Camlin Page

LAB-1

Ans 1) • Take i/p from user in the command line via i/p command and display appropriate output

• Code Snippet Attached later (at the end of all the observations).

Ans 2) (i) Use zeros (r, c) [r, c → no of rows, no. of columns]

(ii) Use ones (r, c)

(iii) Use eye (r, c)

Ans 3) • To find max element in Vector A,

maxnumber = max(max(A))

To find indices of max number,

[x, y] = find (A == maxnumber)

• Similarly for min element,

minnumber = min(min(A))

[x, y] = find (A == minnumber)

(2)

Ans 4) · Take the input via 'input' command in a nested for loop.

· Use linsolve to solve the linear system of eq's.

· $AX = B$ (A → coefficient Matrix, B → Constant Vector)

· Here $X = \begin{bmatrix} 0.3018 \\ -0.0423 \\ -0.4556 \end{bmatrix}$

Ans 5) · Take i/p from command line via input command.
· Use if-else statements to present the desired o/p
· Code snippet attached at the end.

Ans 6) · Declare 2 vectors or mxn matrices A & B.

· $X = A + B$. (Their dimensions must match)

· Code snippet Attached later

Ans 7) · $e^x = 1 + x + \frac{x^2}{2!} + \cdots$

$f^n(x) = e^x$

$\text{Error} = \frac{x^{n+1}}{(n+1)!} \cdot (\text{Max})^{n+1}$

· Check the current value of e from the Taylor series with respect to the actual value in a loop untill the error condition is satisfied

· Code snippet Attached at the end.

(3)

Ans 8) a) O/P : 5 [x(3)]
Gives the 3rd element in the vector

b) O/P : 3 1 5 7 9 2 6 [x(1:7)]
Outputs all the elements from index 1 to index 7

c) O/P : 3 1 5 7 9 2 6 [x(1:end)]
Outputs all the elements from index 1 to the last index of the array

d) O/P : 3 1 5 7 9 2 (x(1:end-1))
outputs all elements from 1st to (n-1)th index in vector

e) O/P : 2 7 1 (x(06:-2:1))
Starts from the 6th index and goes upto the 1st index in jumps of 2

f) O/P : 3 2 1 3 3 [x([1 6 2 1 1])]
outputs elements present in the index mentioned in the array in the argument

g) O/P : 33
outputs sum of all elements in the row / vector.

Ans 9) a) >> A(1,:)

b) >> A(end-1:end, :)

P.T.O.

c) ans = (A, 1) ; // column wise sum

d) ans = (A, 2) ; // row wise sum

e) ans = std(A) / sqrt(3) ;

Ans 10) a) Yes. [3 5 13]
Both x & y have similar dimensions

b) Yes [4 5 14 ; 6 6 15]
x gets added to both rows of A.

c) Yes :

$$\begin{bmatrix} 3 & 2 & 6 \\ 6 & 5 & 9 \\ 10 & 9 & 13 \end{bmatrix}$$

Element wise Addition takes place.
As in ans(1,1) = x(1) + y(1) and so on.

(5)

e) No

A and [x' y'] have different dimensions. Hence while subtracting it gives an error

f) No

Inconsistency in Row and column size. Both x and y' should be row vectors or column vectors else mismatch occurs

β) Yes $\begin{bmatrix} 1 & 4 & 8 \\ 2 & 1 & 5 \end{bmatrix}$

g) Yes $\begin{bmatrix} 0 & -2 & 3 \\ 2 & -1 & 4 \end{bmatrix}$

3 subtracted from every element

Ans (i) g) Gives Transpose of A.

2	3	8
7	1	1
9	5	2
7	6	5

b) Gives all the row elements in column 1 & column 4

2	7
3	6
8	5

(6)

c) Outputs all elements in row 2 & 3 and col 3 and col 1

5 3
2 8

d) Converts it into a 2×6 vector

2 5 1 9 2 6
3 7 1 5 7 5

e) Display all elements into a single column vector

f) flips every column present in the vector

g) flips every row present in the vector

h) Error in the statement. Dimension of arrays to be concatenated are not same

i) Display all column elements from row 1 to row 3

j) Concatenate row 1 and row 2 at the end of the vector to make it a 5×4 vector

k) Gives a row vector with column wise sum of A

l) Gives a row vector with row wise sum of A

m) Gives a column vector with row wise of A.

(7)

Camlin Page

f) Adds an extra row and column which stores the sum of the respective row and column of present above / before it.

Ans 12) ~~· Mean(A)~~

a) · avg = mean(F) // Assigns mean of each column to a cell in avg.

b) · S · std(F) // Assigns standard deviation of each column to a col in s

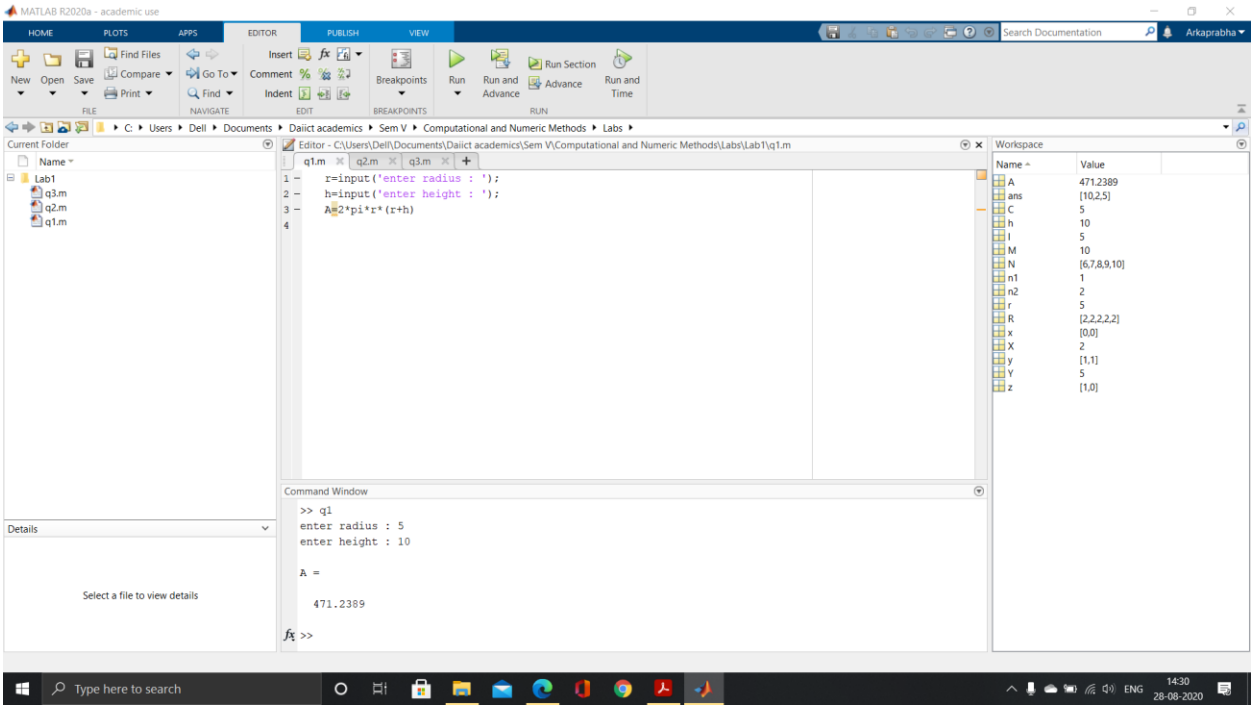
c) t.score = (mean(F) == 0) // Creates a logical array where 0 denotes that mean of that column is not 0 and 1 denotes that the mean is 0)

Ans 13) a) sum(x) // computes total sum.

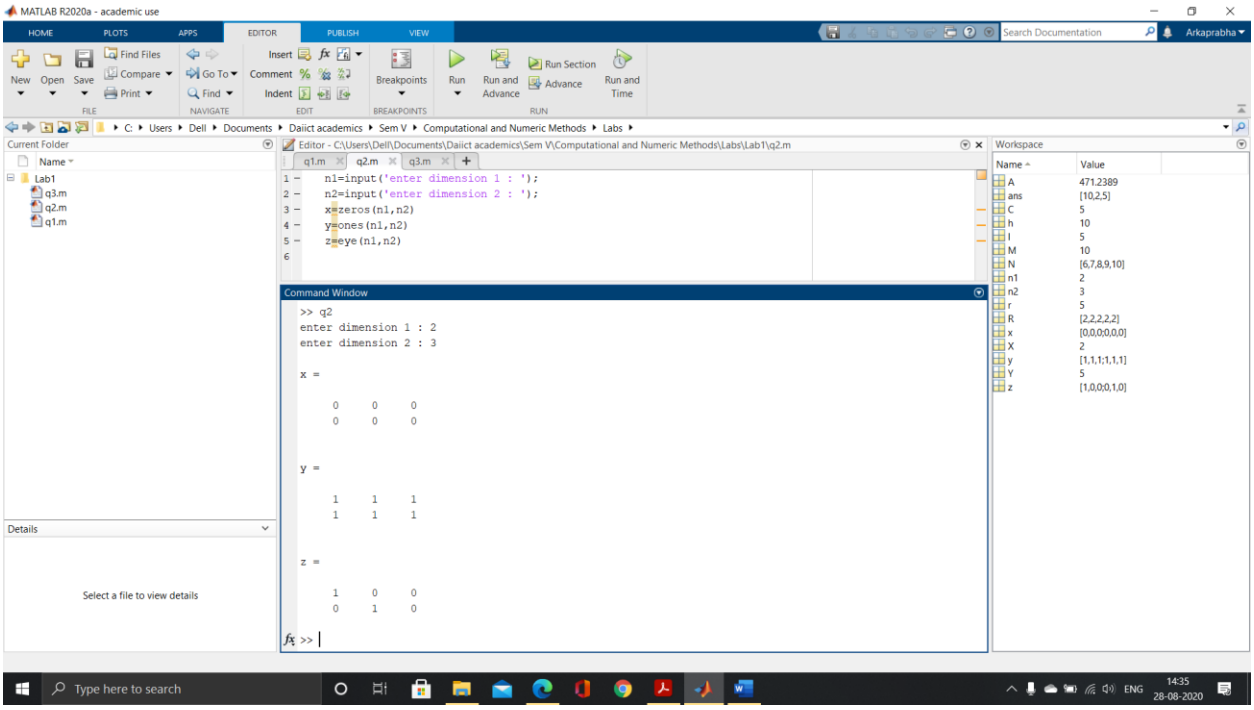
b) cumsum(x) // computes cumulative sum

c) sin(x) // returns an array of same length with sine of the elements

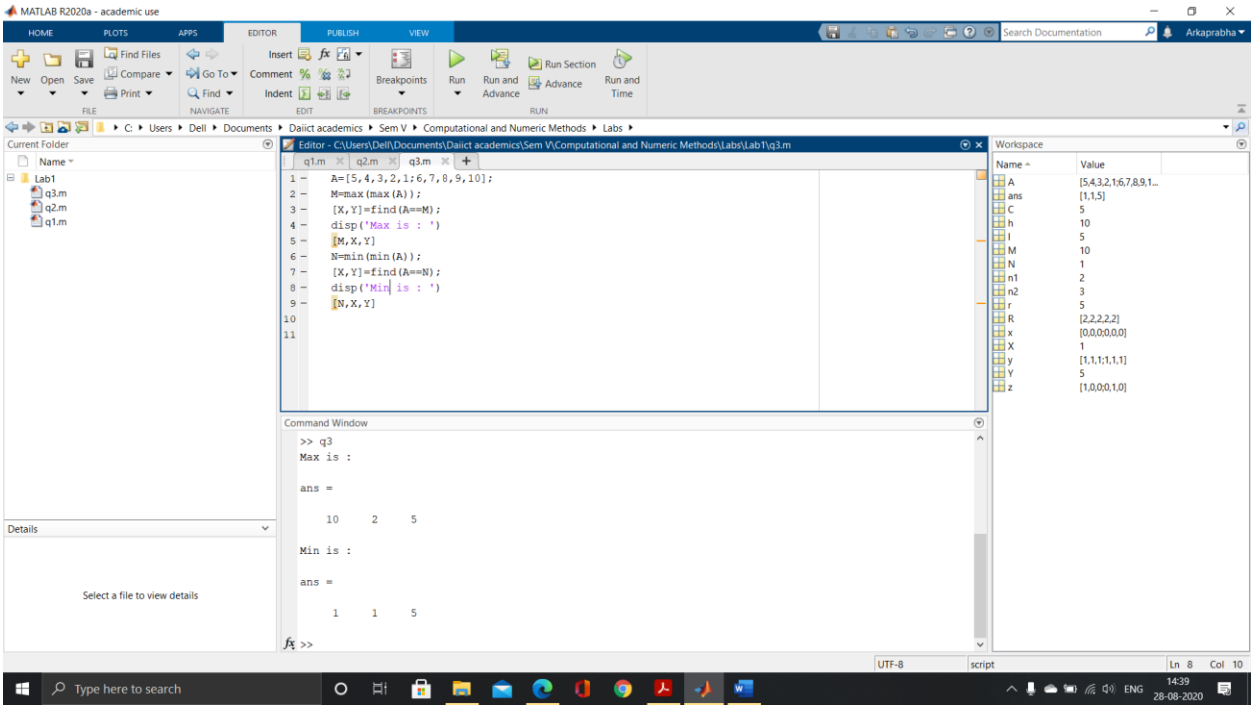
Code Snippet 1



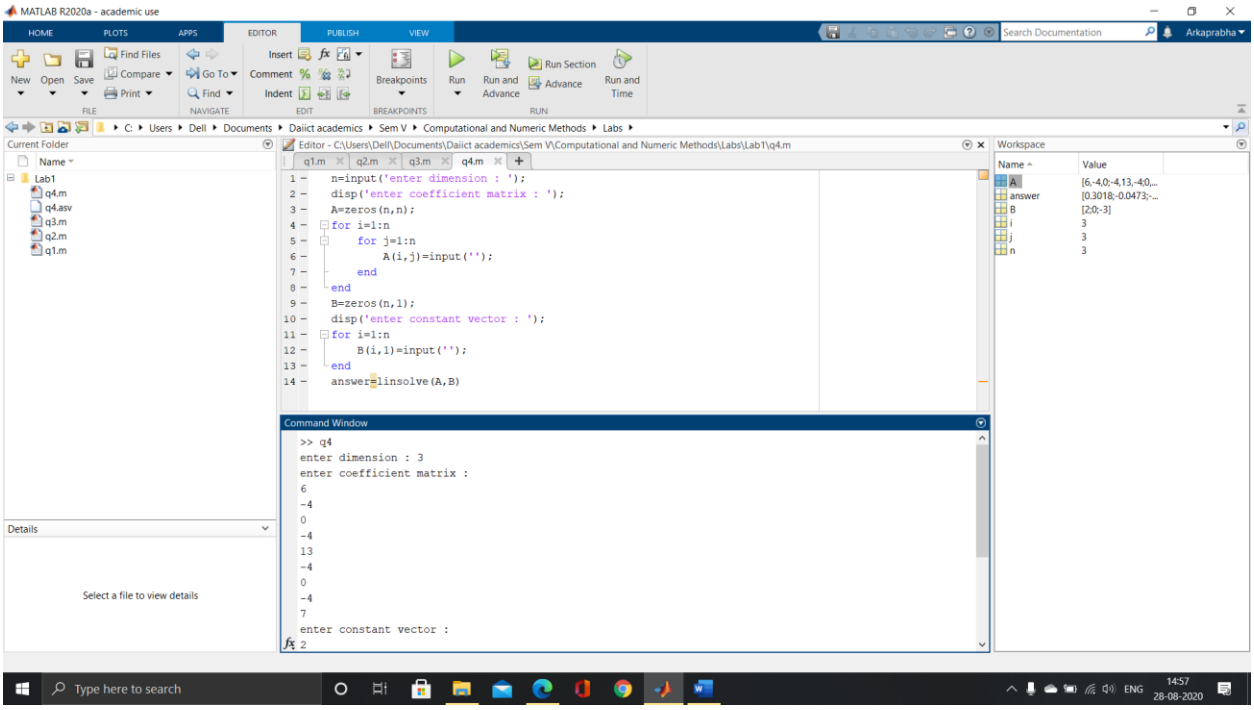
Code Snippet 2

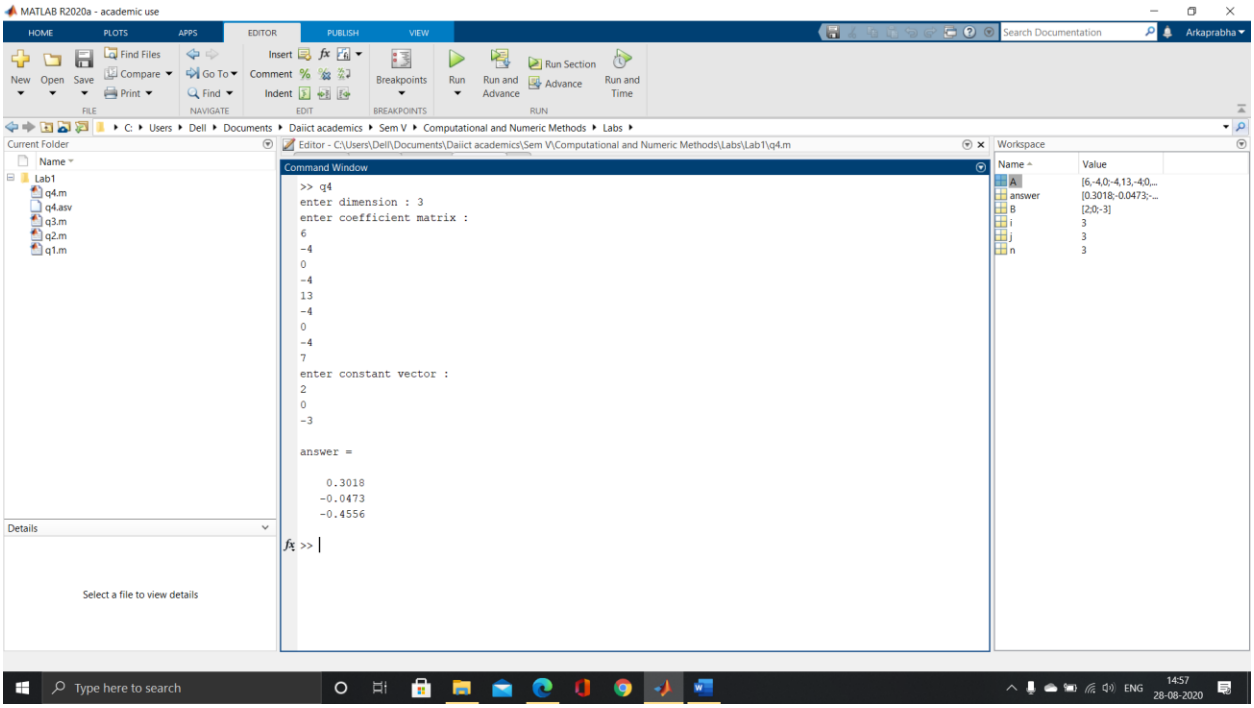


Code Snippet 3

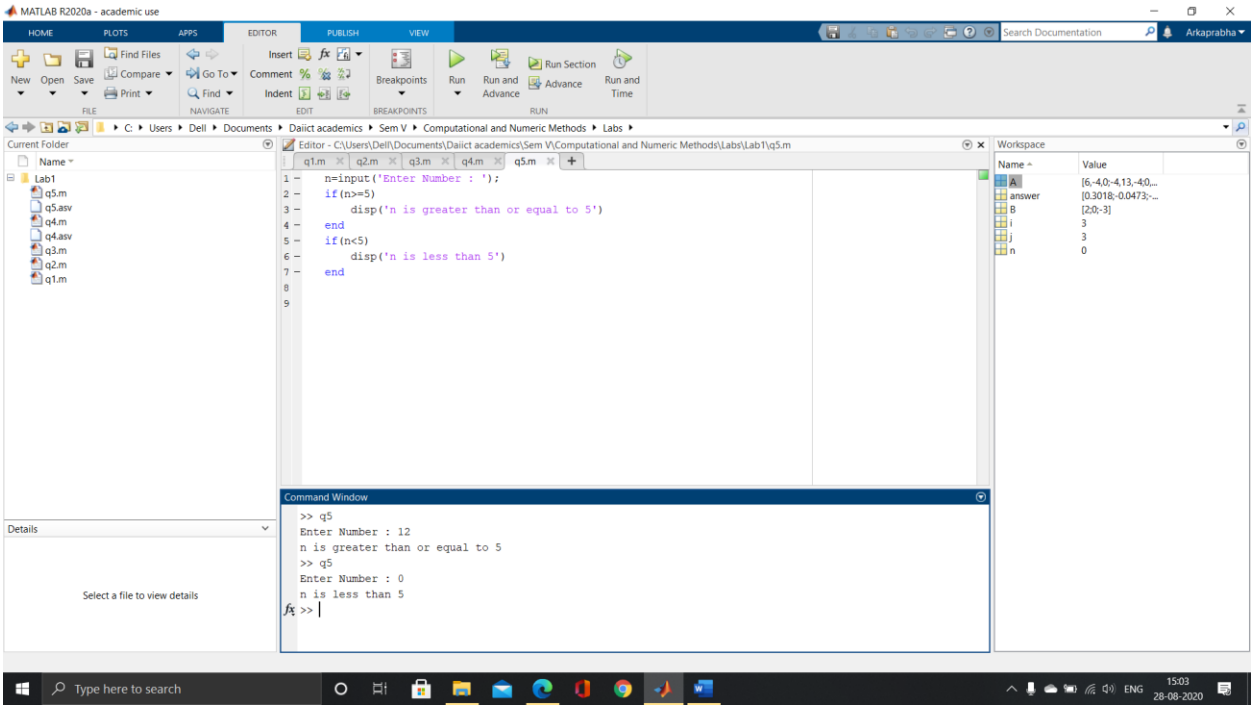


Code Snippet 4

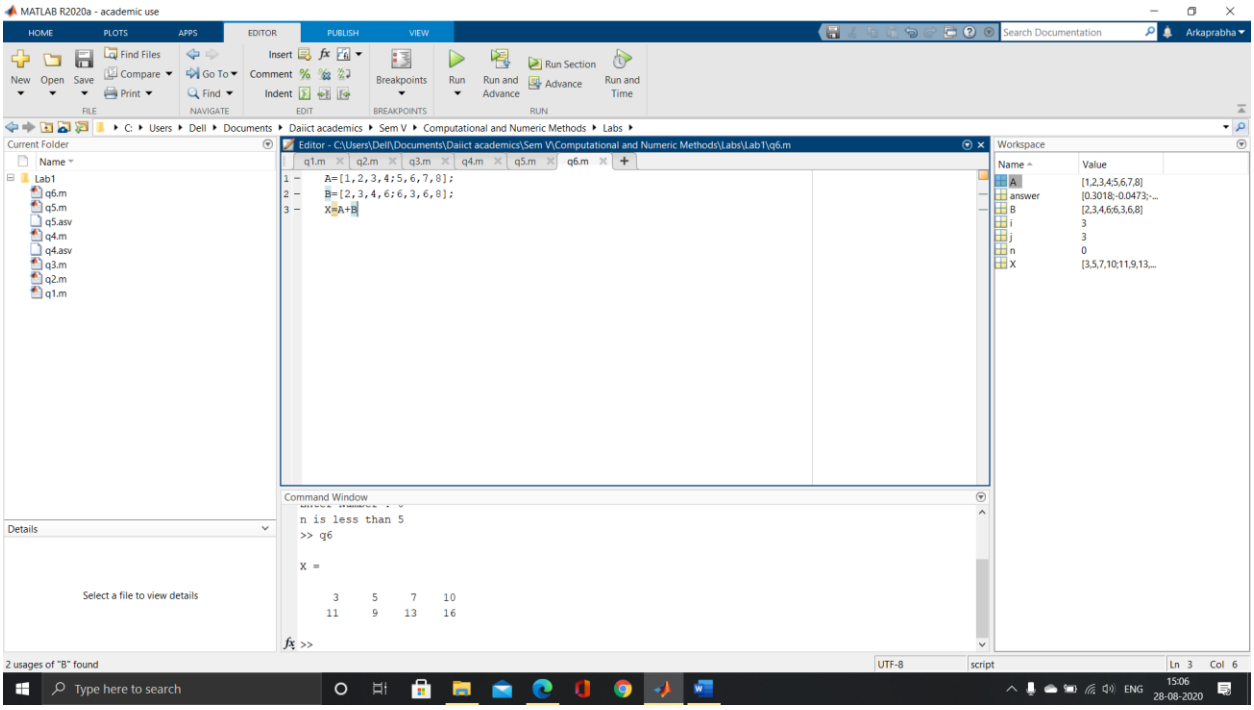




Code snippet 5



Code Snippet 6



Code Snippet 7