**TITLE OF LAB: (INTRODUCTION TO CAUSAL AND NON-CAUSAL SIGNAL)**

**LAB # 08**



**Spring 2022**

**CSE301L Signals & Systems Lab**

Submitted by: **Safi Ullah Khan**

Registration No. **20PWCSE1943**

Class Section: **B**

“On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work.”

Student Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Submitted to:

**Engr. Durr-e-Nayab**

Day, Date (e.g Monday, June 20th, 2022)

**Department of Computer Systems Engineering**

**University of Engineering and Technology, Peshawar**

**OBJECTIVES OF THE LAB**

**‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐**

This lab aims at the understanding of:

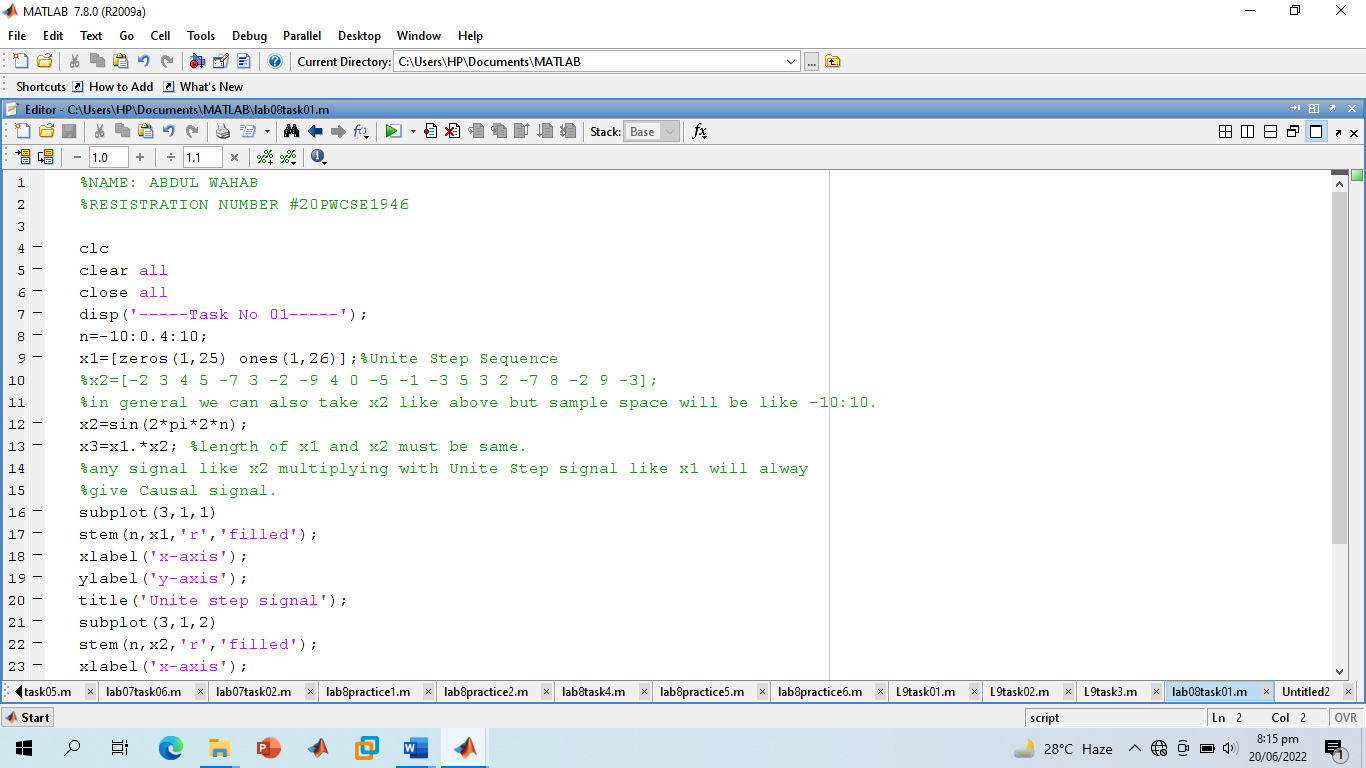
* Making Signals Causal and Non‐Causal
* Convolution
* Properties of Convolution

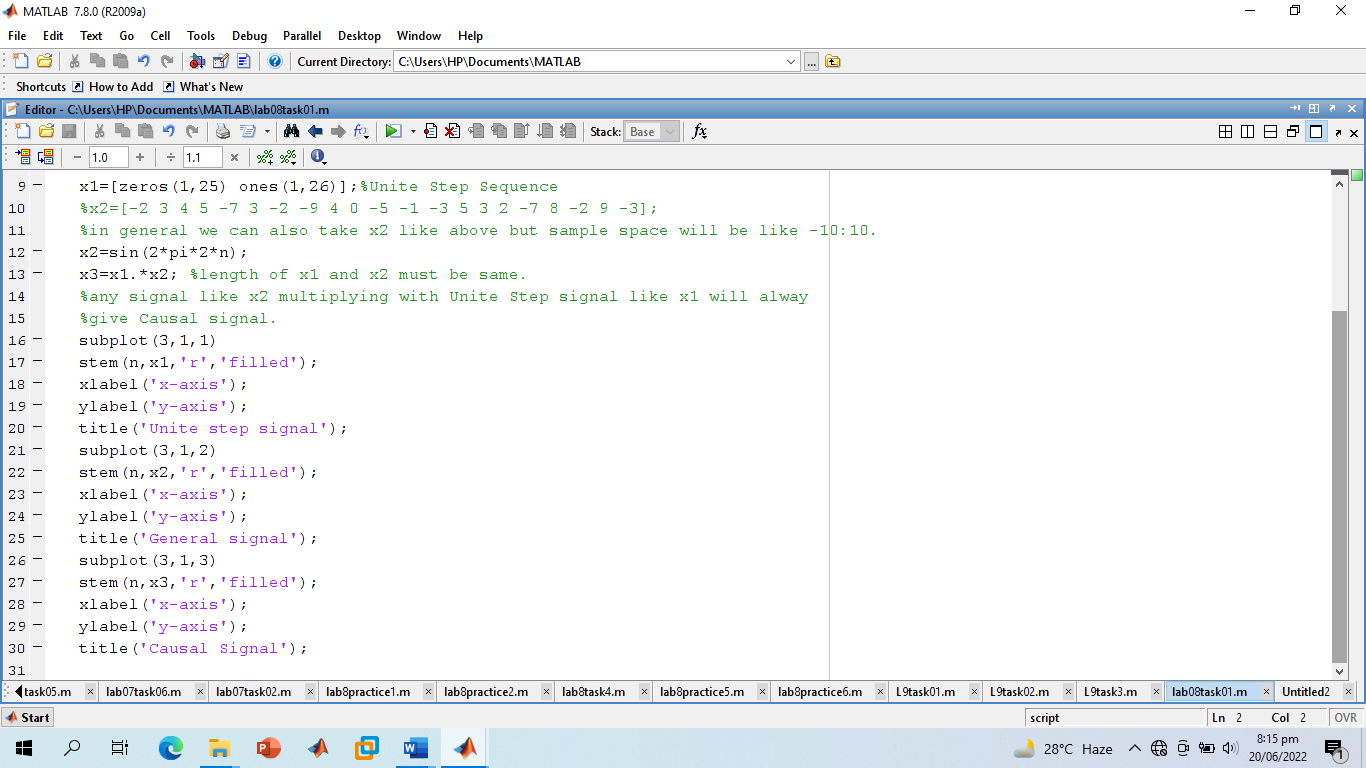
**‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐**

**-------------------------TASK 01--------------------------**

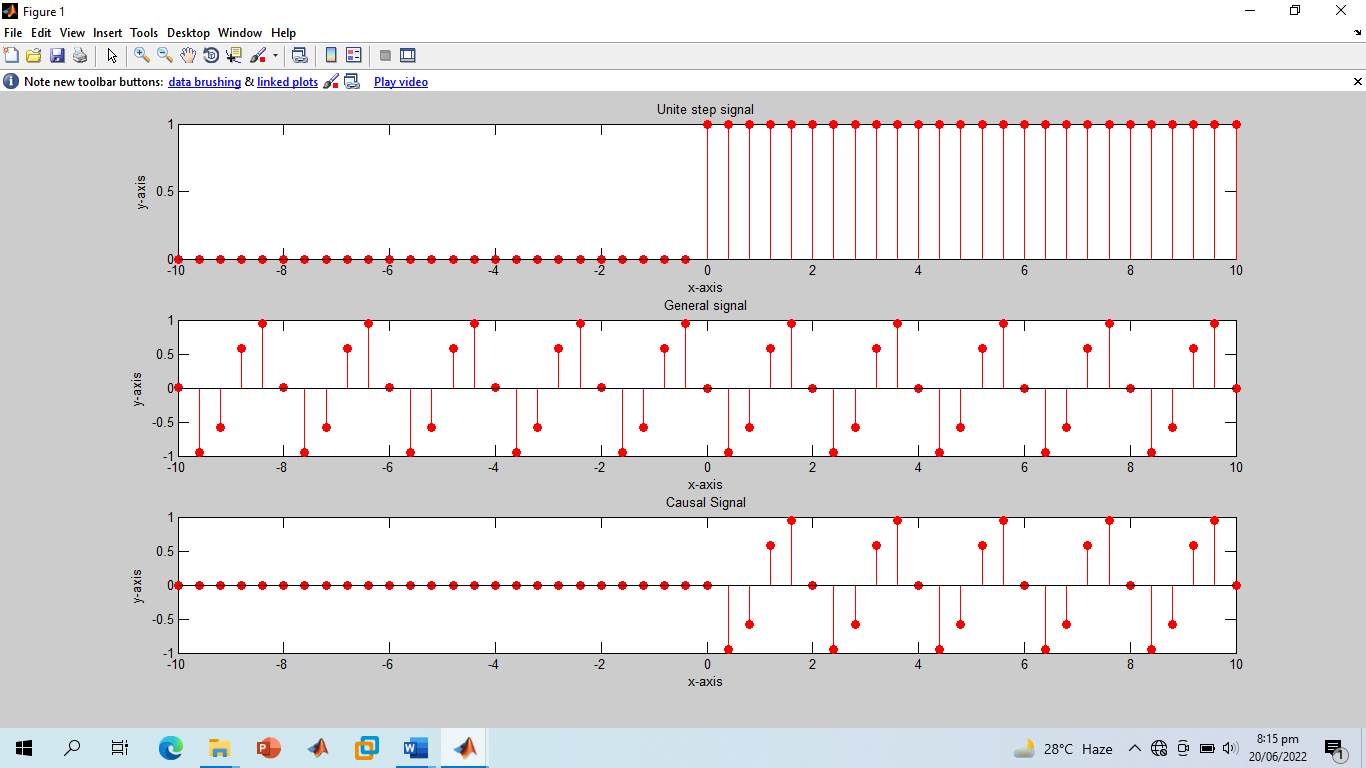
* Sample the signal given in above example to get its discrete‐time counterpart (take 10 samples/sec as sampling rate). Make the resultant signal causal. Display the lollipop plot of each signal.

**Screenshot of Input:**





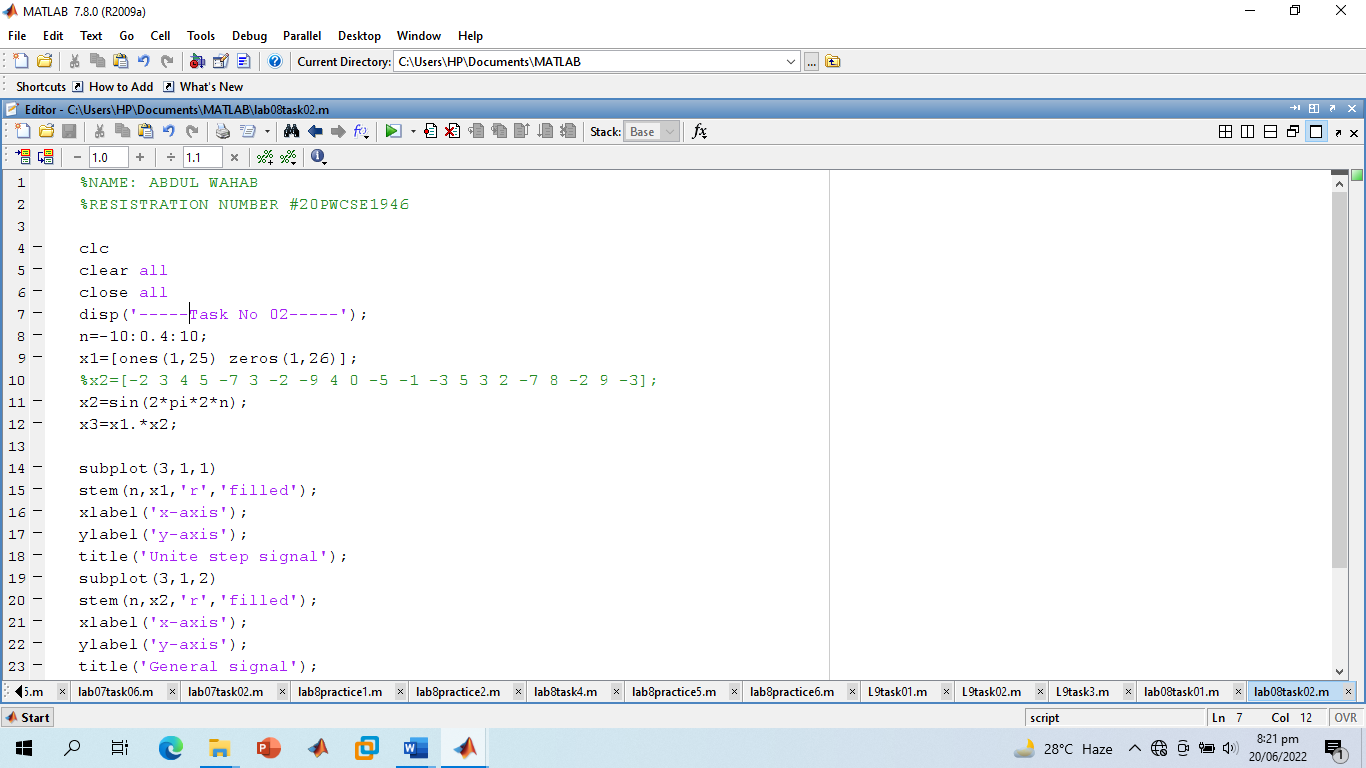
**Screenshot of Output:**

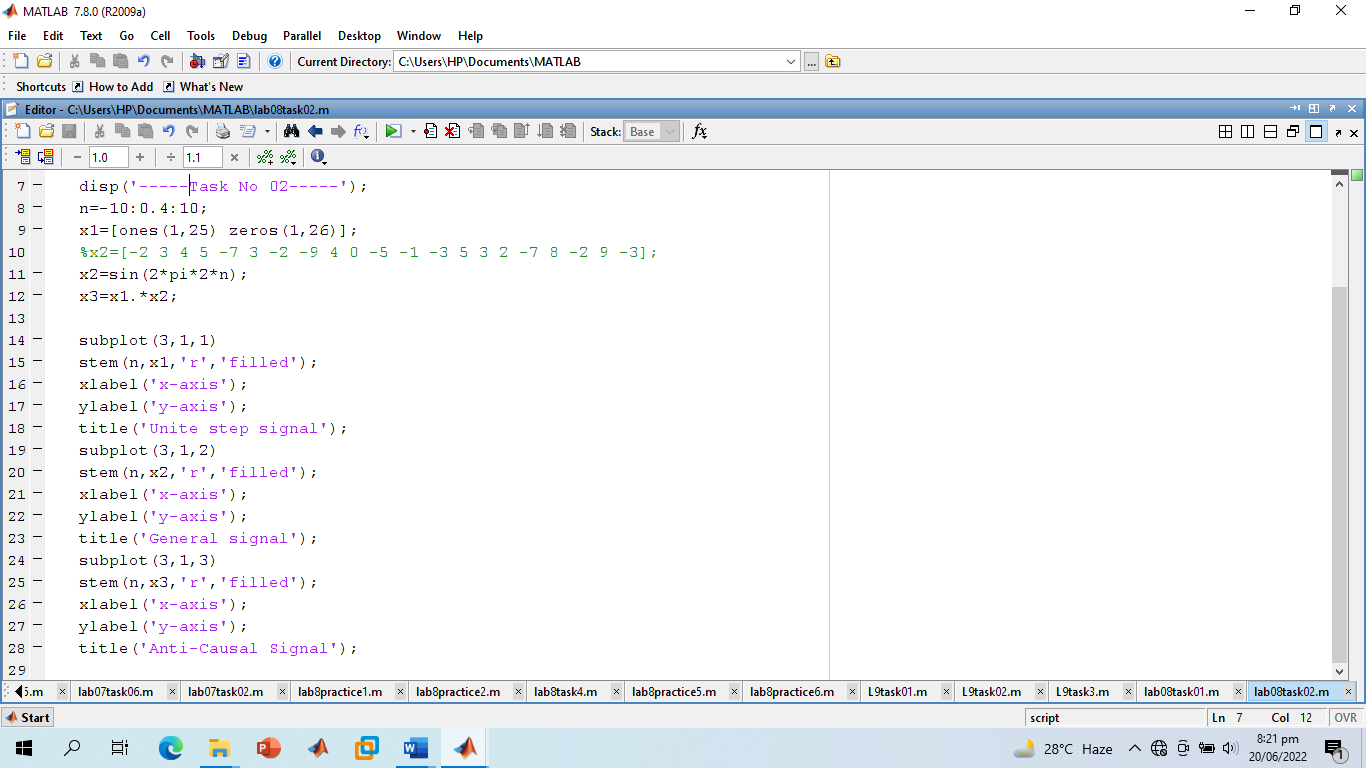


**-----------------------TASK 02--------------------------**

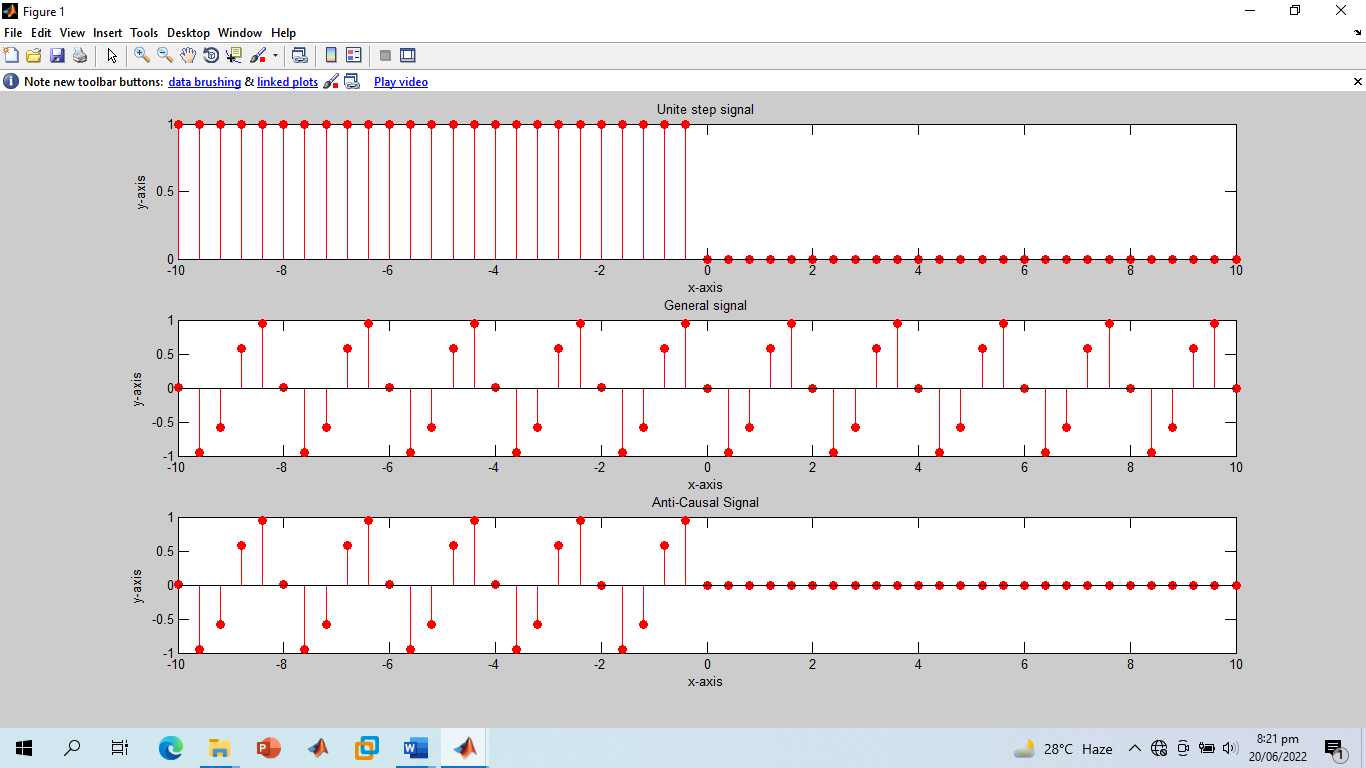
* A signal is said to be anti‐causal if it exists for values of n<0. Make the signal given in above example anti‐causal.

**Screenshot of Input:**





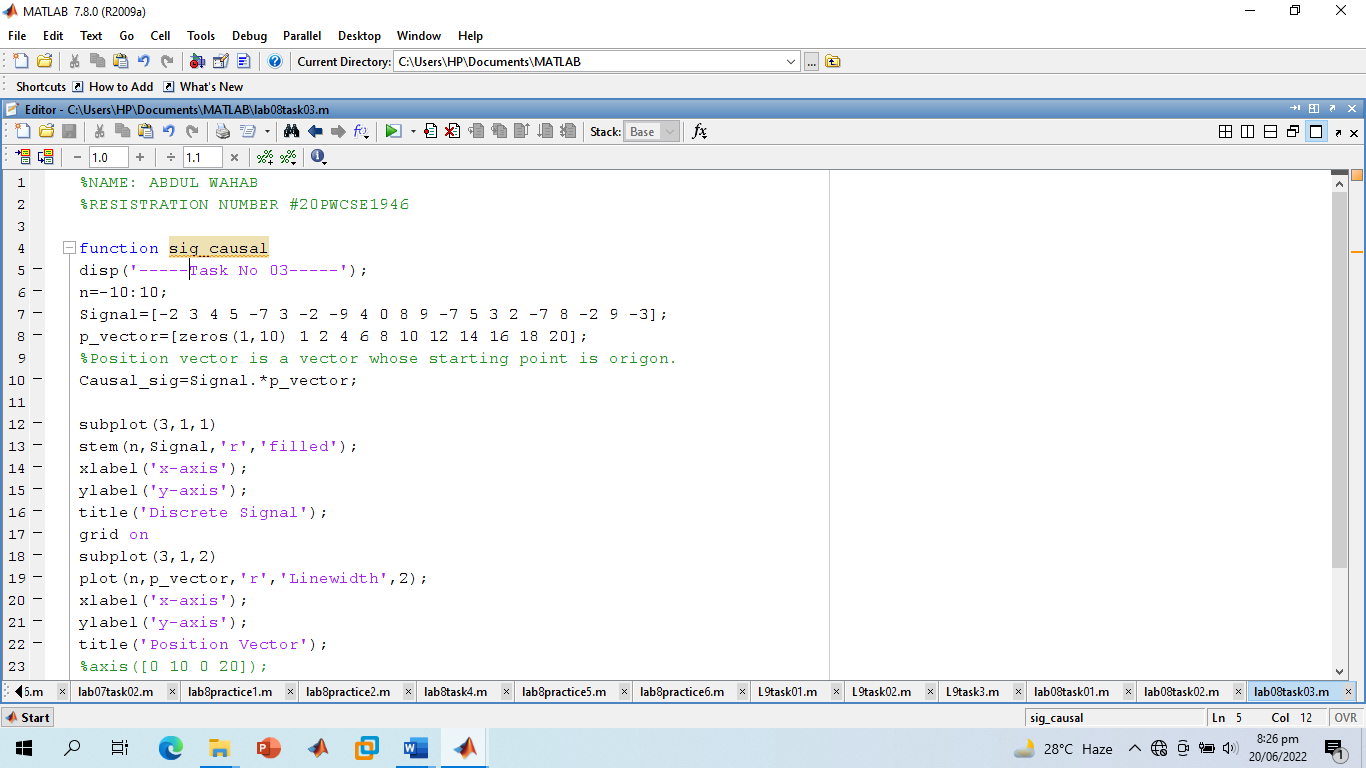
**Screenshot Output:**

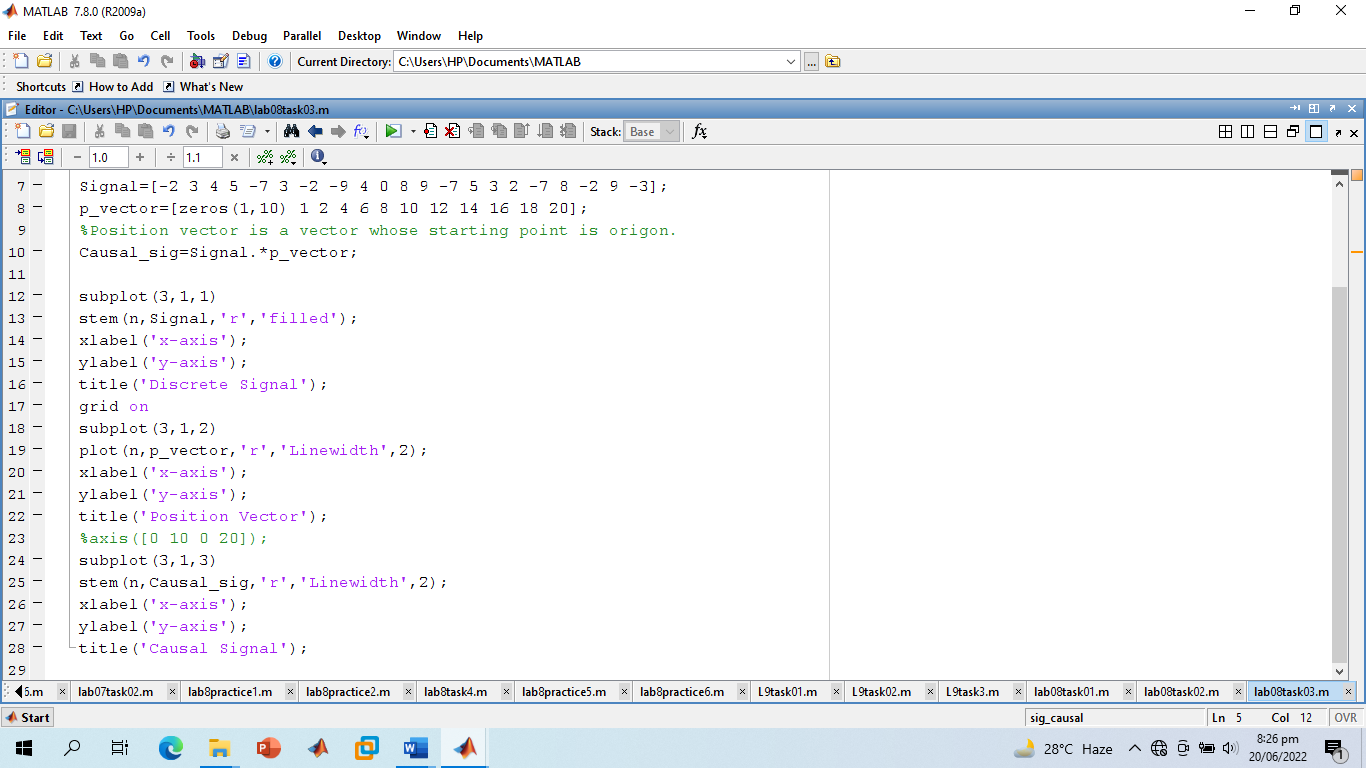


**-------------------------TASK 03--------------------------**

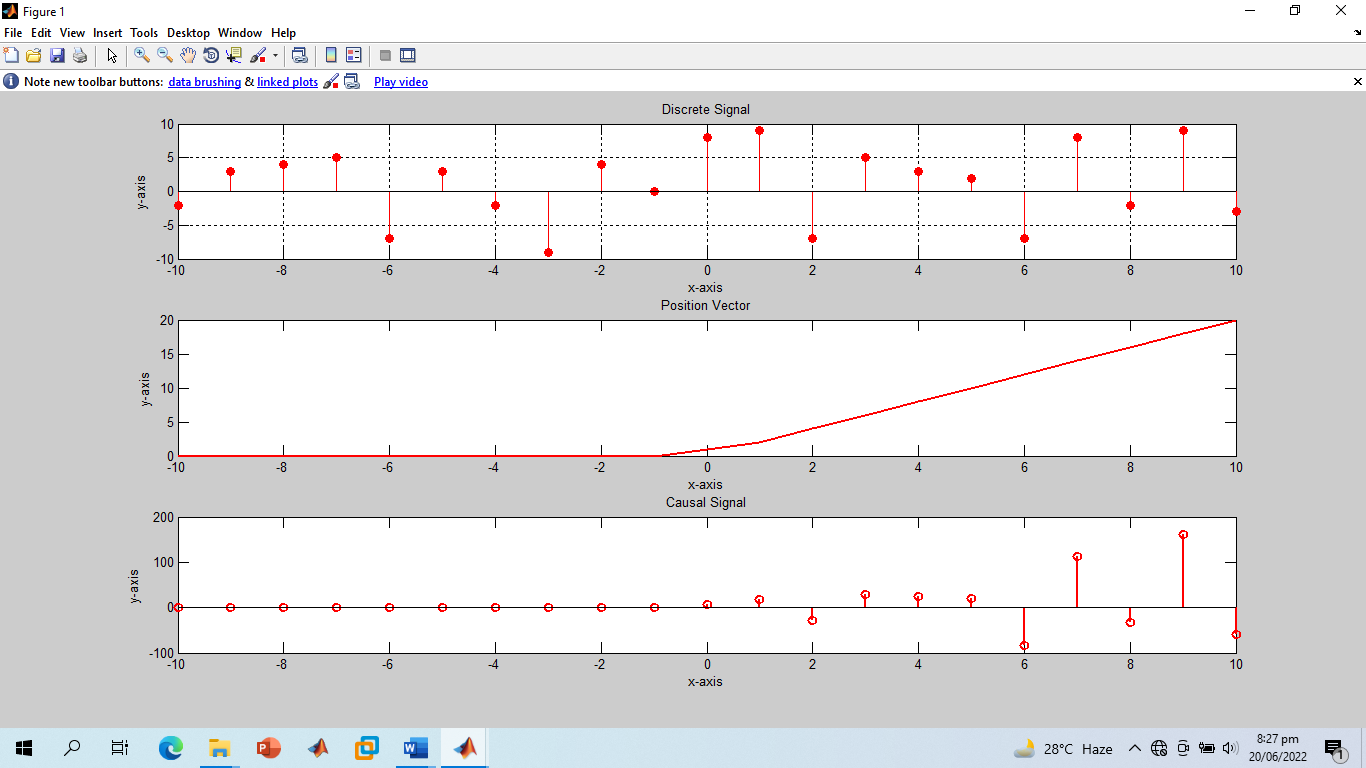
* Create a function by name of **sig\_causal** in matlab that has two input arguments: (i) a discrete‐time signal, and (ii) a position vector. The function should make the given signal causal and return the resultant signal to the calling program.

**Screenshot of Input:**





**Screenshot of Output:**



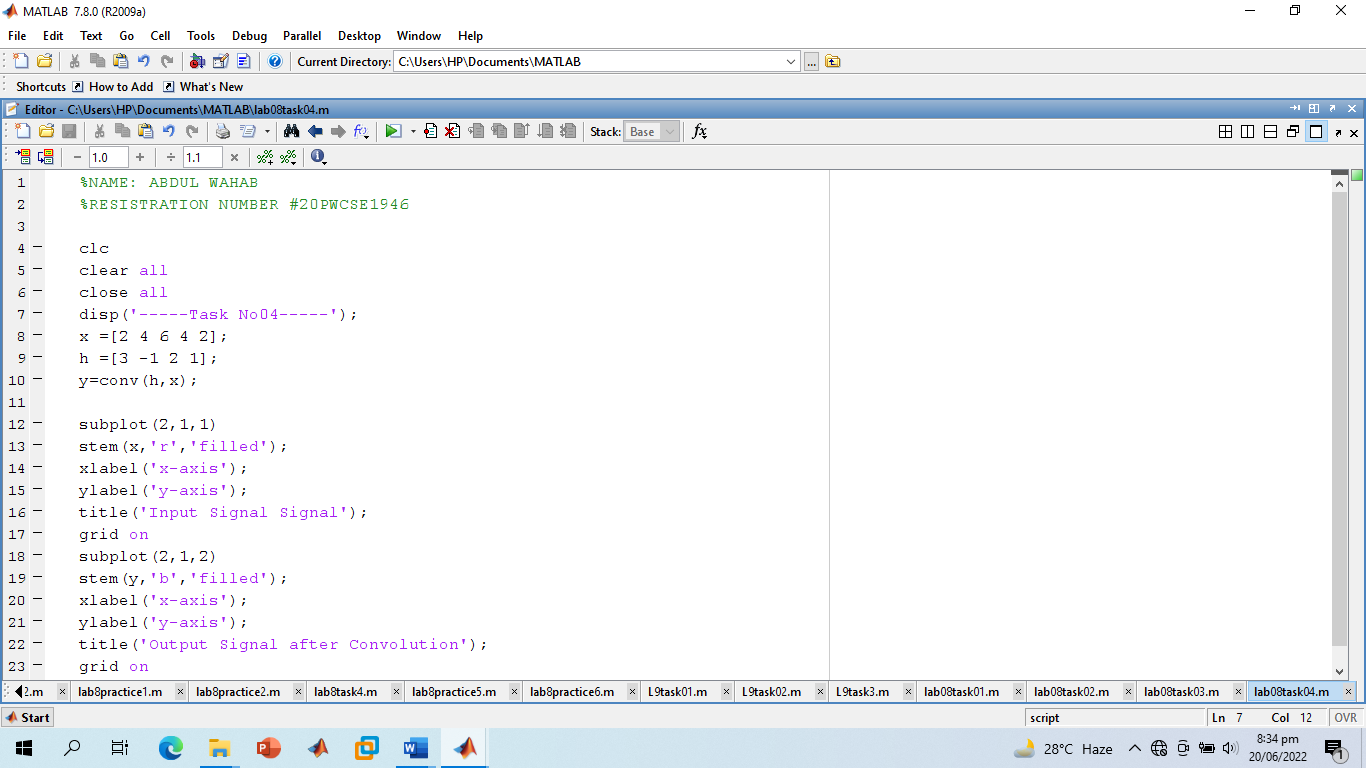
**-------------------------TASK 04--------------------------**

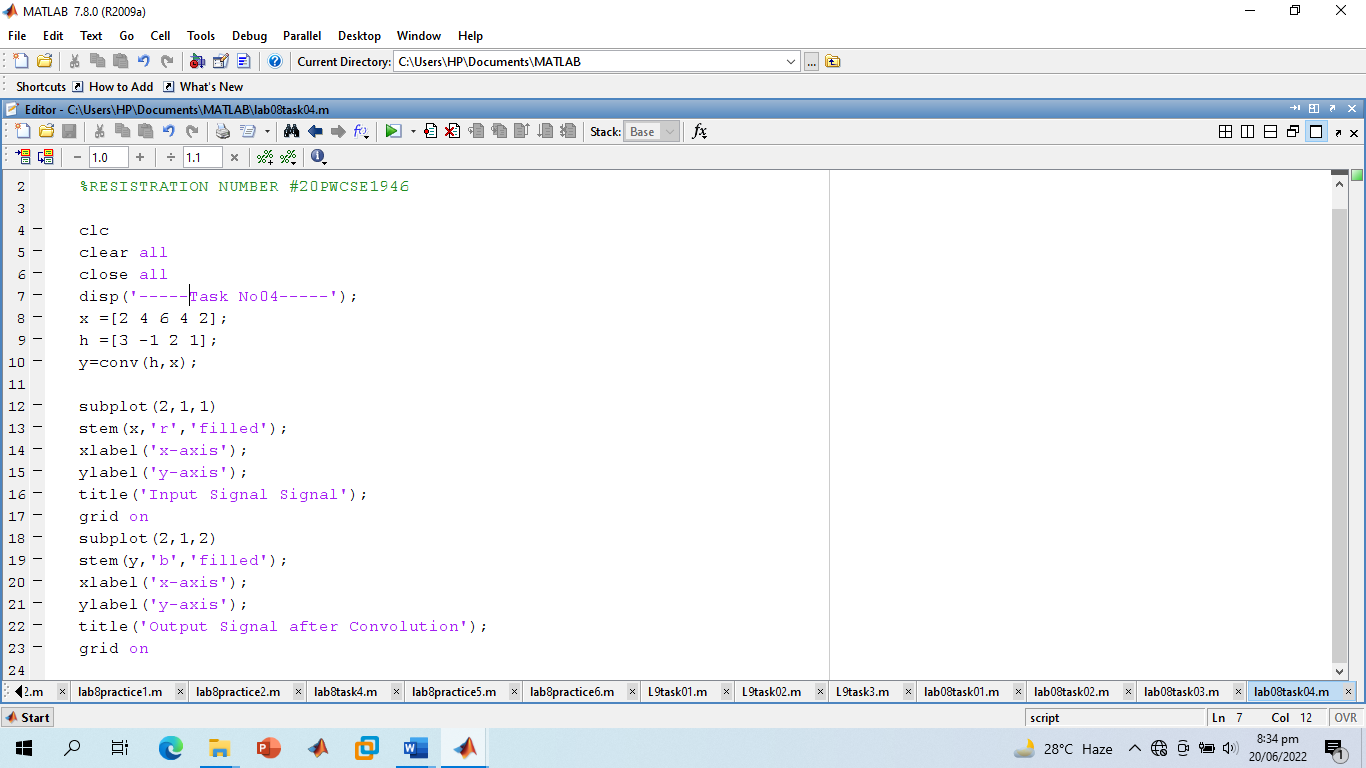
* Convolve the following signals: x = [2 4 6 4 2];

h = [3 ‐1 2 1];

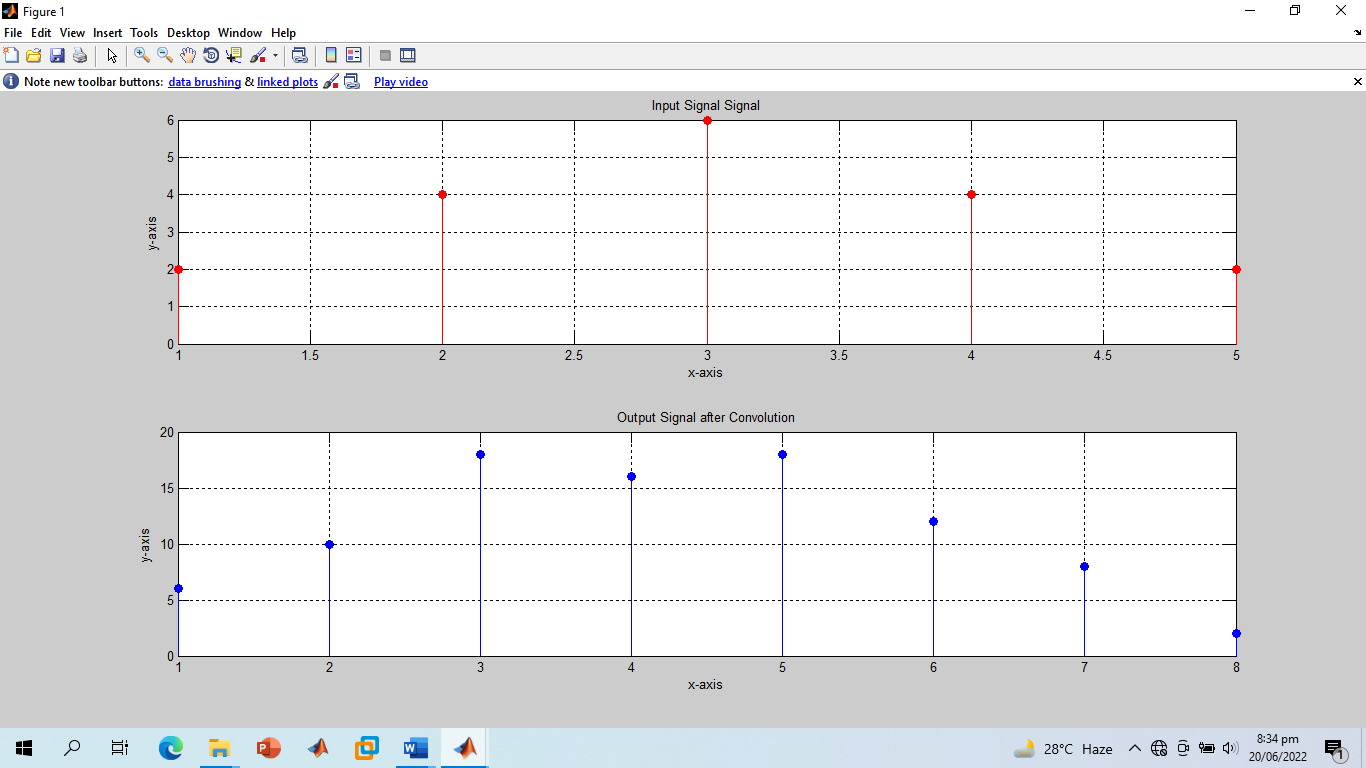
Plot the input signal as well as the output signal.

**Screenshot of Input:**





**Screenshot of Output:**



**-------------------------TASK 05--------------------------**

* Convolution is associative. Given the three signal x1[n], x2[n], and x3[n] as:

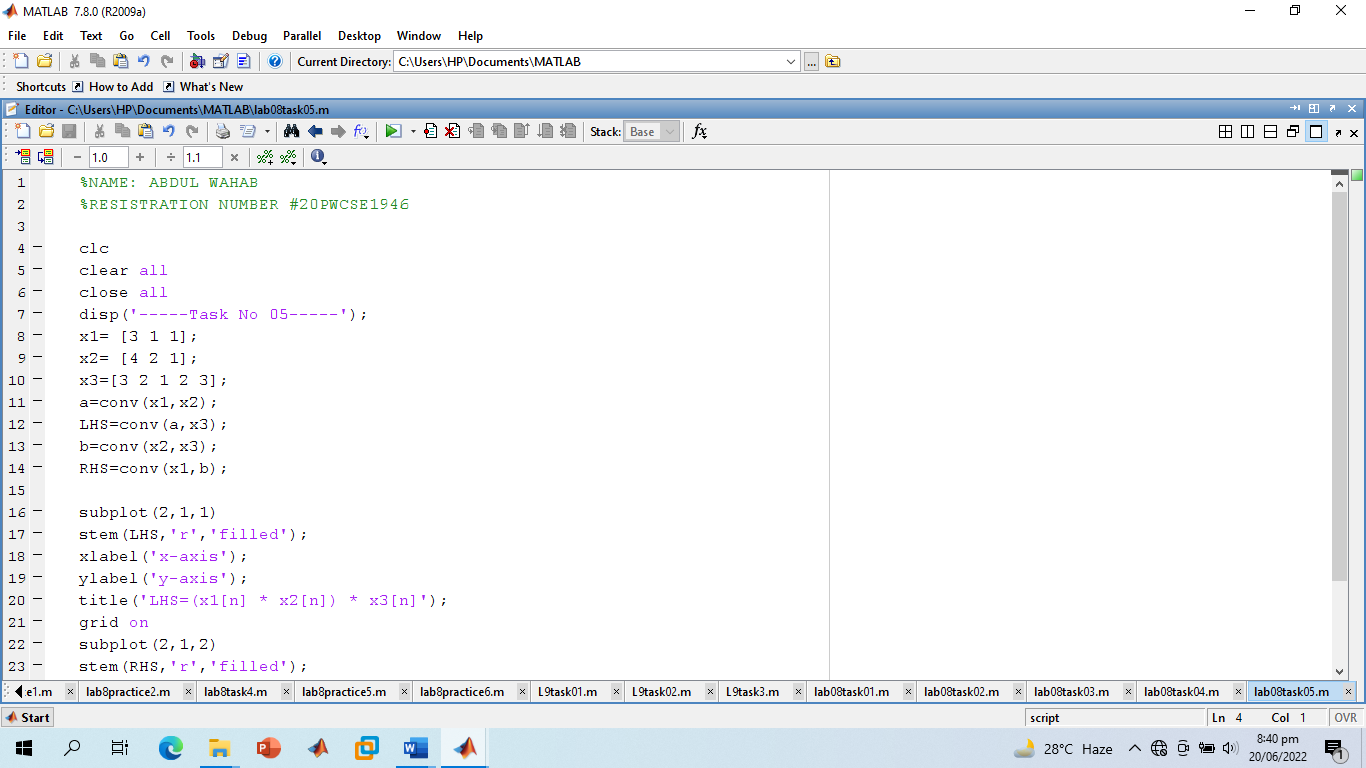
x1[n]= [3 1 1]

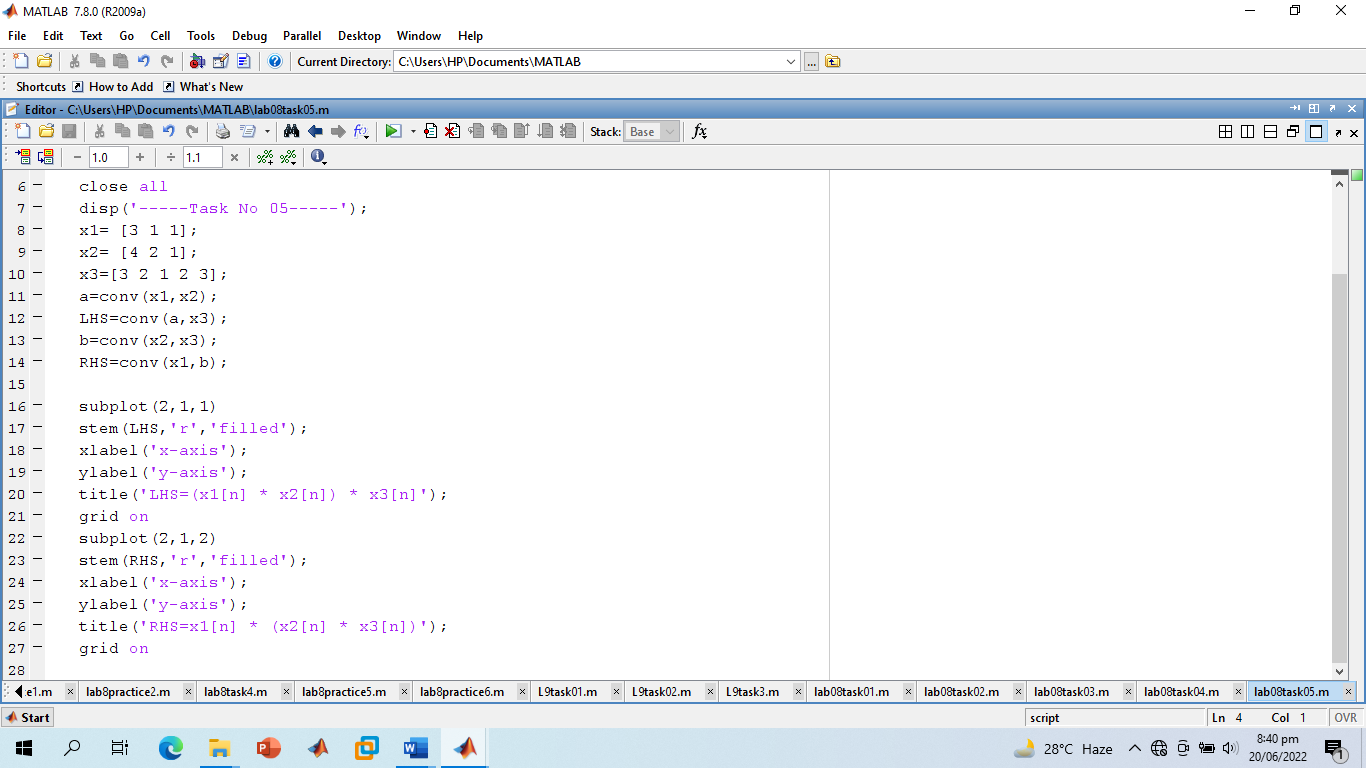
x2[n]= [4 2 1]

x3[n]= [3 2 1 2 3]

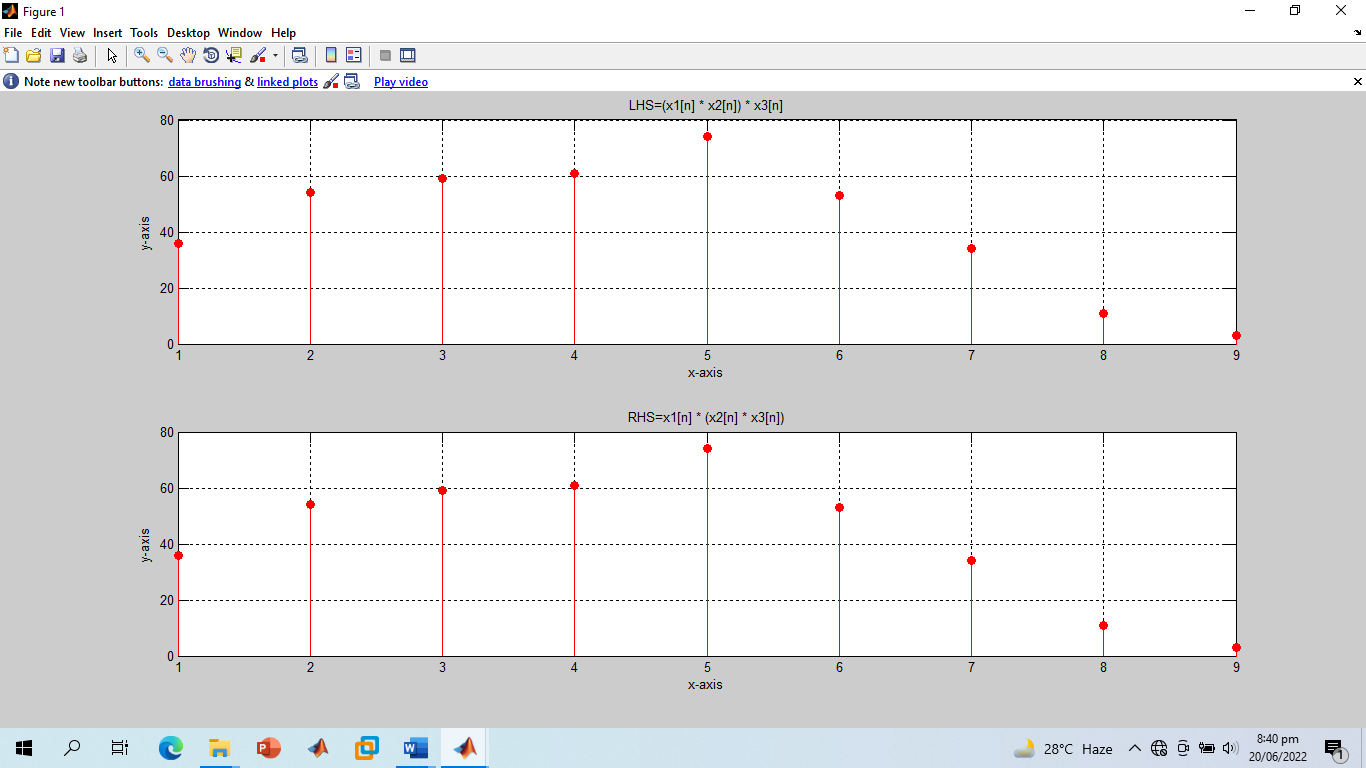
Show that (x1[n] \* x2[n]) \* x3[n] = x1[n] \* (x2[n] \* x3[n]).

**Screenshot of Input:**





**Screenshot of Output:**



**-------------------------TASK 06--------------------------**

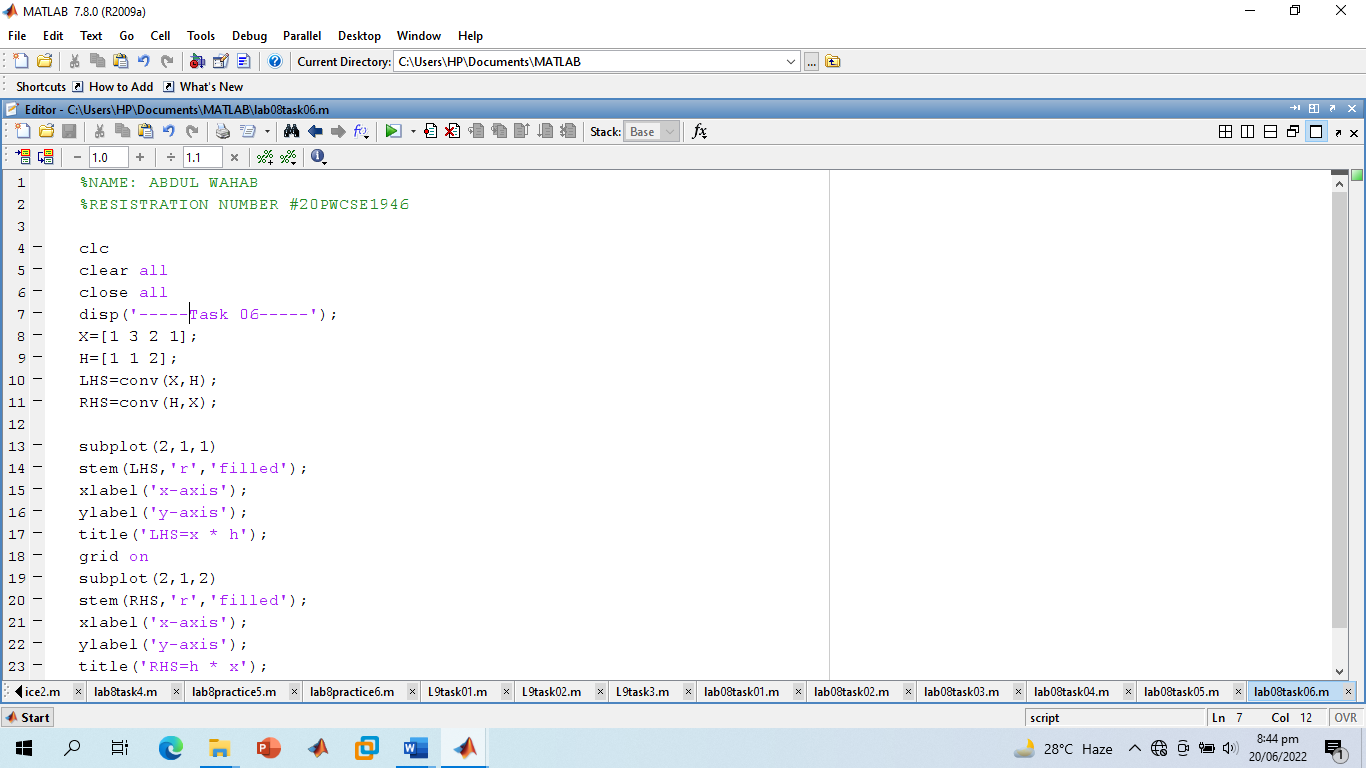
* Convolution is commutative. Given x[n] and h[n] as:

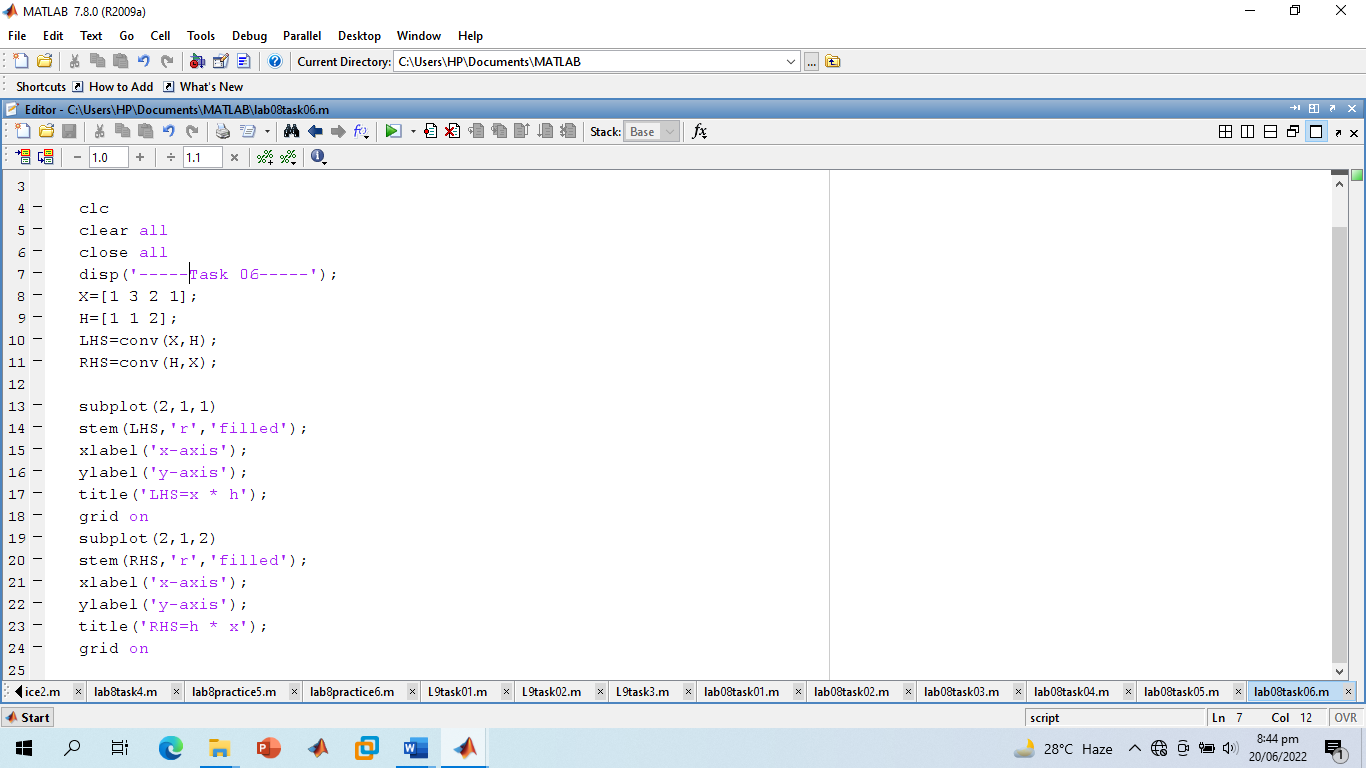
X[n]= [1 3 2 1]

H[n]= [1 1 2]

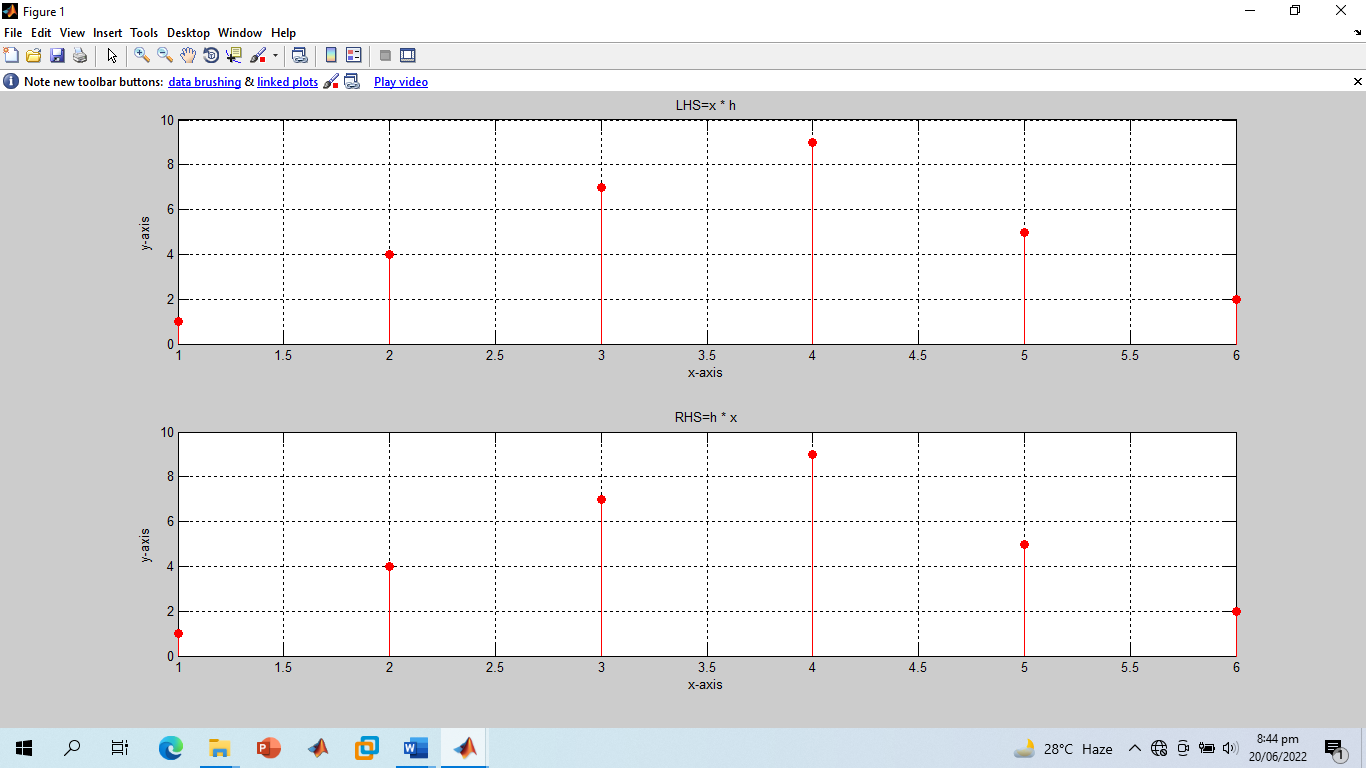
Show that x[n] \* h[n] = h[n] \* x[n].

**Screenshot of Input:**





**Screenshot of Output:**



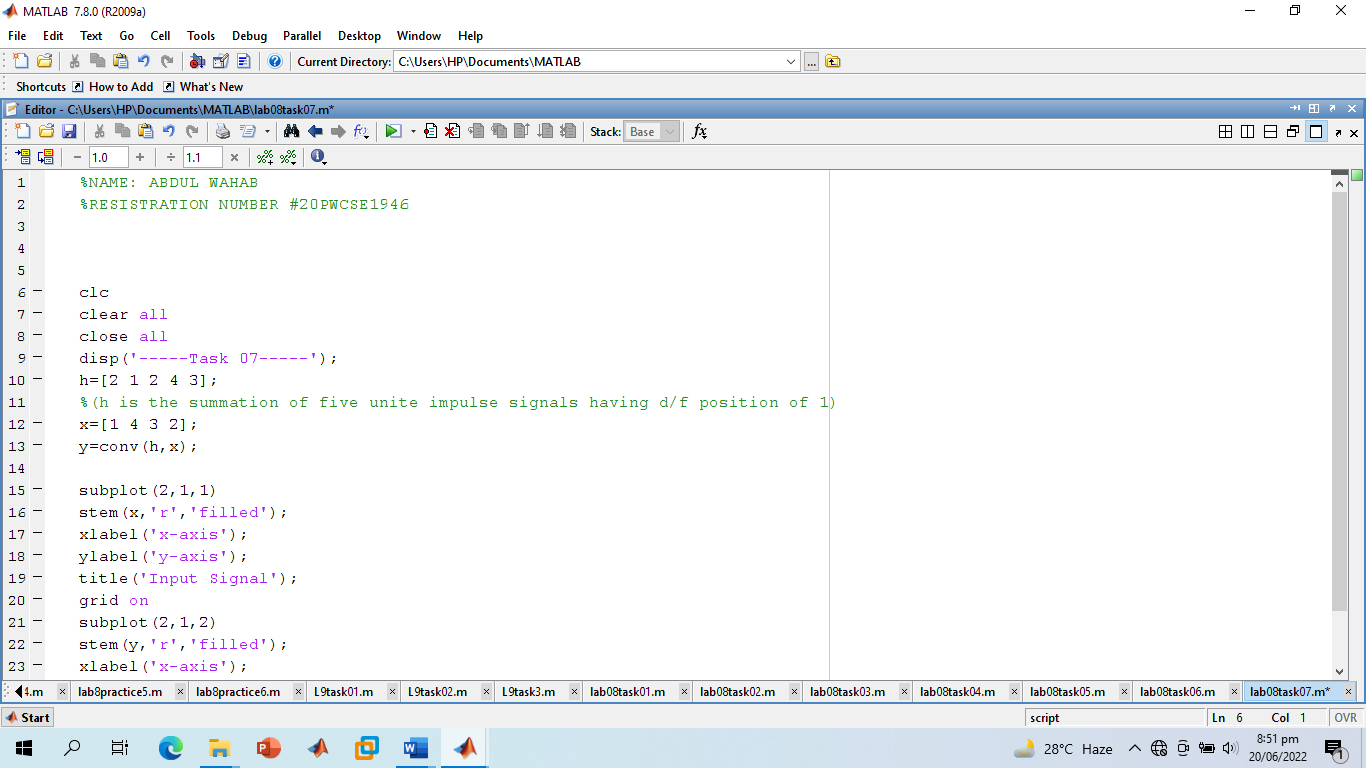
**-------------------------TASK 07--------------------------**

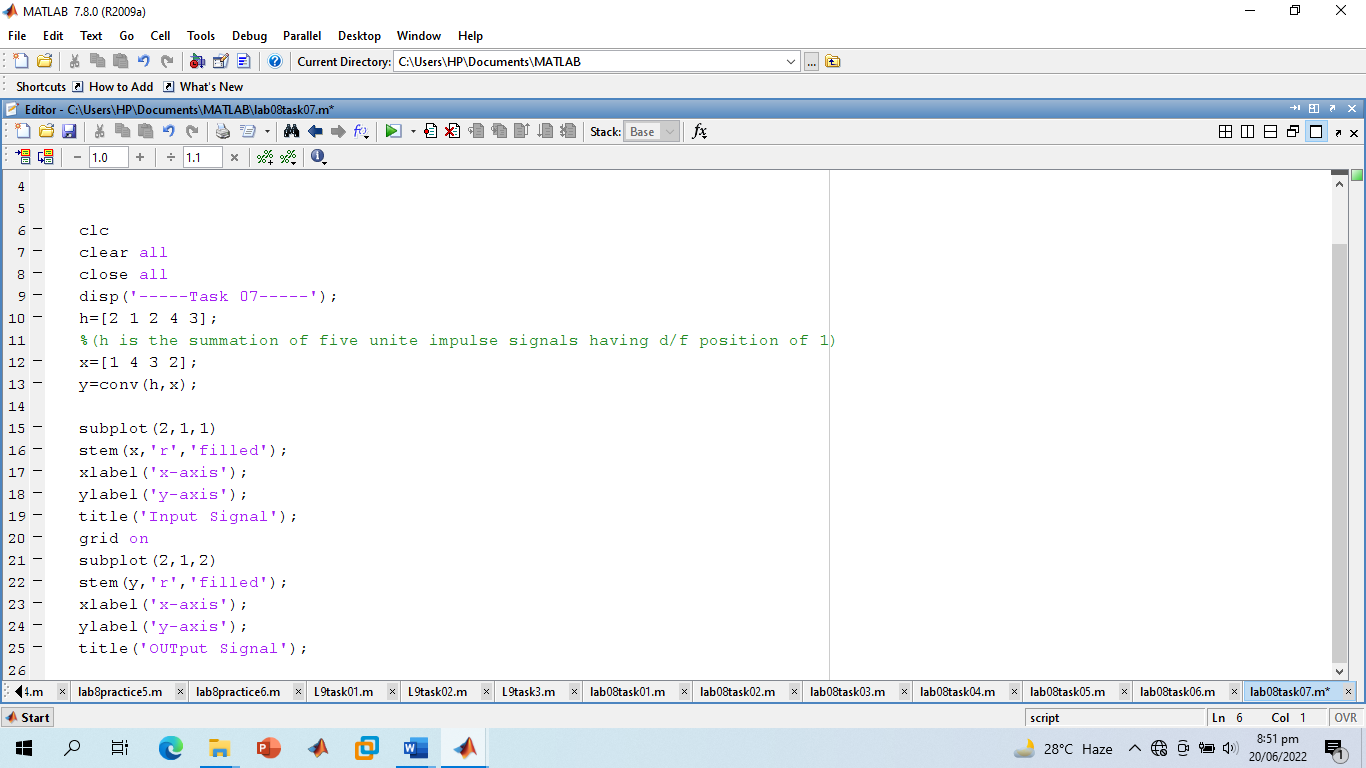
* Given the impulse response of the systems as:

h[n]= 2δ[n] + δ[n‐1]+ 2δ[n‐2]+ 4δ[n‐3]+ 3δ[n‐4]

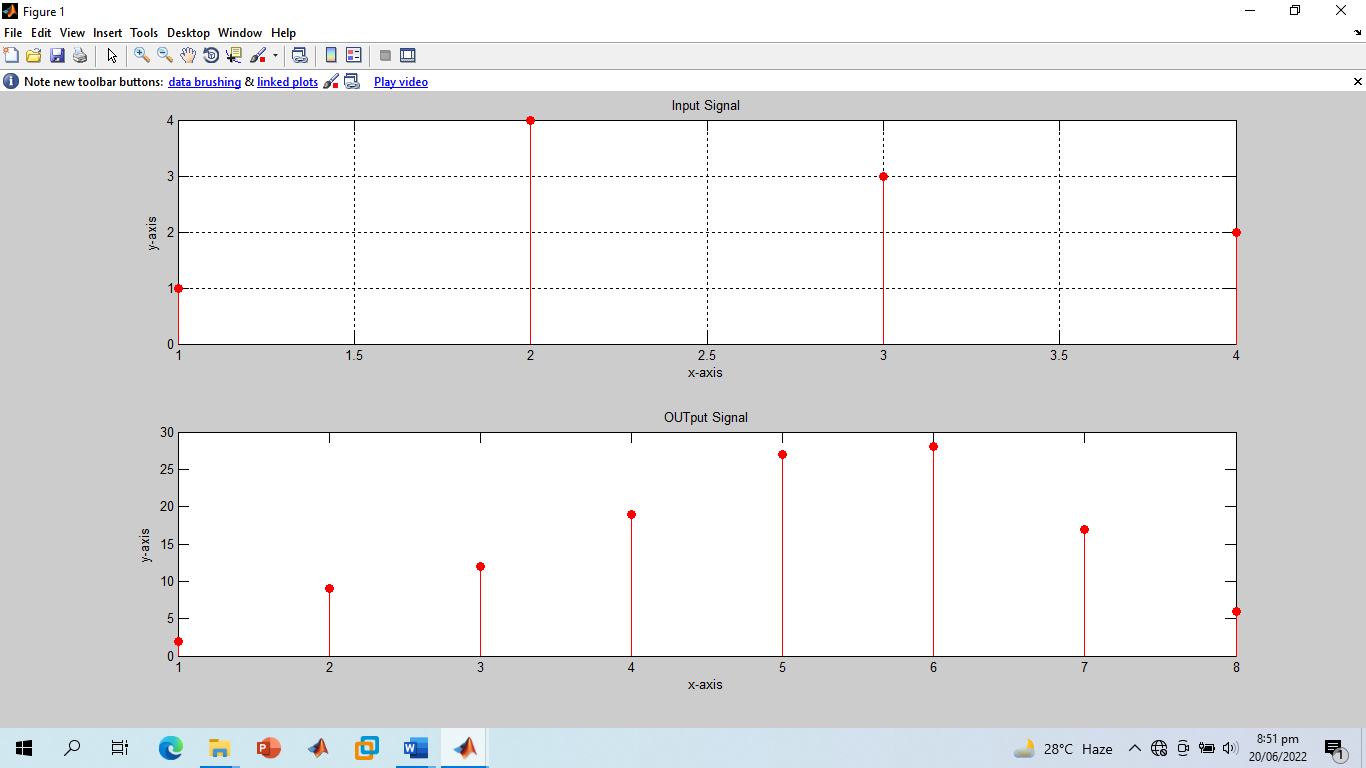
If the input x[n] = δ[n]+ 4δ[n‐1] +3δ[n‐2] + 2δ[n‐3] is applied to the system, determine the output of the system.

**Screenshot of Input:**



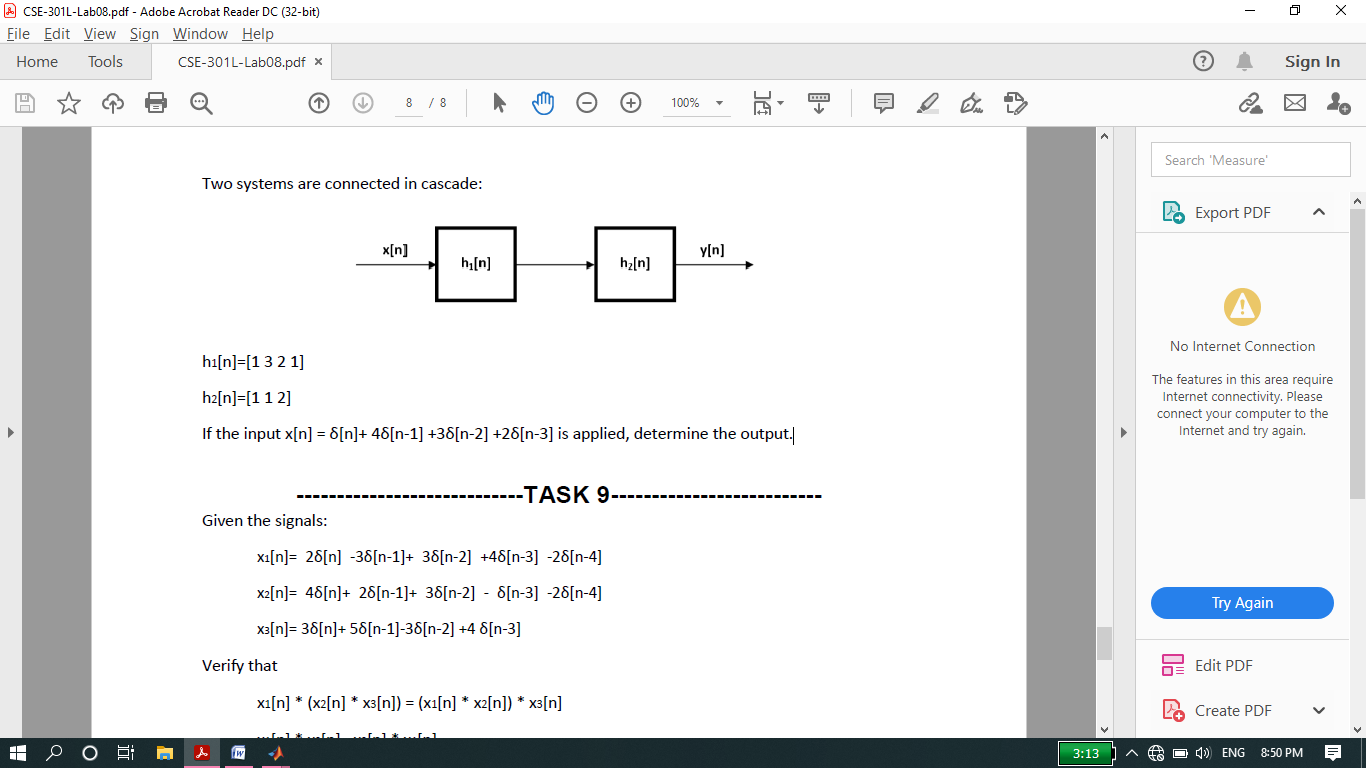


**Screenshot of Output:**



**-------------------------TASK 08--------------------------**

* Two systems are connected in cascade:

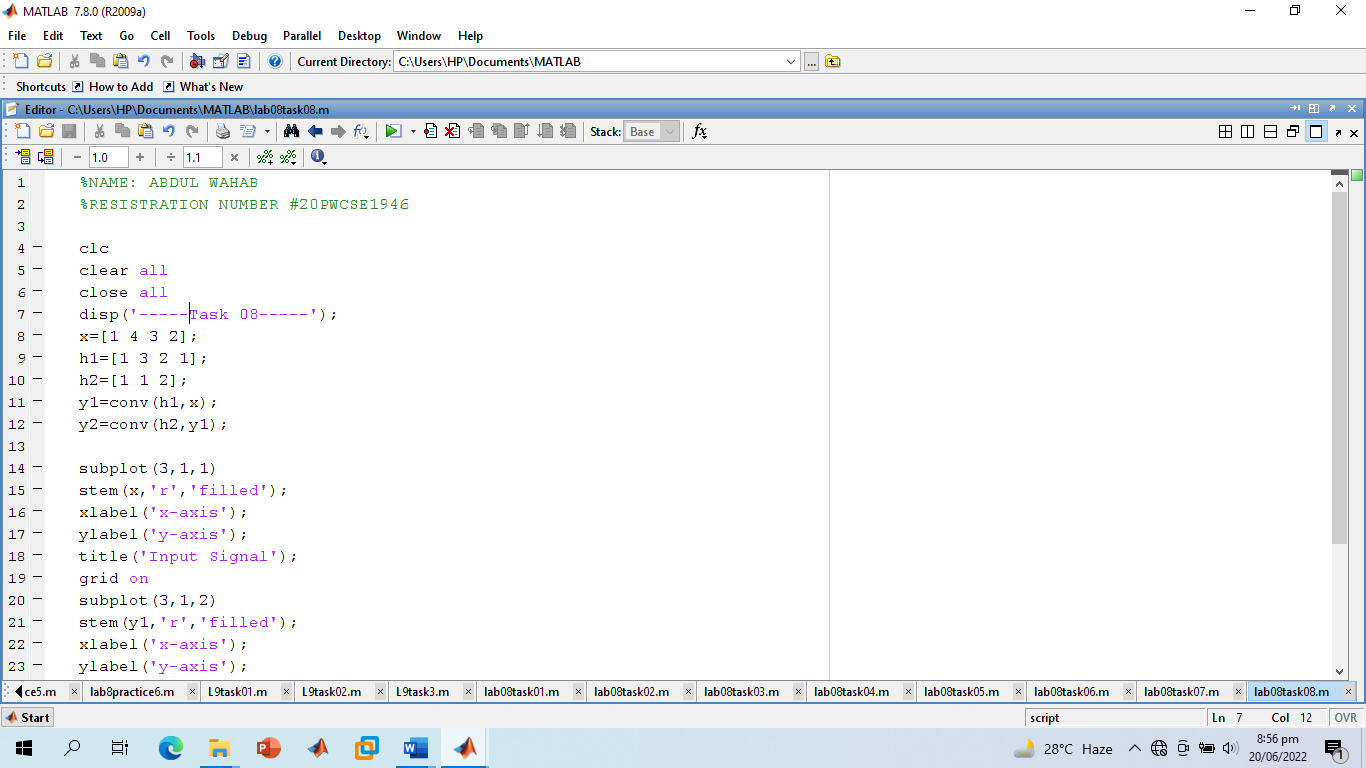


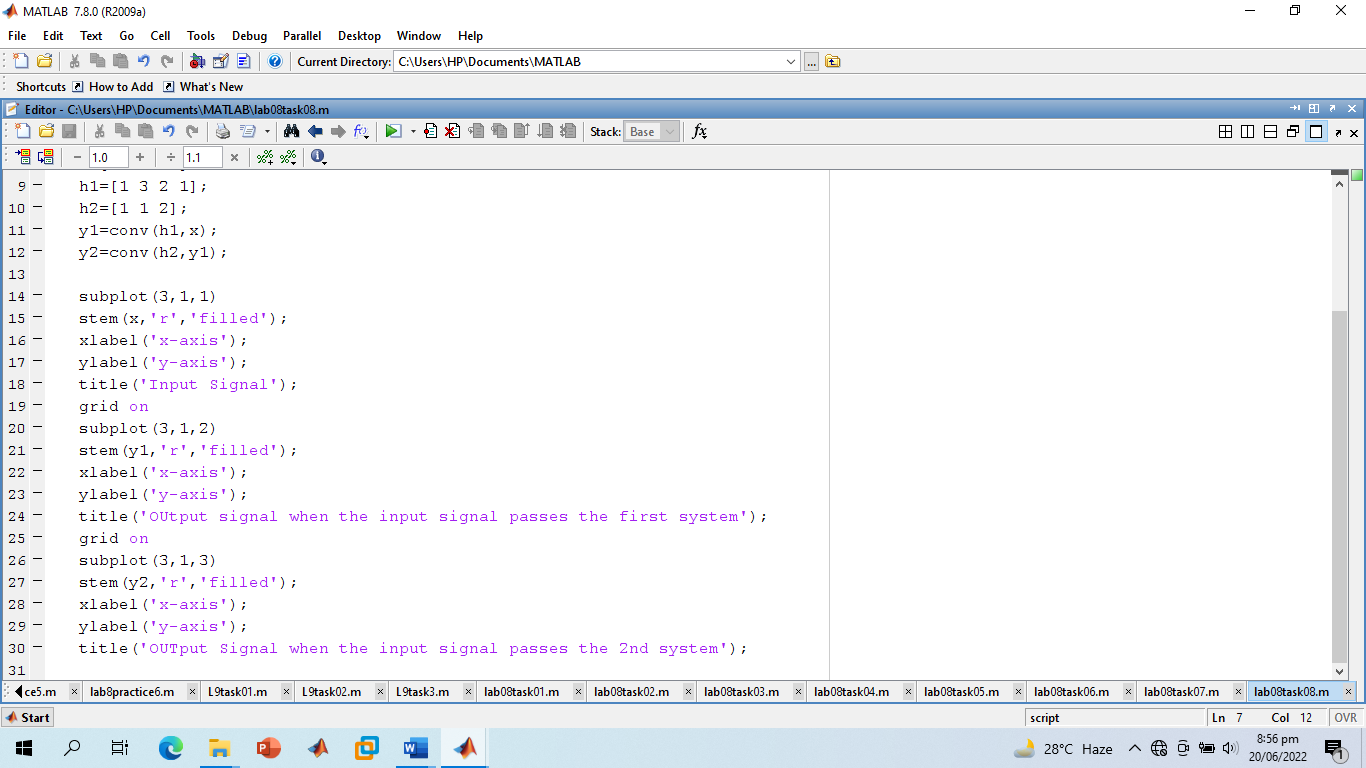
h1[n]= [1 3 2 1]

h2[n]= [1 1 2]

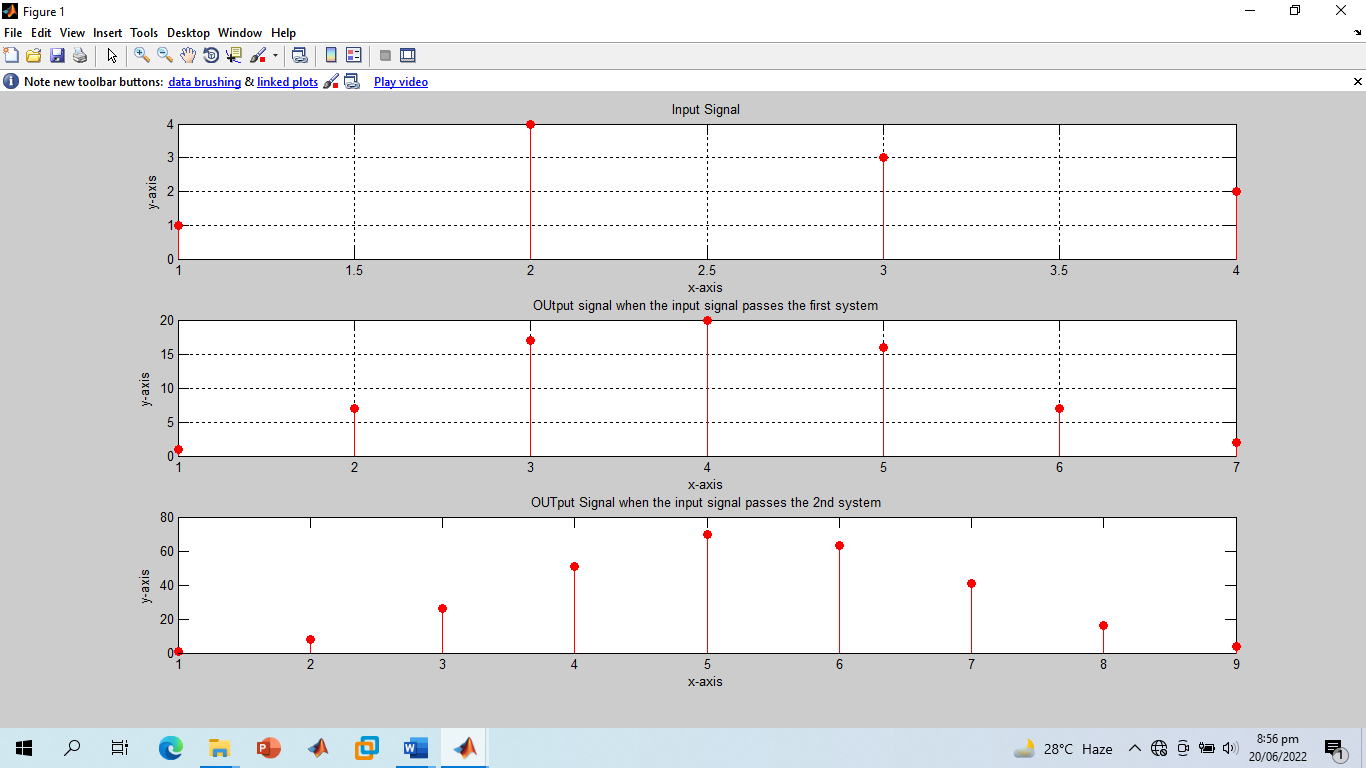
If the input x[n] = δ[n]+ 4δ[n‐1] +3δ[n‐2] +2δ[n‐3] is applied, determine the output.

**Screenshot of Input:**





**Screenshot of Output:**



**-------------------------TASK 09--------------------------**

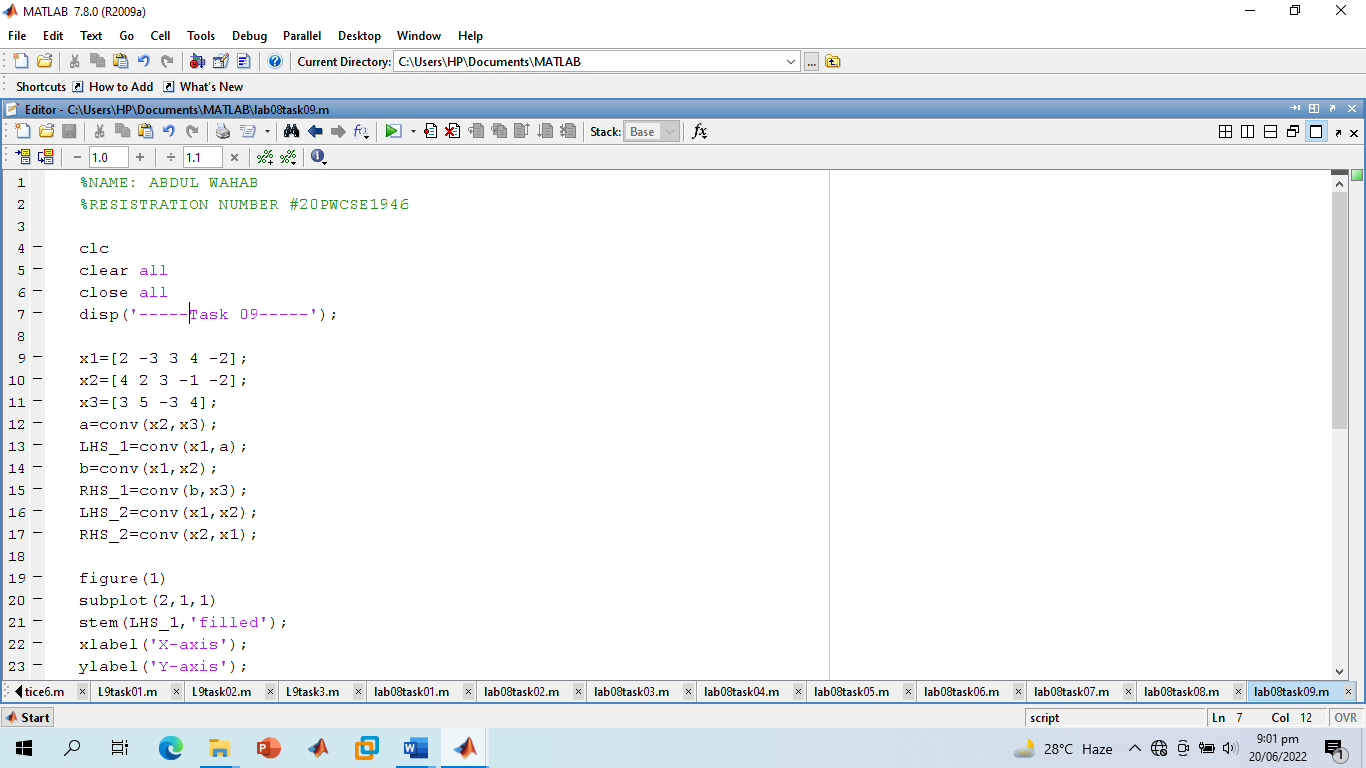
Given the signals:

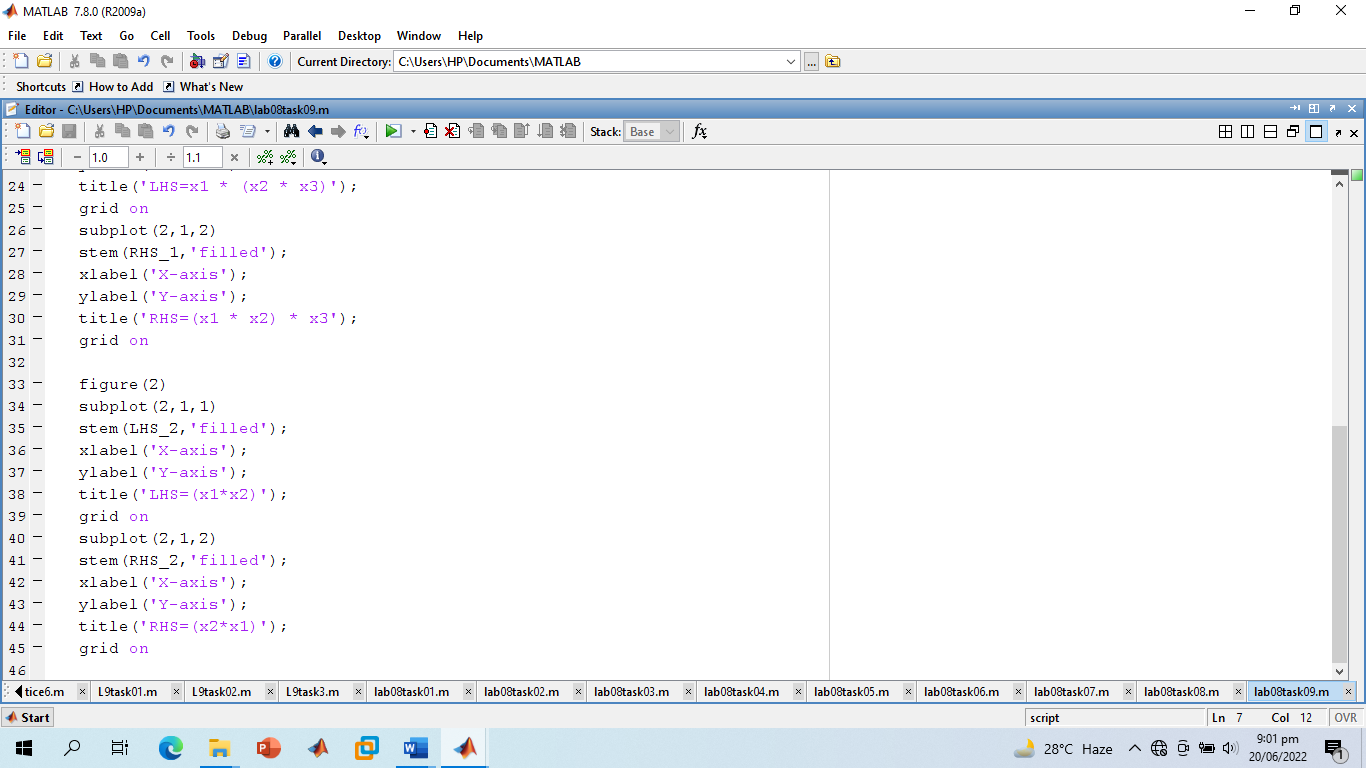
* x1[n]= 2δ[n] ‐3δ[n‐1]+ 3δ[n‐2] +4δ[n‐3] ‐2δ[n‐4]
* x2[n]= 4δ[n]+ 2δ[n‐1]+ 3δ[n‐2] ‐ δ[n‐3] ‐2δ[n‐4]
* x3[n]= 3δ[n]+ 5δ[n‐1]‐3δ[n‐2] +4 δ[n‐3]

Verify that:

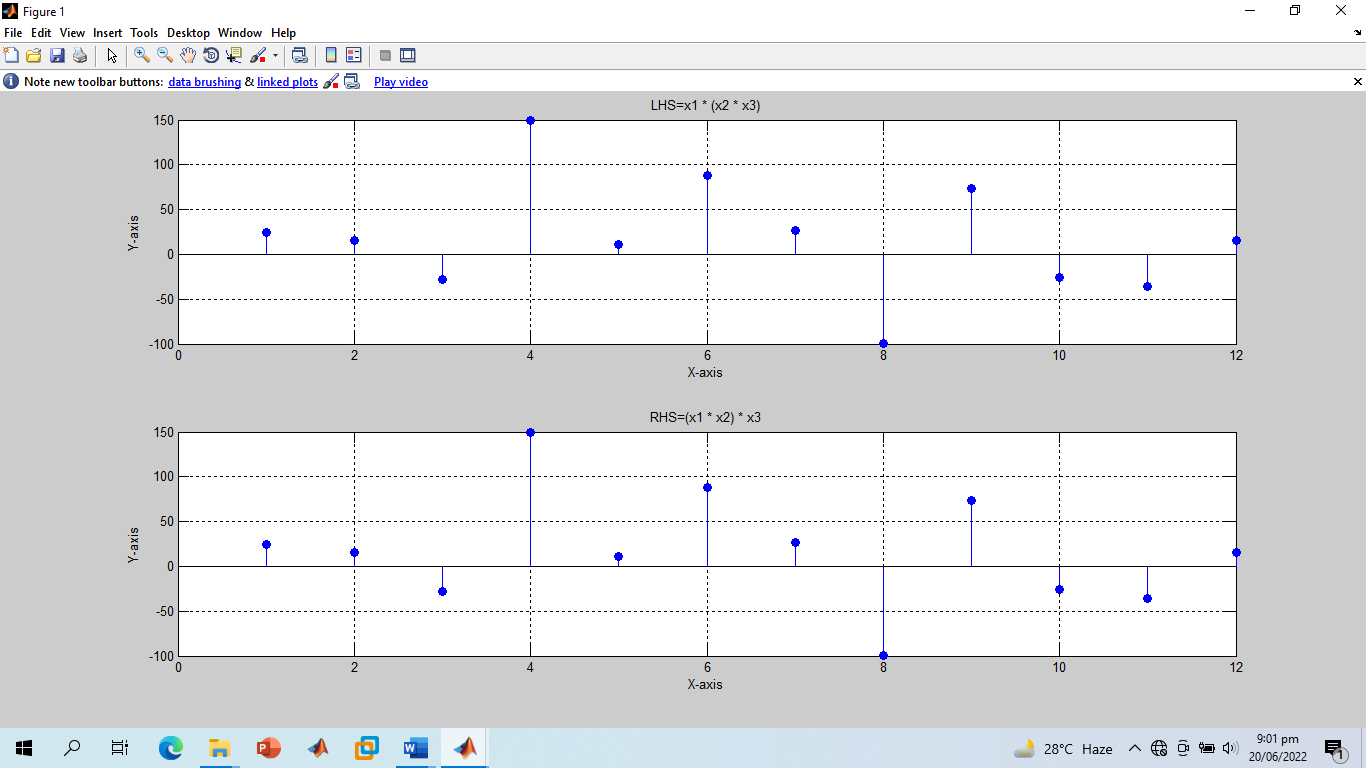
* x1[n] \* (x2[n] \* x3[n]) = (x1[n] \* x2[n]) \* x3[n]
* x1[n] \* x2[n]= x2[n] \* x1[n]

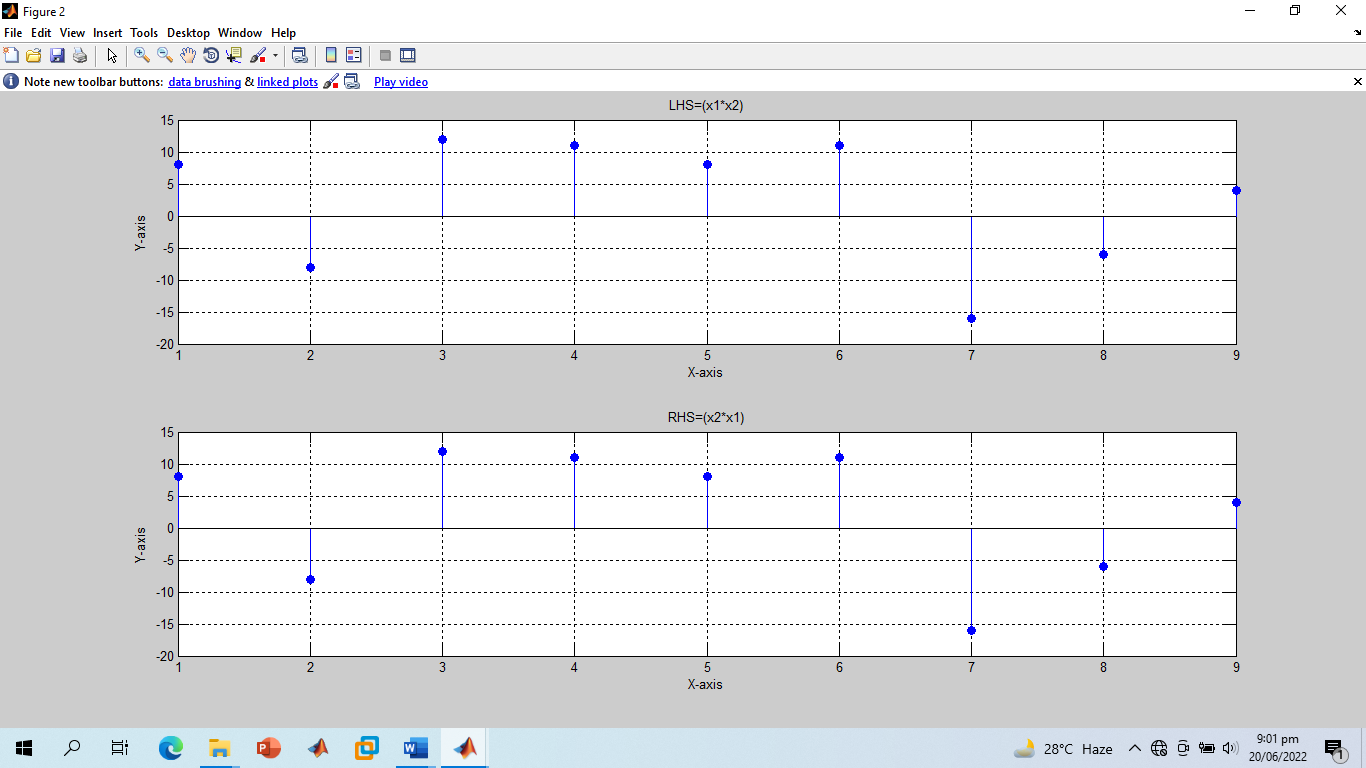
**Screenshot of Input:**





**Screenshot of Output:**





**-------------------------THE END-------------------------**