DAILY ASSESSMENT

Date:	07/07/2020	Name:	Davis S. Patel
Course:	Matlab Onramp	USN:	4AL16EC045
Topic:	Indexing into and Modifying Arrays Array Calculations Calling Functions Obtaining Help Plotting Data	Semester & Section:	8th - A
GitHub Repository:	Davis		

FORENOON SESSION DETAILS
Image of session
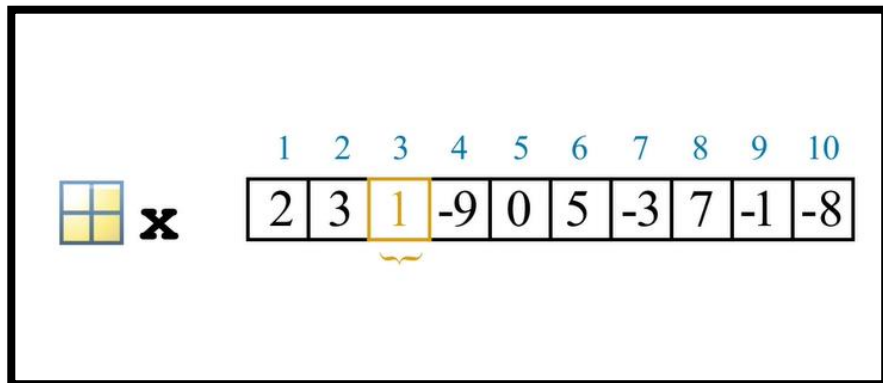
Performing Array Operations on Vectors

Instructions are in the task pane to the left. Complete and submit each task one at a time.

This code sets up the interaction.

```
load datafile
density = data(:,2);
v1 = data(:,3);
v2 = data(:,4);
```

Obtaining Multiple Outputs

Instructions are in the task pane to the left. Complete and submit each task one at a time.

This code sets up the interaction.

```
load datafile
v1 = data(:,3)
v2 = data(:,4)
```

Task 1

REPORT –

Indexing with Element Positions

The most common way is to explicitly specify the indices of the elements. For example, to access a single element of a matrix, specify the row number followed by the column number of the element.

A = [1 2 3 4; 5 6 7 8; 9 10 11 12; 13 14 15 16]

A = $4{\times}4$

```
 1    2    3    4
 5    6    7    8
 9   10   11   12
13   14   15   16
```

e = A(3,2)

e = 10

e is the element in the 3,2 position (third row, second column) of A.

You can also reference multiple elements at a time by specifying their indices in a vector. For example, access the first and third elements of the second row of A.

r = A(2,[1 3])

r = $1{\times}2$

```
 5    7
```

To access elements in a range of rows or columns, use the colon. For example, access the elements in the first through third row and the second through fourth column of A.

r = A(1:3,2:4)

r = $3{\times}3$

```
 2    3    4
 6    7    8
10   11   12
```

An alternative way to compute r is to use the keyword end to specify the second column through the last column. This approach lets you specify the last column without knowing exactly how many columns are in A.

r = A(1:3,2:end)

r = $3{\times}3$

```
2    3    4
6    7    8
10   11   12
```

If you want to access all of the rows or columns, use the colon operator by itself. For example, return the entire third column of A.

r = A(:,3)

r = *4×1*

```
3
7
11
15
```

In general, you can use indexing to access elements of any array in MATLAB regardless of its data type or dimensions. For example, directly access a column of a datetime array.

t = [datetime(2018,1:5,1); datetime(2019,1:5,1)]

t = *2x5 datetime*
```
01-Jan-2018   01-Feb-2018   01-Mar-2018   01-Apr-2018   01-May-2018
01-Jan-2019   01-Feb-2019   01-Mar-2019   01-Apr-2019   01-May-2019
```

march1 = t(:,3)

march1 = *2x1 datetime*
```
01-Mar-2018
01-Mar-2019
```

For higher-dimensional arrays, expand the syntax to match the array dimensions. Consider a random 3-by-3-by-3 numeric array. Access the element in the second row, third column, and first sheet of the array.

A = rand(3,3,3);
e = A(2,3,1)

e = 0.5469

To call a function, such as max, enclose its input arguments in parentheses:

A = [1 3 5];

max(A)

ans = 5

If there are multiple input arguments, separate them with commas:

B = [10 6 4];

max(A,B)

ans = *1×3*

    10   6   5

Return output from a function by assigning it to a variable:

maxA = max(A)

maxA = 5

When there are multiple output arguments, enclose them in square brackets:

[maxA,location] = max(A)

maxA = 5

location = 3

Enclose any character inputs in single quotes:

disp('hello world')

hello world

To call a function that does not require any inputs and does not return any outputs, type only the function name:

clc

The clc function clears the Command Window.

Plot the count.dat Data

Create a time vector, t, containing integers from 1 to n.

t = 1:n;
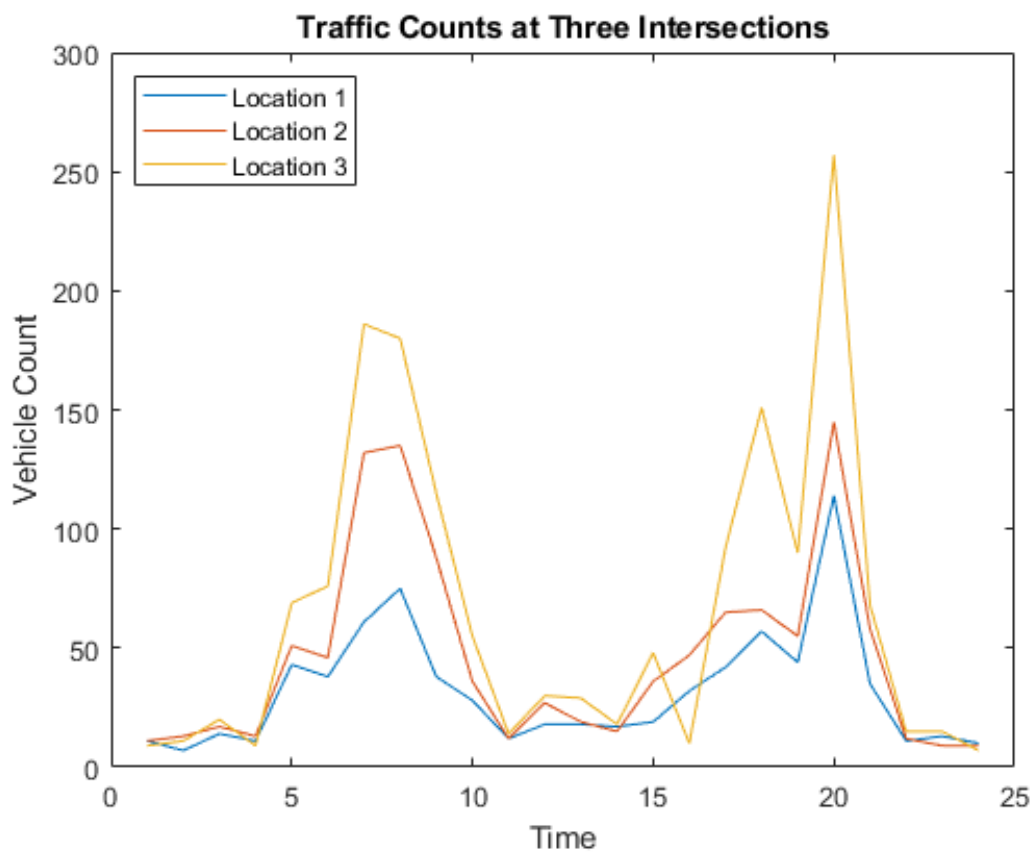
Plot the data as a function of time, and annotate the plot.

plot(t,count),

legend('Location 1','Location 2','Location 3','Location','NorthWest')

xlabel('Time'), ylabel('Vehicle Count')

title('Traffic Counts at Three Intersections')

DAILY ASSESSMENT
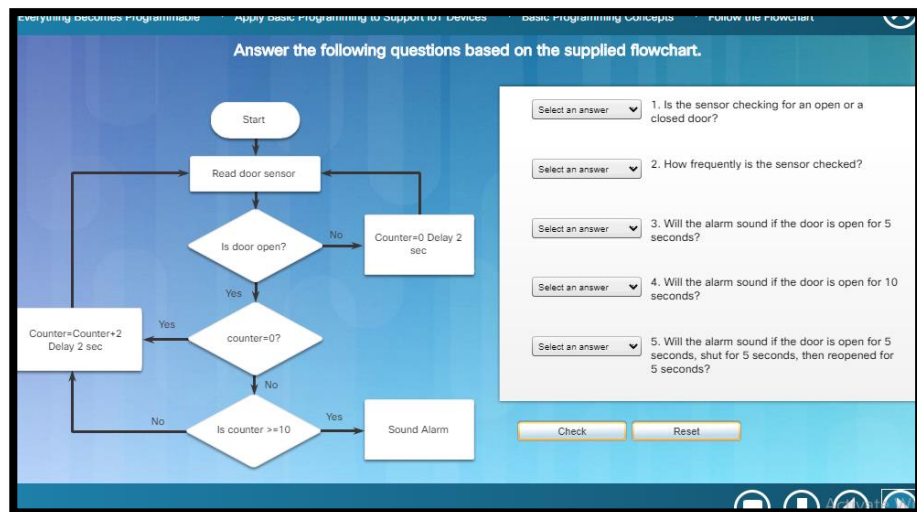
Date:	07/07/2020	Name:	Davis S. Patel
Course:	IOT	USN:	4AL16EC045
Topic:	Everything Becomes Programmable	Semester & Section:	8th - A
GitHub Repository:	Davis		

AFTERNOON SESSION DETAILS

Image Of Session

REPORT –

Smartphones do play a large role in the IoT, however, because many IoT devices can be controlled through an app on a smartphone. You can use your smartphone to communicate with your smart thermostat, for example, to deliver the perfect temperature for you by the time you get home from work. Another plus? This can eliminate unneeded heating or cooling while you're away, potentially saving you money on energy costs.

IoT devices contain sensors and mini-computer processors that act on the data collected by the sensors via machine learning. Essentially, IoT devices are mini computers, connected to the internet, and are vulnerable to malware and hacking.

Machine learning is when computers learn in a similar way to humans — by collecting data from their surroundings — and it is what makes IoT devices smart. This data can help the machine learn your preferences and adjust itself accordingly. Machine learning is a type of artificial intelligence that helps computers learn without having to be programmed by someone.

That doesn't mean your smart speaker will discuss the key points of last night's big game with you. But your connected refrigerator may send you an alert on your smartphone that you're low on eggs and milk because it knows you're near a supermarket.

Convenience is good. But connectedness can also open the door to cybercriminals. Cyberattacks already access data like bank logins, credit card numbers, and more. Strong security is essential.

Consider some of the potential risks of IoT — and those mini CPUs. What if someone hacks your IoT security cameras and watches your every move? What if a cybercriminal commandeers your smart TV, smoke alarms, or front-door lock? Connectedness can create vulnerabilities.

It's smart to help ensure these connections stay safe and secure — just like you protect yourself against cybercrime. The Internet of Things will continue to open up opportunities for new online threats. IoT devices are being attacked on an average of every two minutes, according to the Symantec 2017 Internet Security Threat Report. Cybercriminals have already hacked unsecured IoT devices to use their processing power in large-scale cyberattacks.

One of the best ways to help defend your networked devices is to make sure that your router is secure. That way, it can protect your entire home Wi-Fi network and the devices connected to it. Think of your Wi-Fi router as the front door to your online world.

The Internet of Things is designed to make our lives more convenient. Here are a few examples:

- Smart bathroom scales working in tandem with your treadmill, delivering food preparation ideas to your laptop or smartphone, so you stay healthy.

- Security devices monitoring your home, turning lights on and off as you enter and exit rooms, and streaming video so you can check in while you're away.

- Smart voice assistants placing your usual takeout order on command, making it a breeze to get fresh food delivered to your door.