Engineering Analysis I, Fall 2022

Midterm 1

SOLUTIONS

Section number ________________

Section number	Discussion time	Instructor
23	9:00 a.m.	Prem Kumar
24	10:00 a.m.	Michael Honig
25	11:00 a.m.	Prem Kumar
26	8:00 a.m.	Randy Freeman
27	12:00 p.m.	Michael Honig

This exam is closed-book and closed-notes. Calculators, computers, phones, or other computing/communication devices are not allowed.

Students should skip this page—it is only for graders.

Question	Points	Score
1	30	
2	25	
3	25	
4	20	
Total:	100	

Answer each question in the space provided. There are 4 questions for a total of 100 points.

1. Put a check mark ✓ in the box next to **EACH** correct answer. Note that there may be more than one correct answer for each question!

   (a) [6 points] Which of the following six MATLAB statements will create (or overwrite) a variable `A` and assign it the matrix $\begin{bmatrix}2 & 5 & 8\\ 9 & 5 & 1\end{bmatrix}$?

   □ `A = [2 5 8];[9 5 1]`

   ✓ `A = [2 5 8; 9 5 1]`

   ✓ `A = [2 9; 5 5; 8 1]'`

   □ `[[2;9] [5;5] [8;1]]`

   □ `A = [2 5 8; 9 5 1] = A`

   ✓ `A = [2:3:8; 9:-4:1]`

   (b) [4 points] Which of the following four MATLAB statements will replace the second column of an existing matrix variable `A` with the sum of its first and fourth rows? Assume that `A` is a 4 by 4 square matrix.

   □ `A(2) = A(1)+A(4)`

   □ `A(2,:) = A(:,1) + A(:,4)`

   ✓ `A(:,2) = A(1,:) + A(4,:)`

   ✓
   ```
   A(1:size(A,2),2) = ...
   A(1,1:end) + A(4,1:end)
   ```

   (c) [4 points] Which of the following four MATLAB statements will *not* generate an error message?

   ✓ `x = 2 == [2 3]`

   ✓
   ```
   y = rand(5,1);
   x = y(y<.1 & y>.9)
   ```

   ✓ `x = [2 3].*[3 1]`

   ✓ `x = sqrt(10:1)`

(d) [4 points] Which of the following four MATLAB logical expressions will return a value of logical `1` (meaning "true")?

✓ `(1>=1) || (x==4)`

✓ `~(5~=5) & ~(3<=2)`

□ `(1>2) & (3>2)`

□ `(3>1) && (3=3)`

(e) [6 points] Which of the following six MATLAB statements is a valid first line of a loop? Assume that the variable `k` already exists in the workspace and contains the row vector `[3 2 1]`.

□ `while k = 1:3`

✓ `for k = 2:6`

✓ `for jj = k+2`

✓ `while 0 < k < 4`

✓ `while k < 5`

✓ `for ii = k`

(f) [6 points] Which of the following six blocks of code will square each element of an existing (and possibly non-square) matrix `A`? Note that the squared values should be accessible after the calculations are performed.

✓
```
A = (zeros(size(A))+A).*A
```

□
```
A = A*A;
```

□
```
A = A^2;
```

✓
```
A = A.*ones(size(A)).*A;
```

✓
```
for ii=1:size(A,1)*size(A,2)
    A(ii) = A(ii)*A(ii);
end
```

✓
```
for ii = 1:size(A,1)
    jj = 1;
    while jj <= size(A,2)
       A(ii,jj) = A(ii,jj)^2;
       jj = jj + 1;
    end
end
```

2. Suppose each section of code below is run in MATLAB. If MATLAB generates an error message for the given code section, write “error” on the associated line, and otherwise write the value the variable x will have after the code section is run.

(a) [5 points]

```
clear;
x = 0;
for k = 100:-10:1
    if (k <= 20)
    x = x + k;
    end
end
```

(a) 30

(b) [5 points]

```
clear; x = 3;
if ((x>1) || (x>6)) && x < 5
    x = 17;
end
```

(b) 17

(c) [5 points]

```
clear; x = 0;
while y <= 20
    x = x + 4;
    y = y + 5;
end
```

(c) error

(d) [5 points]

```
clear; x = 0;
for k = 1:3:14
    if k == 10
        x = 4;
    else
        x = x + 1;
    end
end
```

(d) 5

(e) [5 points]

```
clear; x = 0;
for x <= 6
    while k = 5:-1:1
        x = x + 1;
    end
end
```

(e) error

3. [25 points] The following code simulates a series of random walks (trials), each starting at position 0. Each step is randomly chosen from the set $\{-1, 0, 1\}$. For each random walk it counts the number of steps it takes to reach a particular location L, which can be any integer value. Each random walk terminates when either the location reaches L or the number of steps reaches a maximum number, specified by the user. The program then computes the average of the number of steps taken over the trials and the fraction of trials for which the number of steps reaches the maximum number.

   Fill in the blanks in the code to complete the program. Each answer should complete a single statement (one line of code).

```
% set the number of random walks, or trials
trials = 10000;

% prompt the user to input the final location L (an integer)
% and the maximum number of steps
L = input('Enter the final location L: ');             (1 point)
max_steps = input('Enter the maximum number of steps: ');   (1 point)

% initialize array containing the number of steps
% reached by each trial
num_steps = zeros(1,trials);

% loop over all of the trials
for ii = 1:trials                               (2 points)

% fill in the missing statement
    pos = 0;                                    (2 points)
    % start the loop to update the position variable "pos" until
    % it reaches L or the number of steps reaches the maximum
    while (pos ~= L) && (num_steps(ii) < max_steps)  (4 points)

        % generate a random step, either -1, 0, or 1
        step = randi([-1,1]);            (2 points)
        % update the position
        pos = pos + step;           (2 points)
       % update the number of steps
        num_steps(ii) = num_steps(ii) + 1  (3 points)
    end
end

% Compute the average of the number of steps taken over all trials.
avg_steps =  sum(num_steps)/trials;                (2 points)

% PROGRAM CONTINUES TO NEXT PAGE

% Compute the fraction of trials for which the number of steps
% reached the maximum number.
frac_max =  sum(num_steps == max_steps)/trials;     (3 points)

% Print the results using two fprintf statements as follows:
% Avg steps to reach __ is ___.
% Fraction of trials where num_steps = max_steps is ___.

% Each statement should fill in the blanks.
% Print L as an integer and the other values as real numbers.
fprintf('Avg steps to reach %i is %f.',L,avg_steps); (3 pts for both)
fprintf('Fraction of times num_steps = max_steps is %f.',frac_max);

```

4. [20 points] For this question, we will first generate a random matrix of integers (between 2 and 20), with random dimensions (between 1 and 20). Our goal is to find, for each element, the highest power it can be raised to that keeps it under 1e6. For example, if an element is 11, the highest power it can be raised to is 5, since $11^5 = 161051$, while $11^6 = 1771561 > 1e6$. The output should be a matrix of the same size as the original, where each element is this highest power. For example, consider the following input matrix `A` and output `B`.

$$\mathtt{A} = \begin{bmatrix} 11 & 6 \\ 13 & 18 \\ 4 & 3 \end{bmatrix} \qquad \mathtt{B} = \begin{bmatrix} 5 & 7 \\ 5 & 4 \\ 9 & 12 \end{bmatrix}$$

Fill in the blanks below to complete this task.

In this problem, you may use any of the following MATLAB functions:
`abs, factorial, min, max, mean, rand, randi, sqrt, sum, zeros.`

```
% Create the input random matrix.
A = randi([2 20],randi(20),randi(20));

% Find dimensions of A
m =   size(A,1)                                        (2 points)
n =   size(A,2)                                        (2 points)

% Create an output matrix B filled with zeros
B =   zeros(m,n)                                       (4 points)

for ii = 1:m
    for jj =   1:n                                     (2 points)
        my_exp =   1;                                  (2 points)

        % Iterate as stated in the introduction.
        while   A(ii,jj)^my_exp < 1e6                  (4 points)
            my_exp = my_exp + 1;
        end

        % Fill in the corresponding value in the B matrix.
        B(ii,jj) =   my_exp - 1;                       (4 points)
    end
end
```