring 2017

Instructor: Lucas A. J. Bastien

Final Exam

Date: Tuesday May 9, 2017

Duration: 170 minutes

Version: release

Full name: _____

Student ID: _____

Full name of primary GSI: _____

Lab section number: _____

By handing in your responses to this exam, you attest that you have neither given nor received any assistance in the taking of this exam.

Part 1: Overview

This exam is **32 pages long** (including this cover page) and contains **50 multiple-choice questions**. Please follow the instructions below:

- Fill in the required information above.
- **IMPORTANT:** On the front of your Scantron, write **very clearly**:
 - ◊ Your full name in the box labeled “NAME”
 - ◊ Your student ID number in the box labeled “SUBJECT”
 - ◊ The number 259160 in the box labeled “TEST NO.”
 - ◊ Today’s date in the box labeled “DATE”
- Bubble in your answers on the Scantron with a number 2 pencil. **There is one and only one correct answer for each question.** Multiple bubbles, incomplete bubbles, or stray marks will be marked incorrect.
- You may use one 8.5 inches by 11 inches (U.S. letter size) sheet of notes (both sides, typed or hand-written) for this exam.
- No electronic devices (including hand-held calculators) are permitted in your work area.
- There will be no questions regarding the exam during the examination time except in cases where there is a missing page or printing problem with your exam.

- At the end of the exam, hand in the completed Scantron **and** the questions (*i.e.* this document) to the instructor or to a GSI.
- **If we do not have your scantron, you will receive a score of zero for the exam. Please make sure to return your scantron at the end of the exam period.**
- When user-defined functions are mentioned in a question, and unless specified otherwise, you can assume that each of these functions is defined in an m-file that has the same name as the function, in a directory (also known as folder) where Matlab can find it (*i.e.* in a directory that is part of Matlab's path).
- Every array listed **in the multiple choices** of the questions below is of class **double** unless specified otherwise (if not of class **double**, the class of the array precedes the array, for example `logical [1, 0, 1, 1]`).

Part 2: Questions

1. What will the value of the variable “result” be after executing the following code?

```
x = 5;  
val = 2;  
result = x^2 - val*3 - 4;
```

- A) -1
- B) 23
- C) 15
- D) 27
- E) Matlab throws an error

2. Consider the following function:

```
function [output] = my_output_function(value)  
if value > 4  
    output = 100;  
end  
if value > 11  
    output = 200;  
else  
    output = 300;  
end  
if value > 9  
    output = 400;  
elseif value > 7  
    output = 500;  
end  
end
```

What is the value returned by `my_output_function(10)`?

- A) 100
- B) 200
- C) 300
- D) 400
- E) 500

3. What will the value of the variable “value” be after executing the following code?

```
value = 0;
number = 5;
if number > 10
    value = 3;
elseif number > 2
    value = 7;
else
    value = 11;
end
```

- A) 3
- B) 0
- C) 11
- D) 7
- E) Matlab throws an error

4. What will the value of the variable “answer” be after executing the following code?

```
val = 3;
answer = val > 0 | val > 10;
```

- A) logical 0 (*i.e.* false)
- B) logical 1 (*i.e.* true)
- C) logical [1, 0]
- D) logical [0, 1]
- E) Matlab throws an error

5. What will the value of the variable “cake” be after executing the following code?

```
apple = [1, 2, 3, 4];  
biscuit = 3 * ones(1, 4);  
cake = sum(apple < biscuit);
```

- A) [1, 2, 0, 0]
- B) 2
- C) [1, 1, 0, 0]
- D) [1, 2]
- E) 3

6. Consider the following function:

```
function [alpha, beta, chair] = my_short_function(delta, epsilon, fun)  
alpha = epsilon;  
beta = fun;  
chair = delta;  
end
```

What will the value of the variable “gamma” be after executing the following code?

```
delta = 1;  
echo = 2;  
fun = 3;  
[echo, fun, delta] = my_short_function(fun, delta, echo);  
gamma = [delta, echo, fun];
```

- A) [2, 1, 3]
- B) [1, 2, 3]
- C) [3, 2, 1]
- D) [3, 1, 2]
- E) Matlab throws an error

7. Consider the function handle `func` defined as `func = @(x) x`. Which of the following commands can be used to define a function handle `other_func` such that `other_func(2)` evaluates to 4?

- A) `other_func = @(x) func.^2`
- B) `other_func = @(x) func^2`
- C) `other_func = @(x) func(x)^2`
- D) All of the above
- E) None of the above

8. Consider the two following functions:

```
function [result] = my_for_function(input)
result = input;
alpha = [1, 2, 3];
for var = 1:numel(alpha)
    result = result + 1;
end
end
```

```
function [result] = my_while_function(input)
alpha = [1, 2, 3];
while alpha ~= [1, 2]
    input = input + 1;
    alpha = alpha(1:end-1);
end
result = input;
end
```

What are the values returned by `my_for_function(10)` and `my_while_function(10)`, respectively?

- A) 13; 11
- B) 11; 11
- C) 11; 13
- D) 11; Matlab throws an error
- E) 13; Matlab throws an error

9. What will the value of the variable “val” be after executing the following code?

```
val = 0;
for val = 1:5
    if val == 3 | val == 5
        continue
    end
    val = val * val;
end
```

- A) 25
- B) 3
- C) 5
- D) 9
- E) 16

10. Consider the following function:

```
function [result] = my_plus_and_minus(value_one, value_two)
result = 0;
for counter_one = 1:value_one
    for counter_two = 1:value_two
        result = result + counter_two;
    end
    result = result - counter_one;
end
end
```

What is the value returned by `my_plus_and_minus(2, 4)`?

- A) 9
- B) 8
- C) 17
- D) 2
- E) 40

11. What will the value of the variable “**result**” be after executing the following code?

```
vector_one = [];  
vector_two = zeros(2, 2);  
for index=1:4  
    vector_one(index) = index * 2;  
    vector_two(index) = index * 2;  
end  
result = vector_one .* vector_two;
```

- A) [2, 4, 6, 8]
- B) [4, 16, 36, 64]
- C) [4, 16; 36, 64]
- D) Matlab throws an error with message: Index exceeds matrix dimensions.
- E) Matlab throws an error with message: Matrix dimensions must agree.

12. What will the class and value of the variable “**result**” be, respectively, after executing the following code?

```
matrix = [2, 5, 4, 3, 1, 23];  
result = any(matrix < 3);
```

- A) double and [2, 0, 0, 0, 1, 0]
- B) double and [2, 1]
- C) logical and 1
- D) logical and [1, 0, 0, 0, 1, 0]
- E) logical and [1, 1]

13. What will the value of the variable “fourth_array” be after executing the following code?

```
first_array = [1, 2; 3, 4];
second_array = first_array .* first_array;
third_array = second_array(1,:);
fourth_array = first_array(1,:) .* third_array;
```

- A) [1, 8]
- B) [1; 27]
- C) [7, 20]
- D) [7; 45]
- E) Matlab throws an error

14. Consider the following function:

```
function [vector] = my_unknown_function(vector)
while true
    index = 1;
    keep_going = false;
    while index < numel(vector)
        if vector(index) > vector(index+1)
            vector(index) = [];
            keep_going = true;
        else
            index = index + 1;
        end
    end
    if ~keep_going
        break
    end
end
end
```

What is the value returned by `my_unknown_function([6, -8, -2, 9, 6, 10, 3, -1])`?

- A) [-8, -2, 6, -1]
- B) [-8, -2, -1]
- C) [6, -8, -2, 9, 6, 10, 3, -1]
- D) Matlab throws an error with message: Index exceeds matrix dimensions.
- E) Matlab enters an infinite loop

15. Consider the following function:

```
function [out] = my_abc_function(apple, biscuit, cake)
if apple > biscuit || apple > cake
    out = apple + biscuit;
    out = my_abc_function(biscuit, cake, apple);
else
    out = apple;
end
end
```

What is the value returned by `my_abc_function(7, 5, 3)`?

- A) 3
- B) 5
- C) 7
- D) 8
- E) 12

16. Consider the following function:

```
function [output] = my_recursive_function(value)
if value == 1
    output = 1;
elseif value == 2
    output = 2;
else
    output = my_recursive_function(value-1) + my_recursive_function(value);
end
```

What is the value returned by `my_recursive_function(3)`?

- A) 4
- B) 5
- C) 3
- D) 2
- E) The function calls itself until Matlab runs out of memory

17. What is the smallest integer that can be represented using an 8-bit two's complement binary representation?

- A) 0
- B) -127
- C) -128
- D) -255
- E) -256

18. What is the maximum amount of distinct numbers that can be represented using a given ten-bit binary representation?

- A) 20
- B) 100
- C) 512
- D) 1024
- E) An infinite number

For the next 2 questions:

Consider the following function:

```
function [array] = my_switch(vector)

if numel(vector) == 1
    array = {vector};
elseif numel(vector) == 2
    array = {vector, vector([2,1])};
else
    array = {};
    all_others = my_switch(vector(2:end));
    for index = 1:numel(all_others)
        others = all_others{index};
        array{end+1} = [vector(1), others];
        for k = 1:numel(others)
            array{end+1} = [others(1:k), vector(1), others(k+1:end)];
        end
    end
end
end
end
```

19. Assume that the input argument to this function is a non-empty row vector of class `double`. This function uses recursion to:

- A) Sort the elements of the input vector in increasing order
- B) Sort the elements of the input vector in decreasing order
- C) Reverse the order of the elements in the input vector
- D) Determine all the possible permutations (*i.e.* “ordered arrangements”) of the elements of the input vector
- E) Remove duplicate elements from the input vector

20. Consider the following function (which calls the previous function named `my_switch`):

```
function [order] = my_optimal(start, scraps)

calc_distance = @(pos_one, pos_two) sqrt(sum((pos_one-pos_two).^2));
n_scraps = size(scraps, 1);
indices = 1:n_scraps;
possibilities = my_switch(indices);
minimum_distance = Inf;

for i_possibilities = 1:numel(possibilities)

    possibility = possibilities{i_possibilities};

    distance = calc_distance(start, scraps(possibility(1),:));
    for k = 2:n_scraps
        i_one = possibility(k-1);
        i_two = possibility(k);
        distance = distance + calc_distance(scraps(i_one,:), scraps(i_two,:));
    end

    if distance < minimum_distance
        minimum_distance = distance;
        order = possibility;
    end

end

end
```

What will the value of the variable “`order`” be after executing the following code?
(hint: first determine conceptually what this function does, then draw a sketch for the specific numerical values below)

```
scraps = [3, 2; 2, 2; 5, 4; 4, 3];
order = my_optimal([0, 0], scraps);
```

- A) [2, 4, 1, 3]
- B) [4, 3, 2, 1]
- C) [2, 1, 4, 3]
- D) [1, 2, 3, 4]
- E) [1, 4, 2, 3]

21. Which of the following sorting algorithms uses a “divide and conquer” approach?

- A) Selection sort
- B) Insertion sort
- C) Quicksort
- D) Both selection sort and insertion sort
- E) None of the above

22. What is the time complexity, in big-O notation, and when n is very large, of the following function ?

```
function [result] = my_big_o_one(n)
result = 1;
counter = 1;
for index = 1:n
    result = result * counter;
    counter = counter + 1;
end
end
```

- A) $\mathcal{O}(2^n)$
- B) $\mathcal{O}(\log(n))$
- C) $\mathcal{O}(n \log(n))$
- D) $\mathcal{O}(n)$
- E) $\mathcal{O}(n^2)$

23. What is the time complexity, in big-O notation, and when m and p are very large, of the following function ?

```
function [] = my_big_o_two(m, p)
value = 1;
while value < abs(m)
    value = value * 2;
    for index = 1:p
        if index == 10
            break
        end
    end
end
end
end
```

- A) $\mathcal{O}(p \log(m))$
- B) $\mathcal{O}(\log(m))$
- C) $\mathcal{O}(p)$
- D) $\mathcal{O}(\log(p))$
- E) $\mathcal{O}(m)$

24. A mover is moving chairs and tables from one location to another using their truck. The mover makes three trips to transport everything, and stops at a weighing station during each of these trips, to weigh the loaded truck (without people in it). During the first trip, they transport 5 tables and 10 chairs, and the loaded truck weighs 2600 pounds. During the second trip, they transport 0 table and 50 chairs, and the loaded truck weighs 2500 pounds. During the third trip, they transport 2 tables and 25 chairs, and the loaded truck weighs 2450 pounds. All the tables are identical and all the chairs are identical. To determine the weight of the empty truck, of an individual table, and of an individual chair, we write the following incomplete code. Which of the following lines of code correctly fills in the missing line?

```
alpha = [1, 5, 10; ...
1, 2, 25; ...
1, 0, 50];
% MISSING LINE
weights = alpha \ beta;
```

- A) `beta = [2500; 2450; 2600];`
- B) `beta = [2450; 2600; 2500];`
- C) `beta = [2500; 2600; 2450];`
- D) `beta = [2600; 2450; 2500];`
- E) `beta = [2600; 2500; 2450];`

25. Consider a system of m linear algebraic equations with p unknowns such that $m > p$. Assume that this system of equations has zero solutions. In Matlab, we calculate the quantity `pinv(matrix)*rhs` where `matrix` is an $m \times p$ array of class `double` that represents the matrix of the system, and `rhs` is an $m \times 1$ vector of class `double` that represents the right-hand side of the system. Which of the following statements is **TRUE**?

- A) Matlab will throw an error when calculating `pinv(matrix)` because $m > p$
- B) `pinv(matrix)*rhs` will be a $p \times 1$ vector of class `double` that represents the values of the unknowns that minimize the square error
- C) `rank([matrix, rhs]) == rank(matrix)+1` evaluates to `true`
- D) Both A and C are true
- E) Both B and C are true

26. Which of the following statements is **TRUE**?

- A) The bisection method applied to the interval $[a, b]$ finds all the roots of the function f within this interval if the condition $f(a) \times f(b) \leq 0$ is satisfied
- B) Newton-Raphson's method requires the function of interest to be continuous while the bisection method does not require the function of interest to be continuous
- C) Newton-Raphson's method always converges to a root faster than the bisection method for any initial guess (Newton Raphson) and any initial interval (bisection)
- D) Newton-Raphson's method may yield a root that is not the closest root to the initial guess
- E) All of the above statements are false

27. Consider the following function which implements the bisection root-finding method:

```
function [root] = my_bisection(f, left, right, tolerance)
% Find root of function f in interval [left, right] using bisection
root = (left+right) / 2;
while abs(f(root)) > tolerance
    if f(left)*f(right) <= 0
        % MISSING CODE 1
    else
        % MISSING CODE 2
    end
    root = (left+right) / 2;
end
end
```

This function is missing code. Which of the following pieces of code can be used (in the two locations indicated by the comments in the function above, respectively) to properly finish implementing this function?

- A) left = root;
right = root;
- B) left = root;
root = right;
- C) root = left;
root = right;
- D) root = right;
root = left;
- E) right = root;
left = root;

28. Consider the function f defined by $f(x) = (x + 1)^2 - 3$ for all $x \in \mathbb{R}$. What is the value of the root of f as approximated after two iterations of the Newton-Raphson method starting from the initial guess $x = 0$?

- A) 0.75
- B) $\sqrt{3} - 1$
- C) $-\sqrt{3} - 1$
- D) 1
- E) 0

29. We are attempting to identify a mystery real-valued function f given roots approximated using the Newton-Raphson root-finding method. We apply the Newton-Raphson root-finding method several times, each time starting from a different initial guess. In each case, enough iterations of the Newton-Raphson method are used such that the approximate root will essentially be equal to the exact root (unless the approximate root is **NaN**).

Using an initial guess of -1 , the Newton-Raphson method yields a root of -2

Using an initial guess of 0 , the Newton-Raphson method yields a root of **NaN**

Using an initial guess of 3 , the Newton-Raphson method yields a root of 2

What is a possible expression for the function f ?

- A) $f(x) = x^3 - 8$
- B) $f(x) = x - 1$
- C) $f(x) = -x^2 + 4$
- D) $f(x) = 2x + 4$
- E) $f(x) = 3/x + 1$

30. Consider the polynomial P defined for all $x \in \mathbb{R}$ by:

$$P(x) = x^2 - 3x + 2$$

Toward which root of this polynomial will the bisection method converge if applied on the interval $[-2, 1.5]$?

- A) -1
- B) -2
- C) 1
- D) 0
- E) 2

31. Consider the following function:

```
function [coefficients] = my_regression(x_data, y_data)
matrix = [ones(size(x_data)), x_data];
coefficients = pinv(matrix) * y_data;
end
```

What will the value of the variable “coefficients” be after executing the following code?

```
x_data = [-1; 1; 3; 4; 5];
y_data = [3.5; 5; 5.1; 7; 8];
coefficients = my_regression(x_data, y_data)
```

Note: values below have been rounded to four digits after the decimal point.

- A) [4.0483; 0.6966]
- B) [-5.1789; -3.9721]
- C) [-0.9864; 3.4732]
- D) [0.7567; 4.3129]
- E) [-1.1837; 0.3789]

32. At your first day on the job as a hydrologic engineer, you are given a time series of water flow rates Q at 20 different times t just after a rainfall event. You are told that this flow rate is expected to decay exponentially over the entire time period as the function $Q = Q_0 \exp(-t/\tau_r)$, where Q_0 and τ_r are constant quantities. Q_0 is the initial flow rate at the first time in the series, and τ_r is the residence time. You are asked to perform linear least-squares regression on the data to estimate the residence time τ_r .

In order to perform a linear least-squares regression on this data set, you must first transform it. Which of the following equations represents a correct transform of the above equation that could be used to solve for τ_r via linear least-squares regression?

- A) $Q = Q_0(\exp(-t))^{\tau_r}$
- B) $Q/Q_0 = \exp(-t)/\exp(\tau_r)$
- C) $Q/Q_0 = \ln(-t)/\ln(\tau_r)$
- D) $\ln(Q) = Q_0 - \ln(t)/\tau_r$
- E) $\ln(Q) = \ln(Q_0) - t/\tau_r$

33. Consider a data set of x - and y -values. For which of the following equations can the coefficients a and b be fitted using linear least-squares regression without any intermediate mathematical manipulation?

- A) $y = \cos(ax) + \sin(b + x)$
- B) $y = a \cos^2(x) + b \log(x)$
- C) $y = (a + x/a + x^2)^{-1} + bx$
- D) $y = \exp(ax) + b \cos(x)$
- E) None of the above

34. Consider a set of points defined by the following two variables:

```
x_data = [0, 4, 7, 10];  
y_data = [-5, 3, 17, 4];
```

`x_data` represents the x -coordinates of these points and `y_data` represents the corresponding y -coordinates. What is the y -coordinate of a point of x -coordinate 3.5, as best estimated by linear interpolation of the points given above?

- A) -4
- B) -5
- C) 3
- D) 2
- E) 5.75

35. Consider a set of points defined by the following two variables:

```
x_data = [0, 1, 2];  
y_data = [1, 3, 2];
```

x_data represents the x -coordinates of these points and y_data represents the corresponding y -coordinates. Call P_1 , P_2 , and P_3 the basis polynomials (in the same order as the data points) of the Lagrange polynomial L corresponding to these points. Which of the following statements is **TRUE**?

- A) $P_1(0) = 0$ and $L(0) = 1$
- B) $P_1(0) = 3$ and $L(0) = 3$
- C) $P_1(0) = 1$ and $L(0) = 0$
- D) $P_1(0) = 1$ and $L(0) = 1$
- E) $P_1(0) = 3$ and $L(0) = 1$

36. Consider a set of n points. Call L the Lagrange polynomial corresponding to this set of points. What can be said about the degree of the polynomial L ?

- A) The degree of L is $(n - 2)$ or less
- B) The degree of L is $(n - 1)$ or less
- C) The degree of L is n
- D) The degree of L is n or more
- E) The degree of L is $(n + 1)$ or more

37. Consider a real-valued function f that is differentiable over \mathbb{R} . The x -coordinates at which we know the value of this function are $x_1, x_2, \dots, x_{i-1}, x_i, x_{i+1}, \dots, x_{n-1}, x_n$, which are equally spaced with spacing Δx . Which of the following terms is absent from a second-order central-difference approximation of f' at $x = x_i$ using a spacing of Δx ?

- A) x_{i-1}
- B) x_{i+1}
- C) $f(x_i)$
- D) $f(x_{i-1})$
- E) $f(x_{i+1})$

38. Consider the following function:

```
function [out] = my_deriv(alpha, beta)
out = (alpha(3:end)-alpha(1:end-2)) ./ (beta(3:end)-beta(1:end-2));
end
```

This function takes as input arguments two vectors of class double **alpha** and **beta** which have the same size. **alpha** represents the values of a physical quantity α and **beta** represents the values of another physical quantity β . The output argument **out** represents which of the following?

- A) A central difference approximation of $d\beta/d\alpha$
- B) A central difference approximation of $d\alpha/d\beta$
- C) A forward difference approximation of $d\alpha/d\beta$
- D) A backward difference approximation of $d\beta/d\alpha$
- E) A forward difference approximation of $d^2\beta/d\alpha^2$

39. Consider a real-valued function f that is continuous over the interval $[a, b]$, where a and b are real constants. We apply Simpson's rule to estimate the value of

$$\int_a^b f(x)dx$$

Which of the following statements is **TRUE**?

- A) Simpson's rule provides exact results if f is a polynomial of degree two or less
- B) Lagrange polynomials can be used to construct the polynomial approximations used by Simpson's rule
- C) Simpson's rule approximates the integral over pairs of sub-intervals by fitting a quadratic polynomial through the corresponding points, and then calculating the integral of this polynomial analytically
- D) Simpson's rule does not always provide exact results
- E) All of the above statements are true

40. Consider a real-valued function f that is continuous over \mathbb{R} . Assume that this function f is monotonically increasing over \mathbb{R} (*i.e.* $x_2 > x_1 \implies f(x_2) > f(x_1)$). Consider the following integral:

$$\int_a^b f(x)dx$$

where a and b are two real numbers such that $a < b$. We estimate the value of this integral by dividing the interval $[a, b]$ into n sub-intervals ($n > 0$), and using three different numerical integration methods. I_{left} is the value calculated using the Left-Riemann method, I_{right} is the value calculated using the Right-Riemann method, and I_{trap} is the value calculated using the Trapezoid method. Which of the following statements is **TRUE**?

- A) $I_{\text{trap}} < I_{\text{left}} < I_{\text{right}}$
- B) $I_{\text{trap}} < I_{\text{right}} < I_{\text{left}}$
- C) $I_{\text{left}} < I_{\text{right}} < I_{\text{trap}}$
- D) $I_{\text{left}} < I_{\text{trap}} < I_{\text{right}}$
- E) $I_{\text{right}} < I_{\text{trap}} < I_{\text{left}}$

41. Consider the following function, which is missing some code:

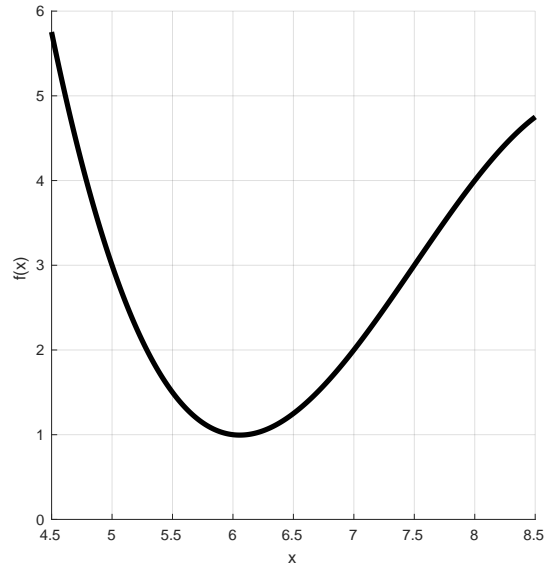
```
function [result] = my_simpson(x, y)
result = 0;
for index = 1:2:numel(x)-2
    x_local = x(index:index+2);
    y_local = y(index:index+2);
    % MISSING CODE
    result = result + area;
end
end
```

Assume that this function's input arguments x and y are column vectors which have the same size, which contain an odd number of elements, and such that $y(i) = f(x(i))$, where f is a continuous real-valued function. Also assume that the elements in x are in increasing order. Which set of commands can be used to complete this function so that it approximates $\int_a^b f(x)dx$ using Simpson's rule (where $a = x(1)$ and $b = x(\text{end})$)?

- A) `values = [x_local.^2, x_local, [1; 1; 1]] \ y_local;`
`antiderivative = @(x) values(1)*x^3/3 + values(2)*x^2/2 + values(3)*x;`
`area = antiderivative(x_local(end)) - antiderivative(x_local(1));`
- B) `quadratic = @(x) x_local(1) + x_local(2)*x + x_local(3)*x^2;`
`average = (quadratic(y_local(end)) - quadratic(y_local(1))) / 2;`
`area = average * (x_local(3)-x_local(1));`
- C) `slope_one = (y_local(2)-y_local(1)) / (x_local(2)-x_local(1));`
`slope_two = (y_local(3)-y_local(2)) / (x_local(3)-x_local(2));`
`area = (slope_one+slope_two)/2 * (x_local(3)-x_local(1));`
- D) `my_sum = y(1) + 4*sum(y_local(2:2:end-1)) + 2*sum(y_local(3:2:end-2)) + y(end);`
`delta_x = x(2) - x(1);`
`area = my_sum * delta_x/3;`
- E) All of the above

For the next 2 questions:

Refer to the following figure:



42. What is the forward difference approximation of f' (i.e. $\frac{df}{dx}$) at $x = 7$? Use an interval width of $\Delta x = 1$.

- A) 1
- B) -2
- C) -1
- D) 1.5
- E) 2

43. What is the left end-point Riemann sum approximation (*i.e.* the Left-Riemann approximation) of $\int_5^7 f(x)dx$? Use a sub-interval width of $\Delta x = 1$.

- A) 3
- B) 3.5
- C) -2
- D) 6
- E) 4

44. Consider the following first-order ordinary differential equation:

$$y' = -\frac{y}{2}$$

where the unknown y is a function of time t . Given that $y(0) = 3$, what is the value of $y(1)$ as approximated by the explicit Euler method using a single time step of width $\Delta t = 1$ s?

- A) 1
- B) 0
- C) 3
- D) 1.5
- E) 2

45. After executing the following code, which of the following quantities represents a Riemann sum approximation of $\int_0^{10} \sqrt{2x+3} dx$?

```
x = 0:0.5:10;
y = sqrt(2*x+3);
i_one = sum(y(1:end-1) .* (x(2:end)-x(1:end-1)));
i_two = sum(y(2:end) .* (x(2:end)-x(1:end-1)));
i_three = sum(0.5 * (y(1:end-1)+y(2:end)) .* (x(2:end)-x(1:end-1)));
```

- A) i_one
- B) i_two
- C) i_three
- D) Both i_one and i_two
- E) i_one, i_two, and i_three

46. We use a 2nd- and a 4th-order integration approximation to estimate the integral of a function, using equally-spaced points (the spacing between any two consecutive points is Δx). Which of the following statements is **TRUE**?

- A) The error made when using the 4th-order method is always twice as small as the error made when using the 2nd-order method
- B) On average, if we reduce Δx by a factor of 2, the error made when using the 4th-order method is divided by 16
- C) The error made when using the 4th-order method is always smaller than the error made when using the 2nd-order method
- D) On average, if we reduce Δx by a factor of 2, the error made when using the 4th-order method is divided by 4
- E) All of the statements above are true

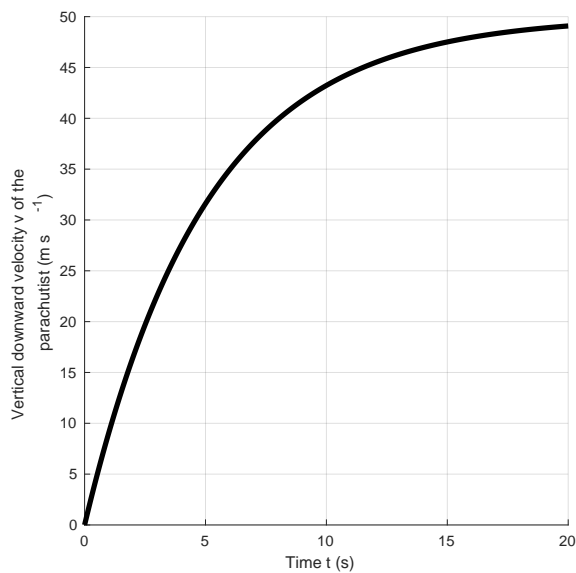
47. A parachutist falls with vertical downward velocity v , which is a function of time t , according to:

$$v' = g - \frac{c}{m}v$$

where:

- $v(t = 0) = 0$
- $m = 70 \text{ kg}$ is the mass of the parachutist
- $g = 10 \text{ m s}^{-2}$ is the acceleration of gravity
- $c = 14 \text{ kg s}^{-1}$ measures drag

Rank, in decreasing order, the parachutist's vertical downward velocity v at time $t = 10 \text{ s}$, as estimated using the explicit Euler method, the implicit Euler method, and the midpoint method, using in each case two time steps of $\Delta t = 5 \text{ s}$ each. The figure below shows the analytical solution of this equation.



- A) Implicit Euler; Midpoint; Explicit Euler
- B) Implicit Euler; Explicit Euler; Midpoint
- C) Midpoint; Implicit Euler; Explicit Euler
- D) Explicit Euler; Midpoint; Implicit Euler
- E) Explicit Euler; Implicit Euler; Midpoint

48. Consider the following third-order differential equation, where the unknown function u is a function of time t :

$$u''' + 5u'' - 4tu' + 3u = 0$$

Let:

$$X = \begin{bmatrix} u \\ u' \\ u'' \end{bmatrix}$$

Which of the following systems of first-order differential equations is equivalent to the third-order differential equation shown above?

A) $X' = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -3 & 4t & -5 \end{bmatrix} X$

B) $X' = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -5 & 4t & -3 \end{bmatrix} X$

C) $X' = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 3 & -4t & 5 \end{bmatrix} X$

D) $X' = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -3 & 4 & -5 \end{bmatrix} X$

E) $X' = \begin{bmatrix} -3 & 0 & 0 \\ 0 & 4t & 0 \\ 0 & 0 & -5 \end{bmatrix} X$

49. Consider the following first-order initial value problem:

$$y' = \sqrt{t}y + 2$$
$$y(t = 0) = -1$$

where the unknown y is a function of time t . For the purpose of calculating a numerical approximation of the solution to this equation over the interval $[0, 3]$, we wrote the following code:

```
n_points = 101;
times = linspace(0, 3, n_points);
step_size = times(2) - times(1);
% The vector "solution" will contain the numerical approximation
% of the solution at different time steps
solution = zeros(1, n_points);
solution(1) = -1;
for k = 2:n_points
    value_one = sqrt(times(k-1))*solution(k-1) + 2;
    value_two = solution(k-1) + value_one*step_size;
    value_three = sqrt(times(k))*value_two + 2;
    value_four = value_one + value_three;
    solution(k) = solution(k-1) + value_four*step_size/2;
end
```

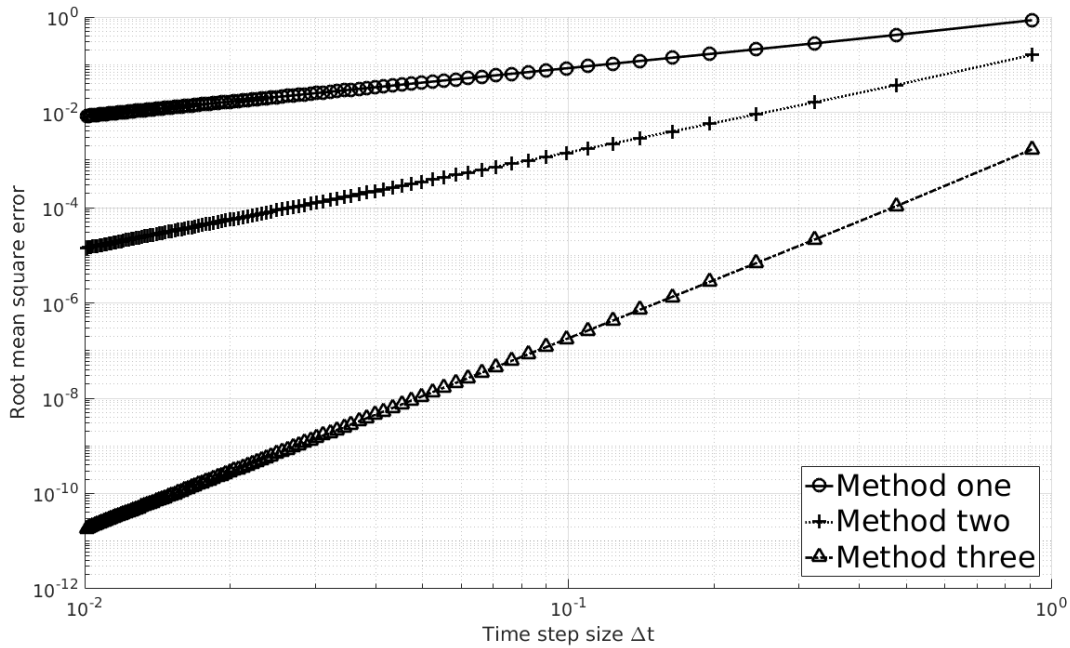
What is the numerical method that is used in this code to calculate a numerical approximation to the solution of this equation?

- A) Heun method
- B) Midpoint method
- C) Fourth-order Runge-Kutta method
- D) Implicit Euler method
- E) Explicit Euler method

50. We use different numerical methods (explicit Euler, midpoint, and fourth-order Runge-Kutta) to calculate approximate solutions of a first-order ordinary differential equation where the unknown is a function y of time t . These numerical approximations are calculated at n equally-spaced times t_1, t_2, \dots, t_n . The spacing between these times is $\Delta t = t_{i+1} - t_i$. We calculate the root mean square error between the exact solution y and the numerical approximation \hat{y} of this solution as:

$$\sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}(t_i) - y(t_i))^2}$$

We have plotted on the figure below the root mean square error as a function of Δt for these different methods.



Which method is which?

- A) Method one: Explicit Euler
Method two: Fourth-order Runge-Kutta
Method three: Midpoint
- B) Method one: Fourth-order Runge-Kutta
Method two: Midpoint
Method three: Explicit Euler
- C) Method one: Explicit Euler
Method two: Midpoint
Method three: Fourth-order Runge-Kutta
- D) Method one: Fourth-order Runge-Kutta
Method two: Explicit Euler
Method three: Midpoint
- E) Method one: Midpoint
Method two: Explicit Euler
Method three: Fourth-order Runge-Kutta