**TITLE OF LAB: (INTRODUCTION TO FUNCTIONS, LOOPS, CONDITIONAL AND RELATIONAL OPERATORS)**

**LAB # 03**



**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 Friday, April 22th, 2022)

**Department of Computer Systems Engineering**

**University of Engineering and Technology, Peshawar**

**OBJECTIVES OF THE LAB:**

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

**In this lab, we will get an understanding of the following topics:**

* Making Functions
* Control Structures
* Relational Constructs
* Logical Constructs
* Branching Constructs
* Looping constructs

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

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

* Write a function that accepts temperature in degrees F and computes the corresponding value in degrees C.

**Source Code:**

function[D]= temperature(f)

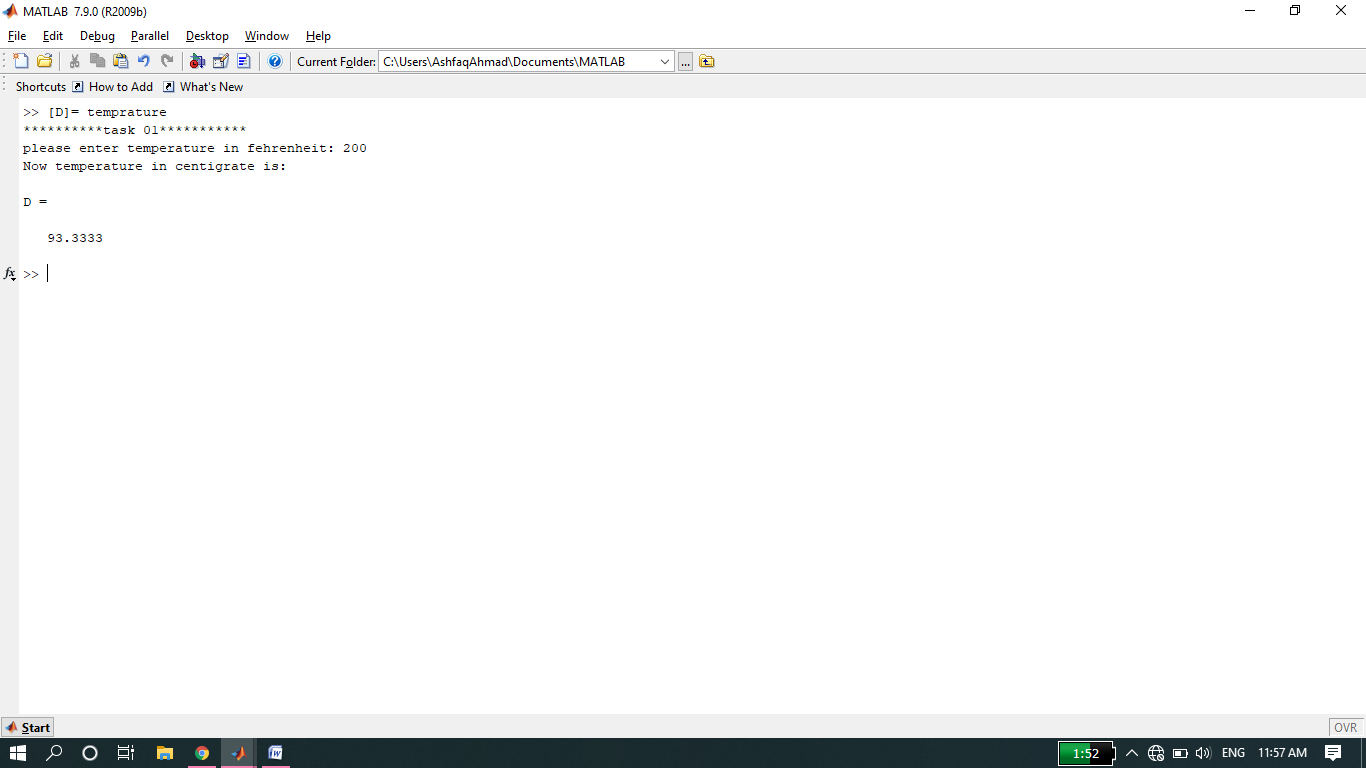
disp('\*\*\*\*\*\*\*\*\*\*task 01\*\*\*\*\*\*\*\*\*\*\*');

f=input ('please enter temperature in Fahrenheit: ');

D=5/9\*(f-32);

disp('Now temperature in centigrade is: ');

**Screenshot of Output:**



**Note:**

* If we write the function like ‘function temperature (f)’ in editor window then we will write disp(D) at the end.
* we can also write above function like ‘function[D]=temperature’.
* We can run above function by click on run button or f5.
* If we write the word temperature or in command window then output will be assigned to variable ans.
* If we write ‘[D or any other letter] =function’ the output will be assigned to required letter.

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

* For the arrays x and y given below, write matlab code to find all the elements in x that are greater than the corresponding elements in y.
  + x = [‐3, 0, 0, 2, 6, 8] y = [‐5, ‐2, 0, 3, 4, 10]

**Source Code:**

clc

clear all

close all

x = [-3,0,0,2,6,8];

y = [-5,-2,0,3,4,10];

disp('the elements in vector x which are greater than elements in vector y at corresponding position are: ')

for i=1:1:6;

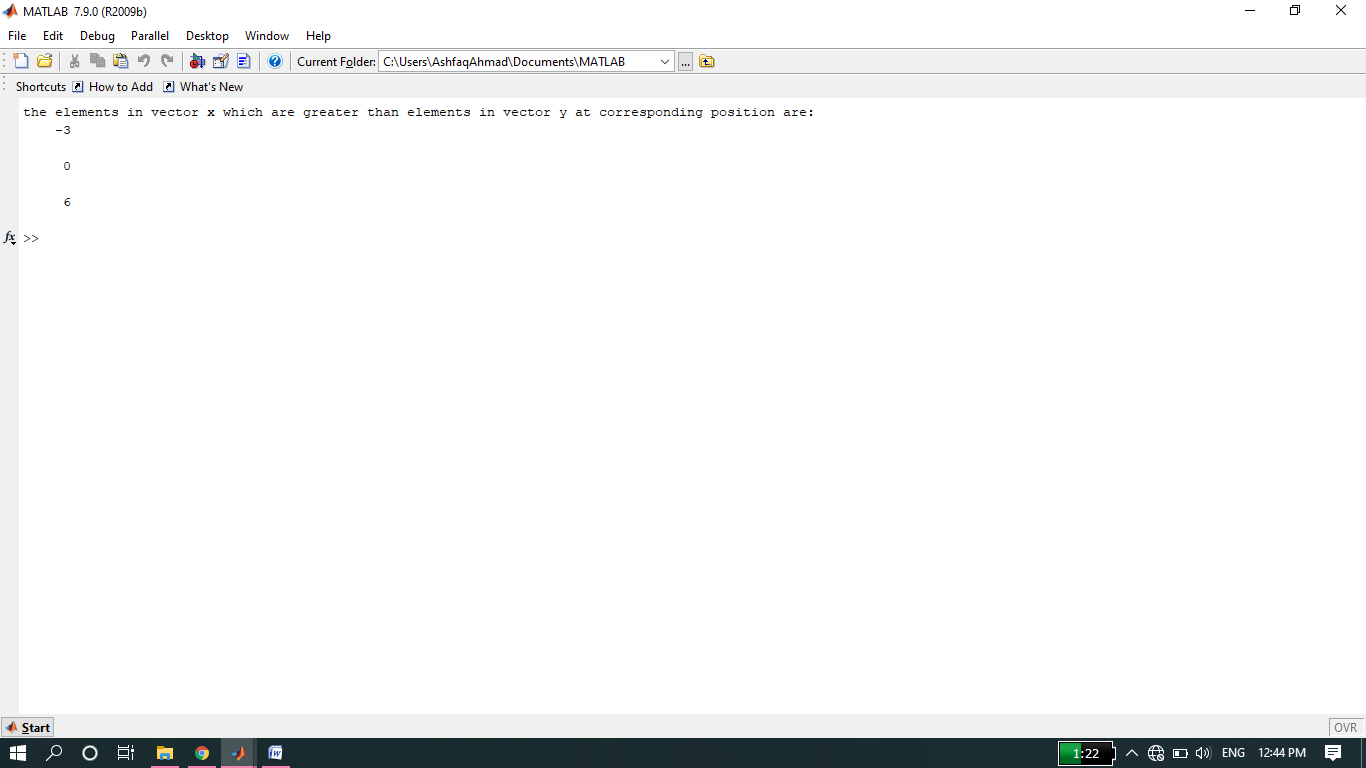
if x(i)>y(i)

disp(x(i));

end

end

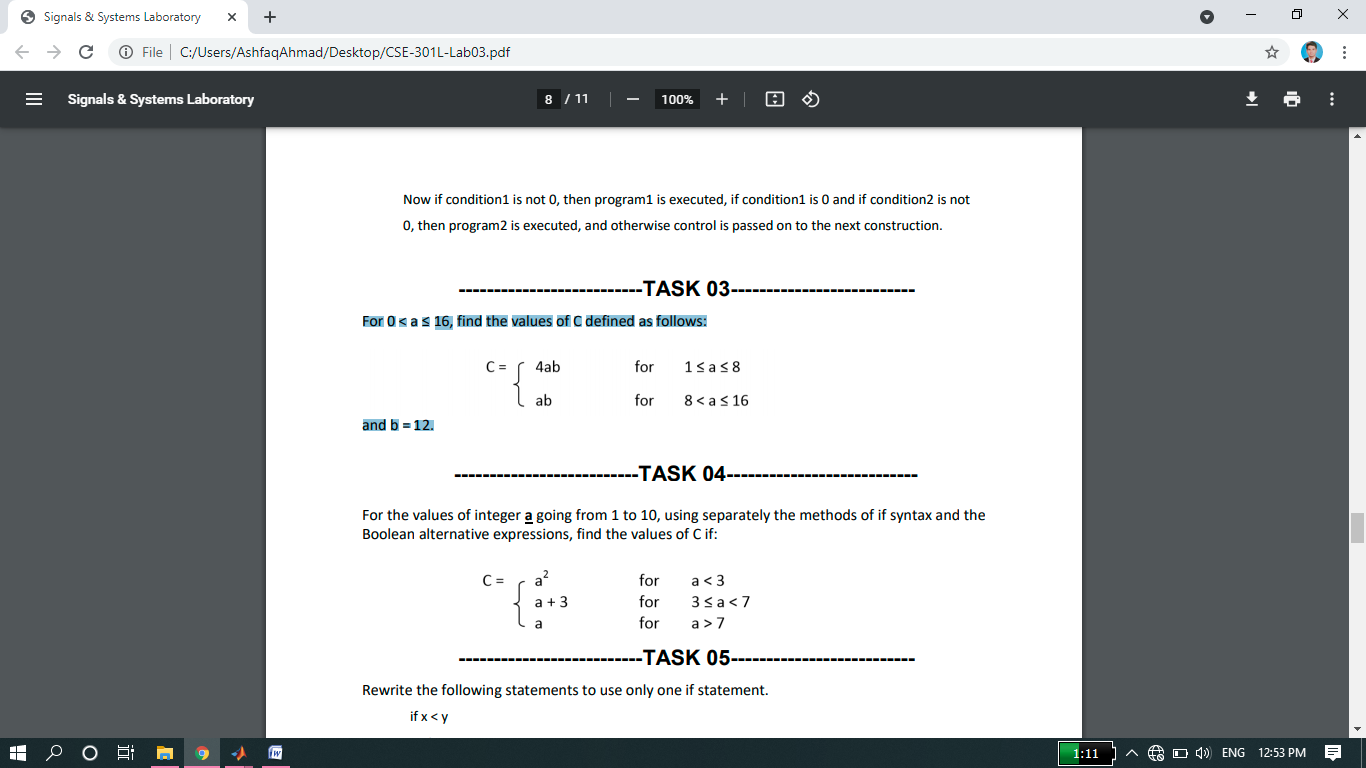
**Screenshot of Output:**



**Note:**

* In above coding for loop will run for six times because elements in each vector are 6. In each time it will check the condition x>y, if condition true it will display corresponding element.

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

* For 0 < a ≤ 16, find the values of C defined as follows:
  + 
  + and b = 12.

**Source Code:**

function product

disp('\*\*\*\*\*\*task 03\*\*\*\*\*\*');

b=12;

disp('All the values of c in the range 1<=a<=8 and b=12');

for a=1:1:8;

c=4\*a\*b;

disp(c);

end

disp('All the values of c in the range 8<a<=16 and b=12');

for a=9:1:16;

c=a\*b;

disp(c);

end

%we can also do it using if else statements.

disp('\*\*\*\*\*\*\*\*\*2nd method\*\*\*\*\*\*');

disp('All the values of c in the range 1<=a<=8 and b=12');

for a=1:1:16

if a>=1 &&a<=8;

disp(4\*a\*b);

else

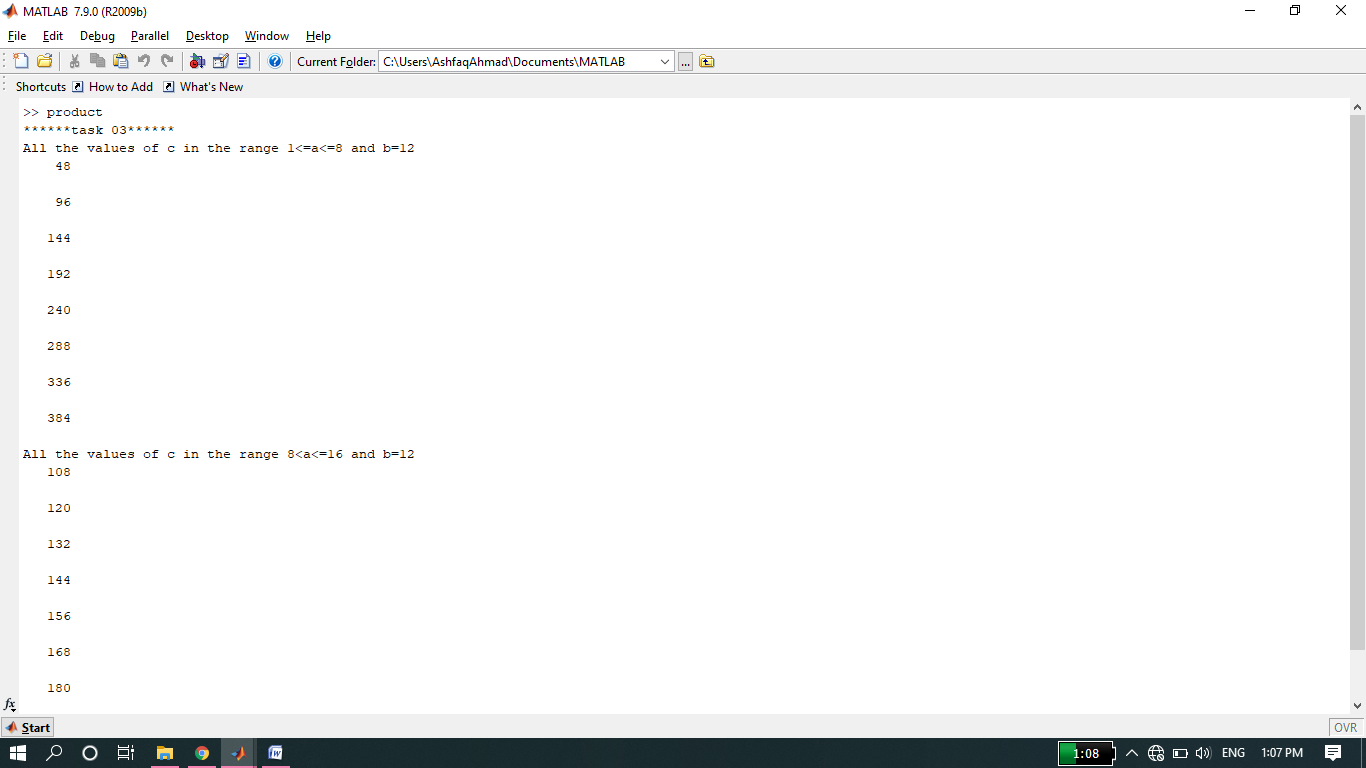
disp('All the values of c in the range 8<a<=16 and b=12');

disp(a\*b)

end

end

**Screenshot of Output:**

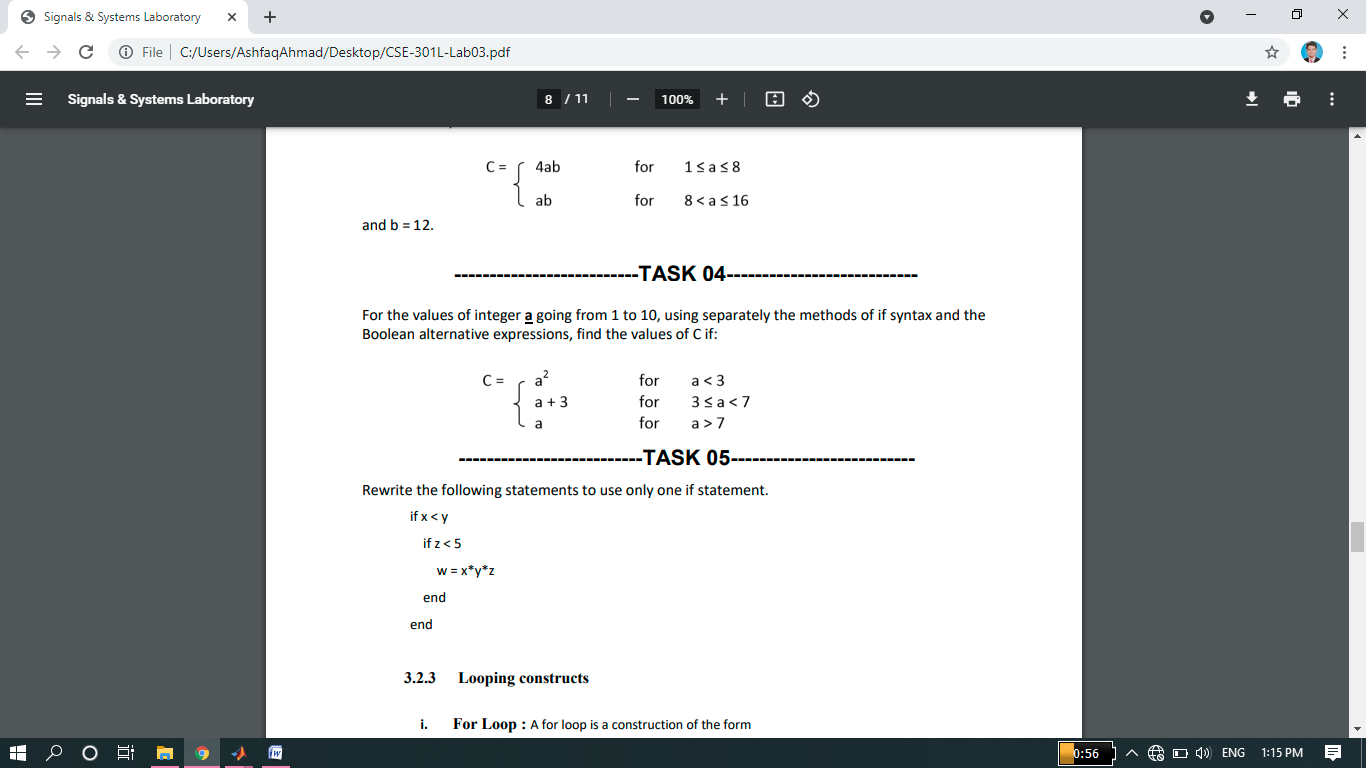


**Note:**

* Here we used for loop for the range of value a, then we write formula of c given in question and finally display all the values of c in given range.
* We can also do it using if else statement inside one for loop.

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

* For the values of integer, a going from 1 to 10, using separately the methods of if syntax and the Boolean alternative expressions, find the values of C if:

****

**Source Code:**

clc

clear all

close all

for a=1:1:10;

if a<3;

b=(a.^2);

disp('value of c in the range a<3: ');

disp(b);

else

if a>=3 && a<7;

disp('value of c in the range 3<=a<7: ');

c=(a+3);

disp(c);

else

if a>7;

disp('value of c in the range a>7: ');

c=(a);

disp(c);

end

end

end

end

**Second coding:**

clc

clear all

close all

disp('\*\*\*\*\*\*task 04\*\*\*\*\*\*');

for a=1:1:10;

if a<3;

b(a)=(a.^2);

else

if a>=3 && a<7;

c(a)=(a+3);

else

if a>7;

d(a)=(a);

end

end

end

end

disp('value of c in the range a<3: ');

disp('C= ');

disp(b);

disp('value of c in the range 3<=a<7: ');

disp('C= ');

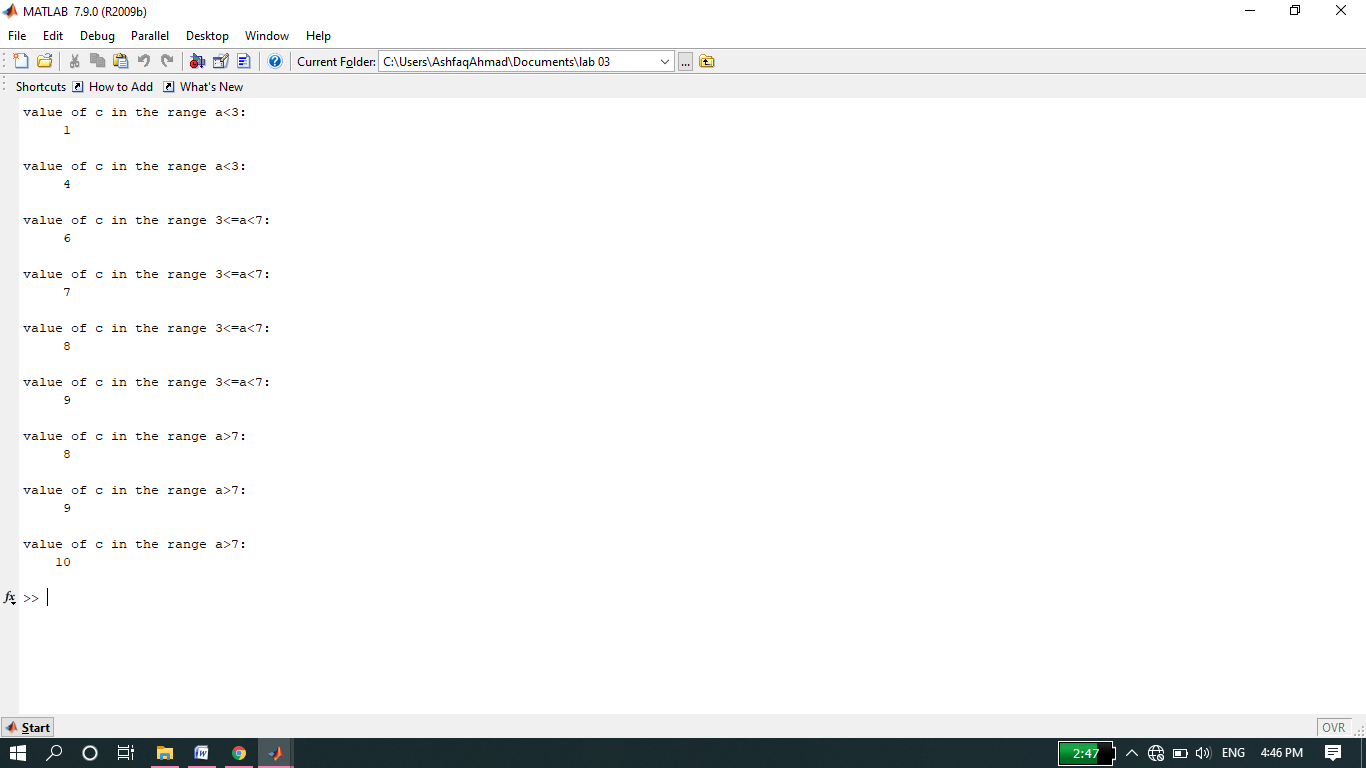
disp(c);

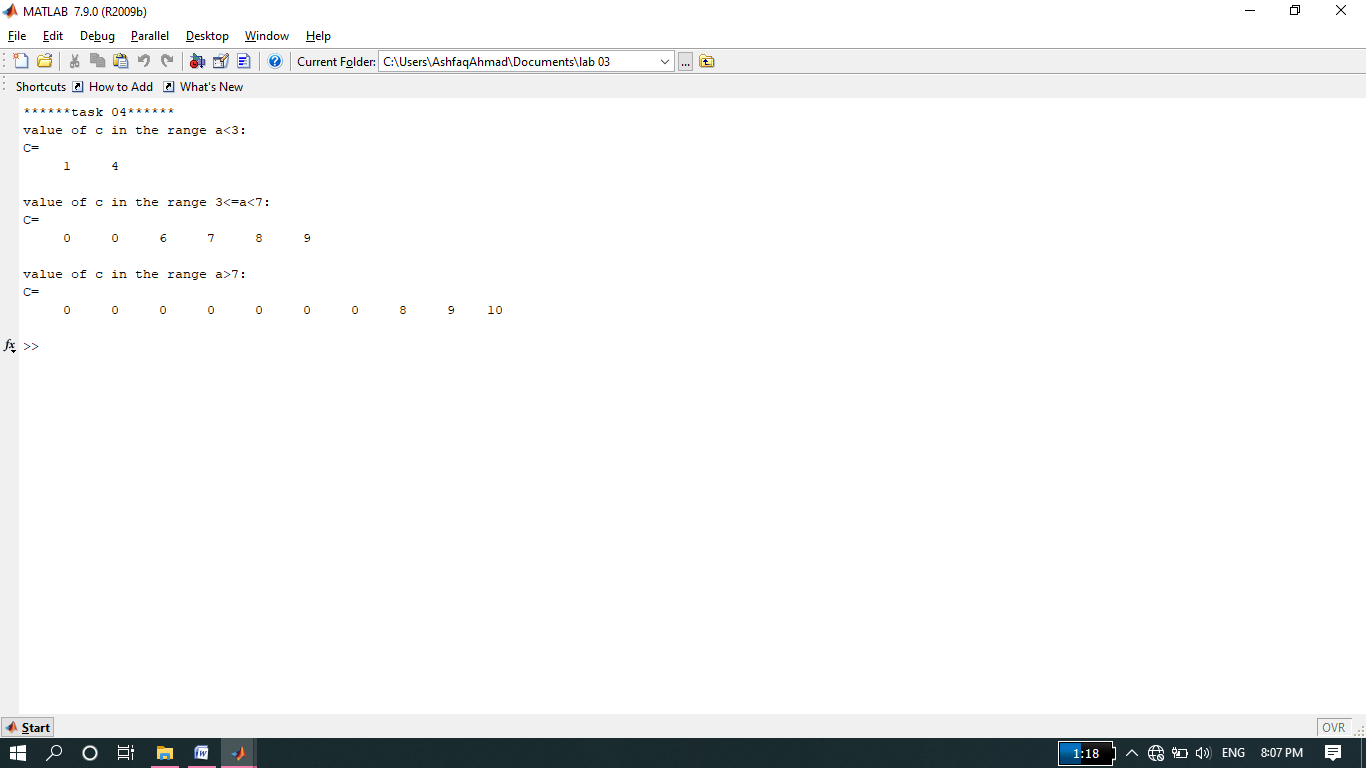
disp('value of c in the range a>7: ');

disp('C= ');

disp(d);

**Screenshot of Output:**





**Note:**

* We can also coding task 4 in function format just like we coded task 3 in function format.
* For the values of a we used single for loop.
* As there are 3 expression and each expression has own range of a, for that we used three if else statements.

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

* Rewrite the following statements to use only one if statement.
  + if x < y
  + if z < 5
  + w = x\*y\*z
  + end
  + end

**Source code:**

function[w]=task05(x,y,z)

disp('\*\*\*\*\*task 05\*\*\*\*\*\*');

x=input('please enter the value of x: ');

y=input('please enter the value of y: ');

z=input('please enter the value of z: ');

if x<y && z<5;

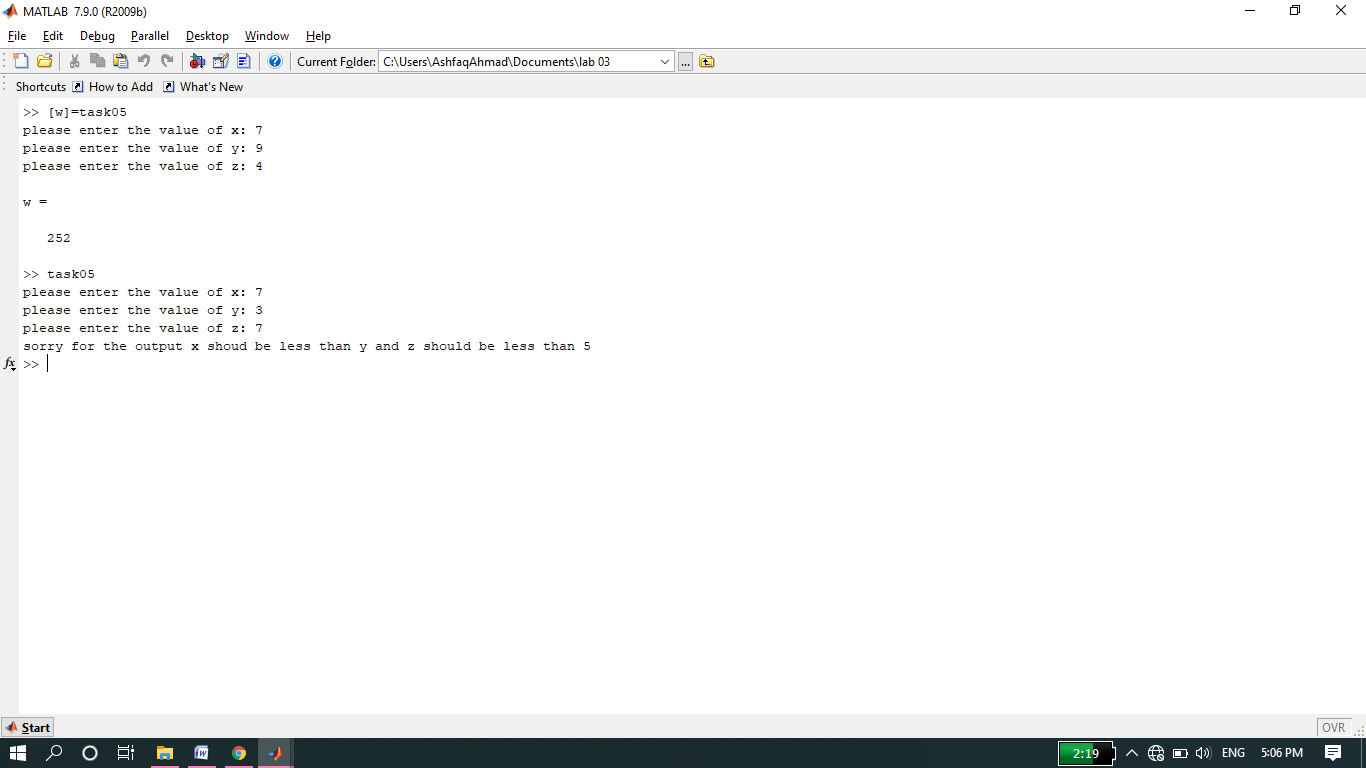
w=x\*y\*z;

else

disp('sorry for the output x should be less than y and z should be less than 5');

end

**ScreenShot of Output:**



**Note:**

* Here we used one if statement instead of two.

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

* Using for loop, generate the cube of the first ten integers.

**Source code:**

function cube

disp('\*\*\*\*\*\*\*\*task 06\*\*\*\*\*\*\*\*');

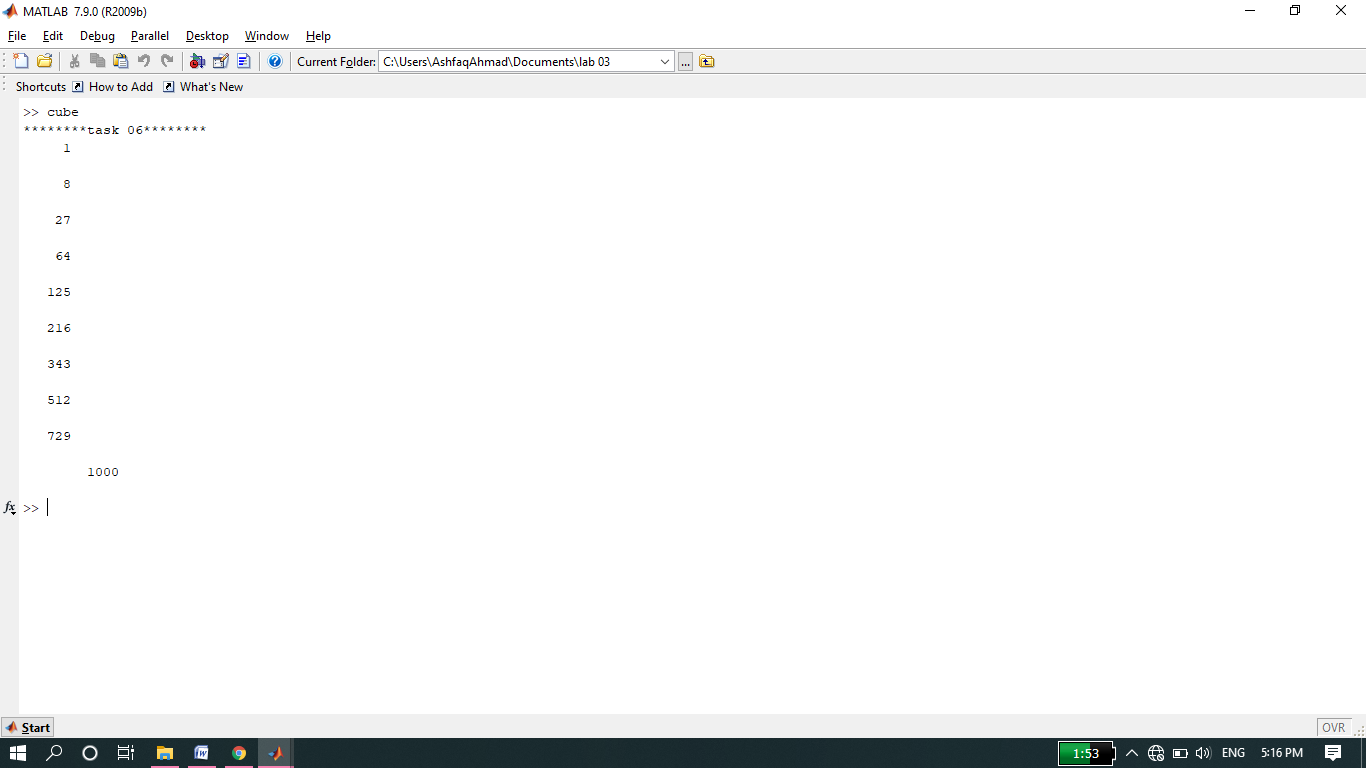
for i=1:1:10;

x=i.^3;

disp(x);

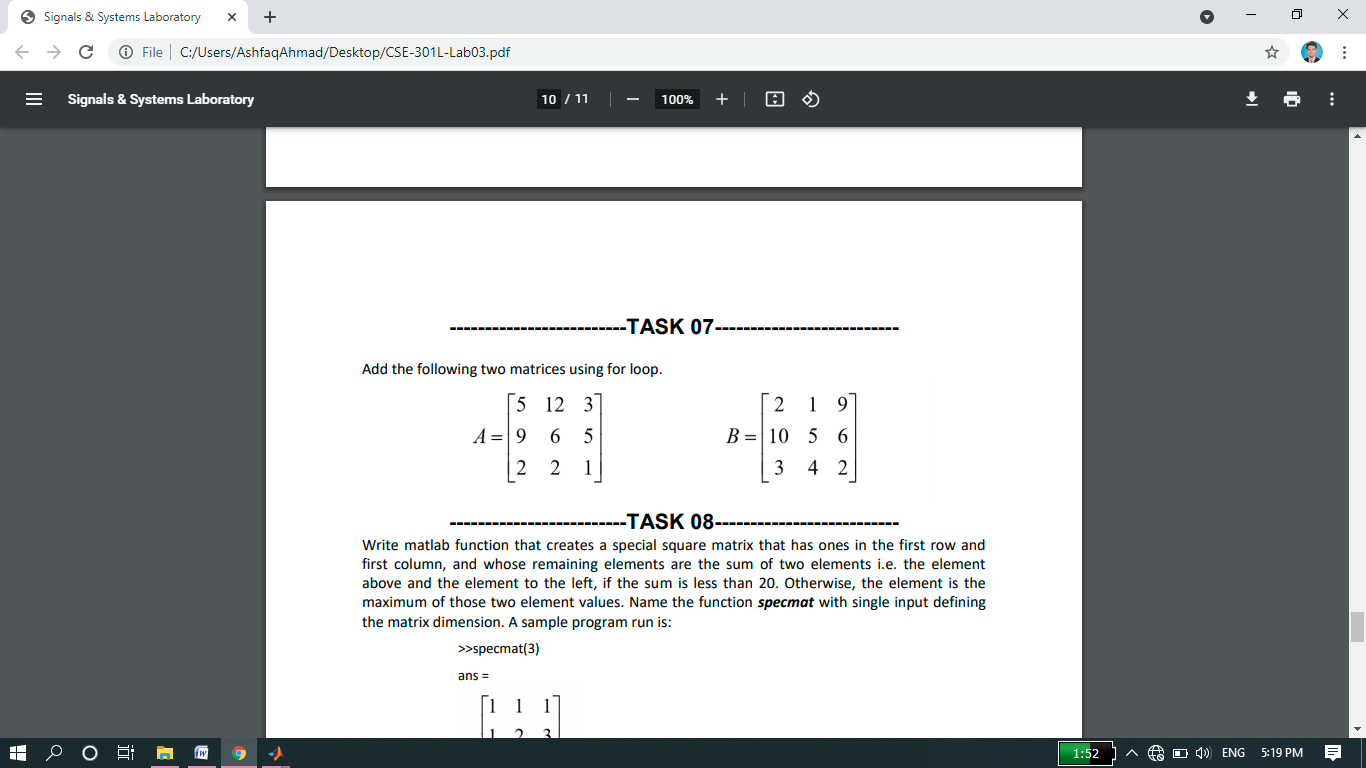
end

**Screenshot of Output:**



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

* Add the following two matrices using for loop.

****

**Source code:**

function addition

disp('\*\*\*\*\*task 07\*\*\*\*\*\*\*');

a=[5 12 3;9 6 5;2 2 1]

b=[2 1 9;10 5 6;3 4 2]

disp('sum of matrix a and b is :')

for i=1:1:3;

for j=1:1:3;

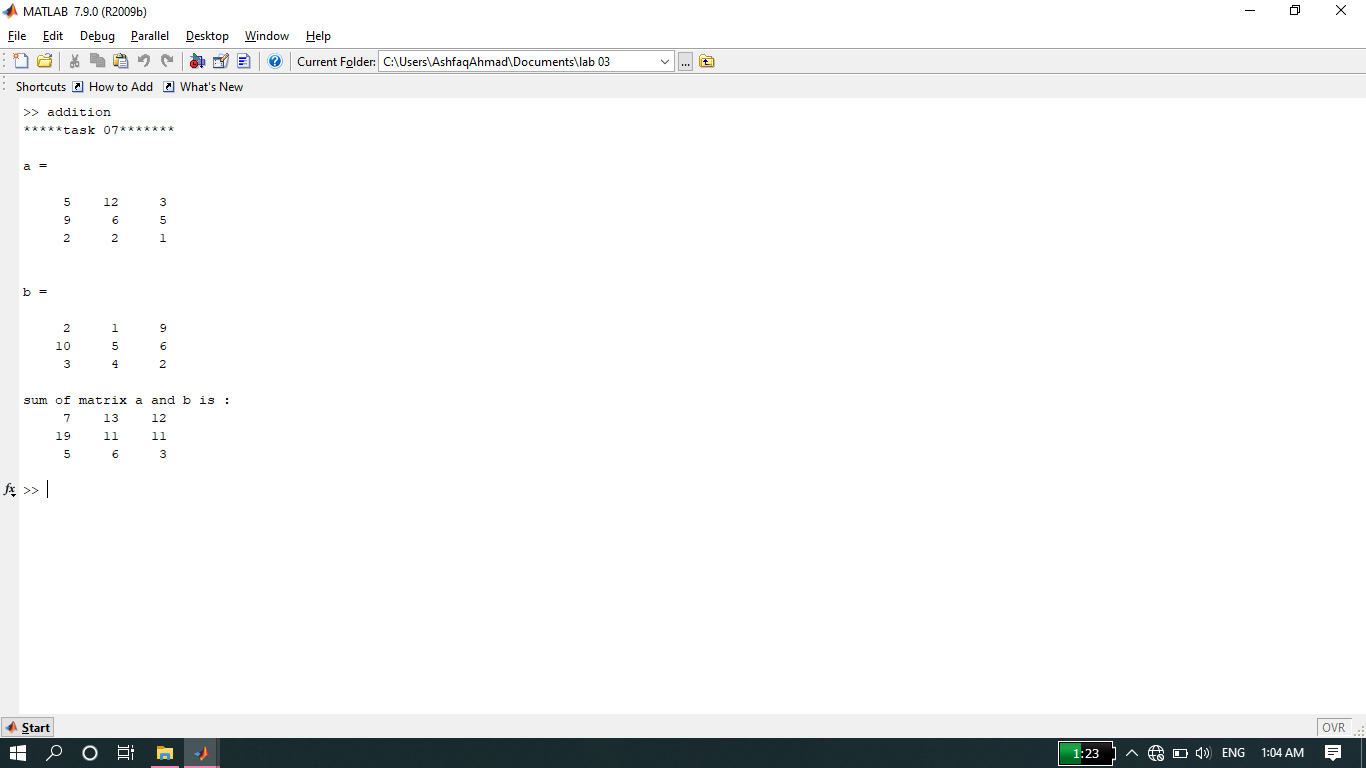
c(i,j)=a(i,j)+b(i,j);

end

end

disp(c);

**ScreenShot of Output:**

****

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

**Source code:**

function specmat

disp('\*\*\*\*task 08\*\*\*\*\*\*');

n=input('please enter the size of matrix: ');

for i=1:1:n;

for j=1:1:n;

if i==1 && j<=n;

m(i,j)=1;

else

if i<=n && j==1;

m(i,j)=1;

else

temp=m(i-1,j)+m(i,j-1);

if temp<20;

m(i,j)=temp;

else

if m(i-1,j)>m(i,j-1);

m(i,j)=m(i-1,j);

else

m(i,j)=m(i,j-1);

end

end

end

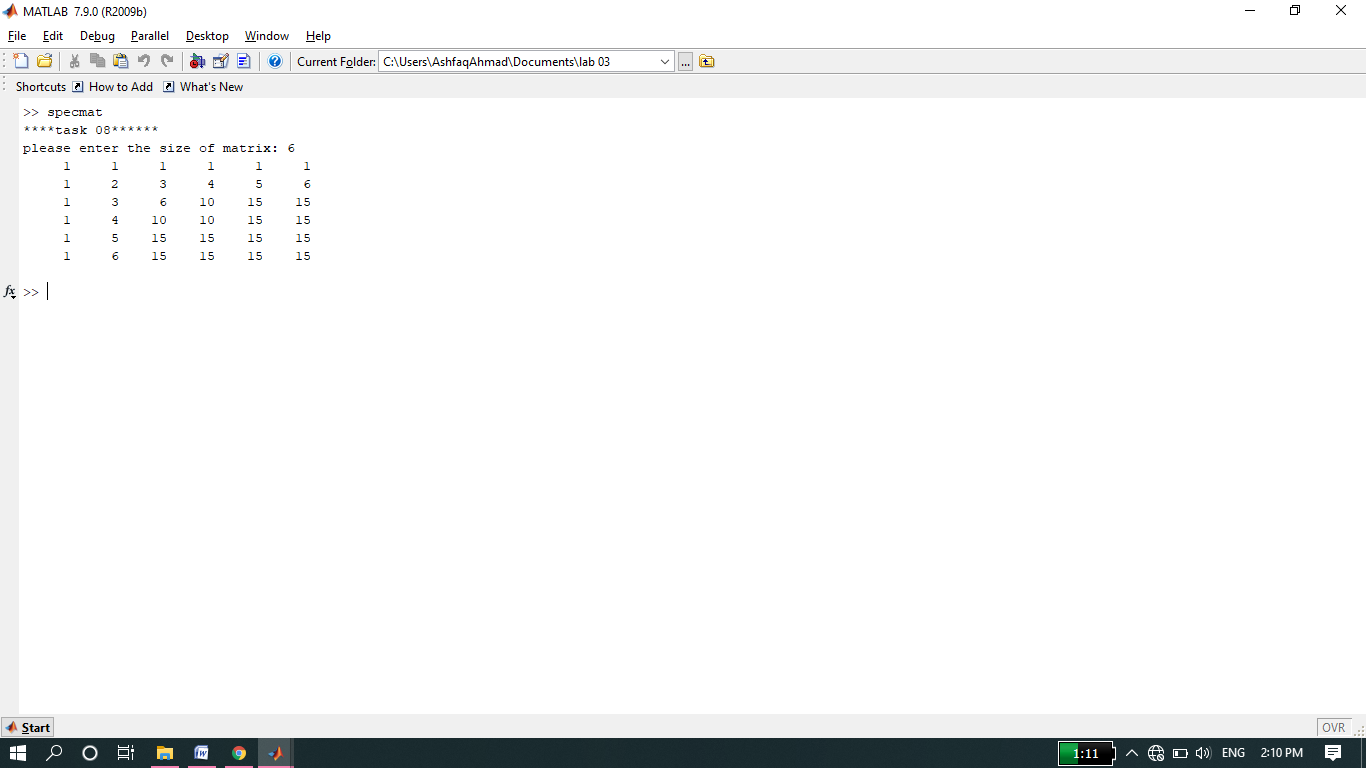
end

end

end

disp(m);

**Screenshot of Output:**



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

* Consider the following script file. Fill in the lines of the following table with the values that would be displayed immediately after the while statement if you ran the script file. Write in the values the variables have each time the while statement is executed. You might need more or fewer lines in the table. Then type in the file, and run it to check your answers.

**Source code:**

clc

clear all

close all

disp('\*\*\*\*\*\*\*task 09\*\*\*\*\*\*\*');

k = 1; b = -2; x = -1; y = -2;

while k <= 6;

disp('\*\*\*\*\*\*\*new iteration\*\*\*\*\*\*\*\*\*');

disp(k);

disp(b);

disp(x);

disp(y);

y = x^2 -3;

if y < b

b = y;

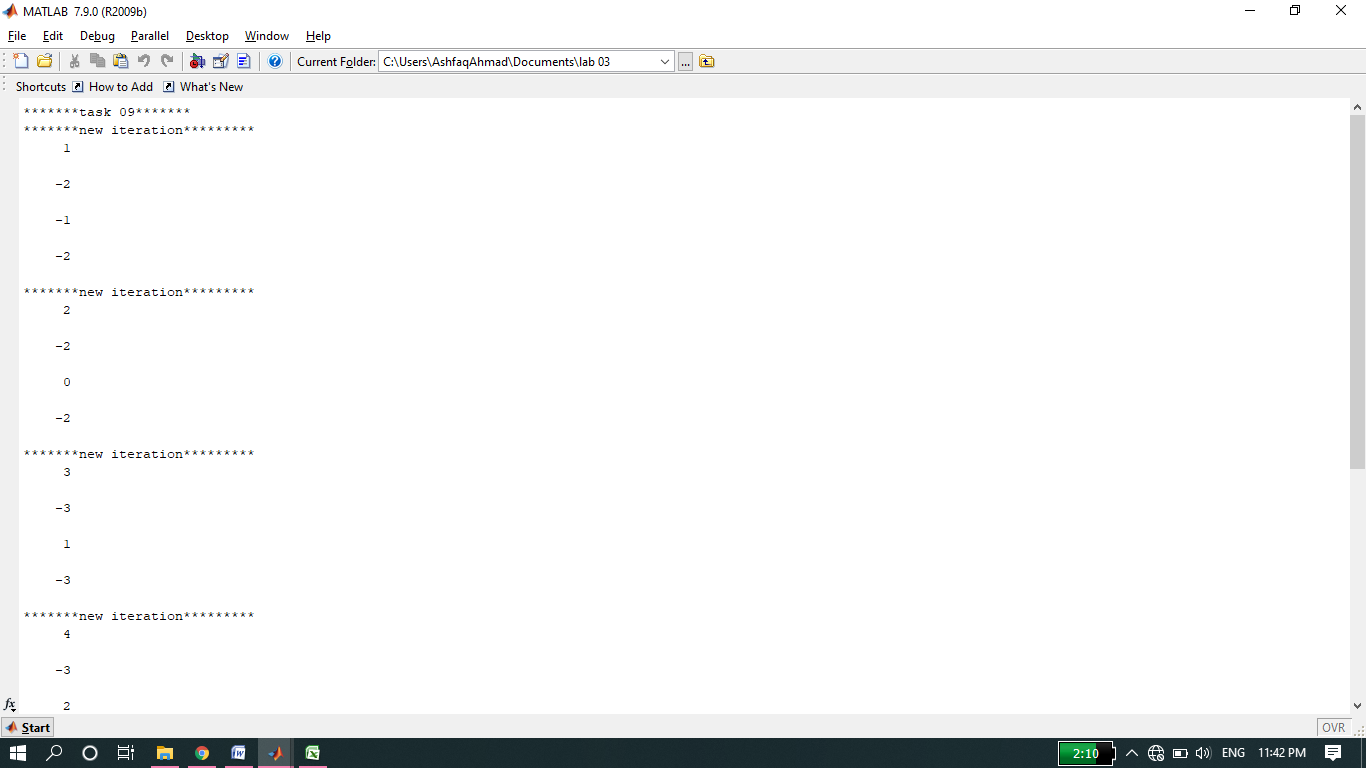
end

x = x + 1;

k = k + 1;

end

**Output:**



**Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pass | k | b | x | y |
| first | **1** | **-2** | **-1** | **-2** |
| second | **2** | **-2** | **0** | **-2** |
| third | **3** | **-3** | **1** | **-3** |
| fourth | **4** | **-3** | **2** | **-2** |
| fifth | **5** | **-3** | **3** | **1** |
| sixth | **6** | **-3** | **4** | **6** |

**-------------------------TASK 10--------------------------**

* Create an m‐file that inputs a number from user and then finds out the factorial of that number.

**Source code:**

function[f]= factorial(n)

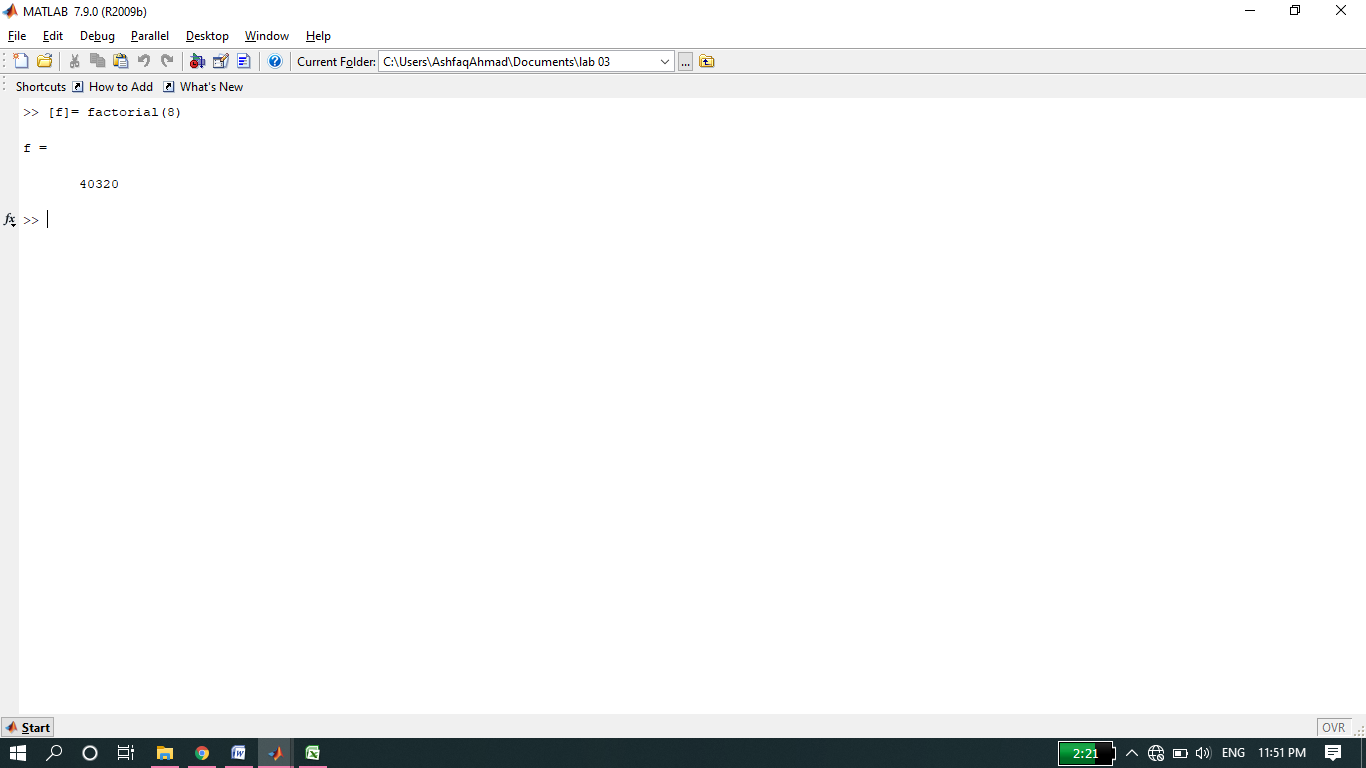
f=1;

for i=1:n;

f=f\*i;

end

**Screenshot of Output:**



**-------------------------TASK 11--------------------------**

* Create an m‐file that takes two vectors from user. Make sure that the second vector taken is of the same size as the first vector (Hint: use while loop). In a while loop, generate a third vector that contains the sum of the squares of corresponding entries of both the vectors.

**Source code:**

function[sum]=task11

n=input('please enter the size of vector: ');

i=1;

while(i<=n)

A(i)=input('please enter the vector A element: ');

B(i)=input('please enter the vector B element: ');

sum(i)=A(i)+B(i);

i=i+1;

end

disp('A= ');

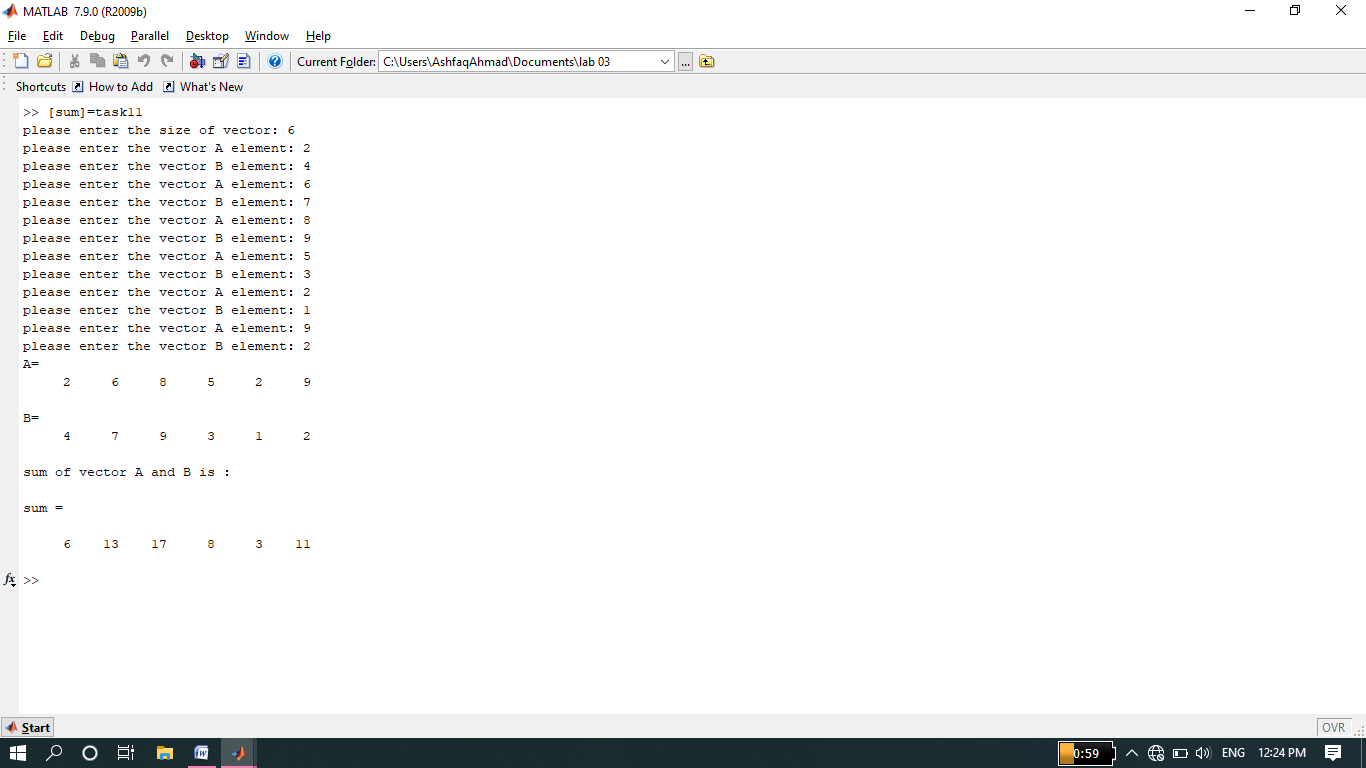
disp(A);

disp('B= ');

disp(B);

disp('sum of vector A and B is : ');

**Output:**



**-------------------------TASK 12--------------------------**

* **Perform given commands on various matrices.**

**>> ~A**

**>>A&B**

**>>A & ~B**

**>>A|B**

**>>A|~B**

**Source code:**

clc

clear all

close all

disp('\*\*\*\*\*\*Task no 12\*\*\*\*\*');

A=[2 3 0;5 6 7;6 0 9];

B=[3 0 0;8 6 3;7 6 0];

disp('A=');

disp(A);

disp('B=');

disp(B);

disp('\*\*\*\*\*result of (~A) command\*\*\*\*\*\*');

disp(~A);

disp('\*\*\*\*\*result of (A&B) command\*\*\*\*\*\*');

disp(A&B);

disp('\*\*\*\*\*result of (A&~B) command\*\*\*\*\*\*');

disp(A&~B);

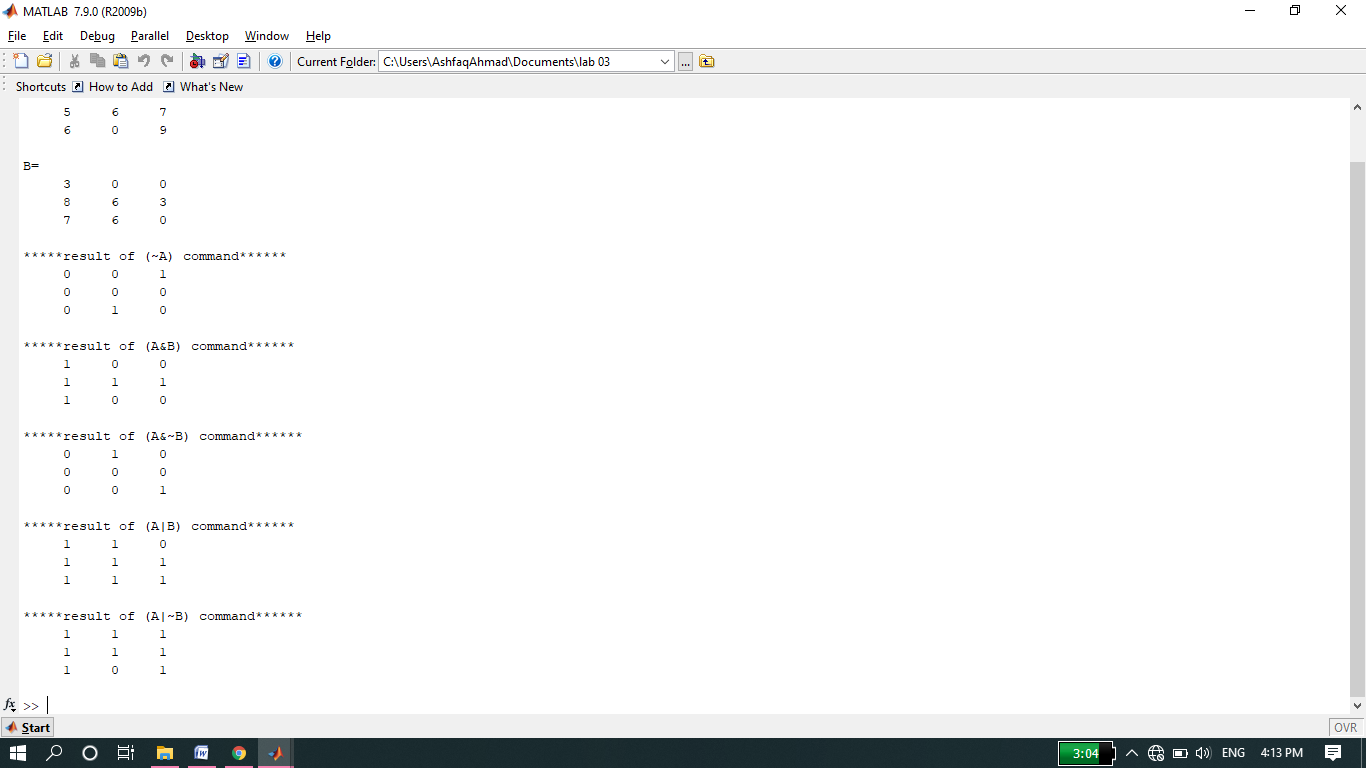
disp('\*\*\*\*\*result of (A|B) command\*\*\*\*\*\*');

disp(A|B);

disp('\*\*\*\*\*result of (A|~B) command\*\*\*\*\*\*');

disp(A|~B);

**Screenshot of Output:**

****

**Comments:**

* **>> ~A:** It convert all non-zero elements to zero and all zeros to non-zero elements in a vector.
* **A&B:** It act like AND operation I, e if any element in AND operation is zero output will also be zero. If all elements are non-zeroes in AND operation output will be 1.
* **A & ~B: ~**B first convert all non-zero elements to zero and all zeros to non-zero elementsin matrix B then it will be AND with matrix A I,e 0&nonzero=0 and nonzero&nonzero=1.
* **A|B :** it act like OR operation I,e **zero nonzero=1, nonzero+nonzero=1, zero+zero=0**
* **A|~B: ~**B first convert all non-zero elements to zero and all zeros to non-zero elementsin matrix B then it will be OR with matrix A.

**-------------------------TASK 13--------------------------**

* **Design a function fib(n) where n is input and generate Fibonacci series for n.**

**Source code:**

function fib(n)

disp('\*\*\*\*\*\*\*task no 13\*\*\*\*\*\*\*');

n=input('please enter the range of Fibonacci sequence: ');

first=0;

second=1;

i=1;

%we can also use for loop........

while(i<=n)

disp(first);

next=first+second;

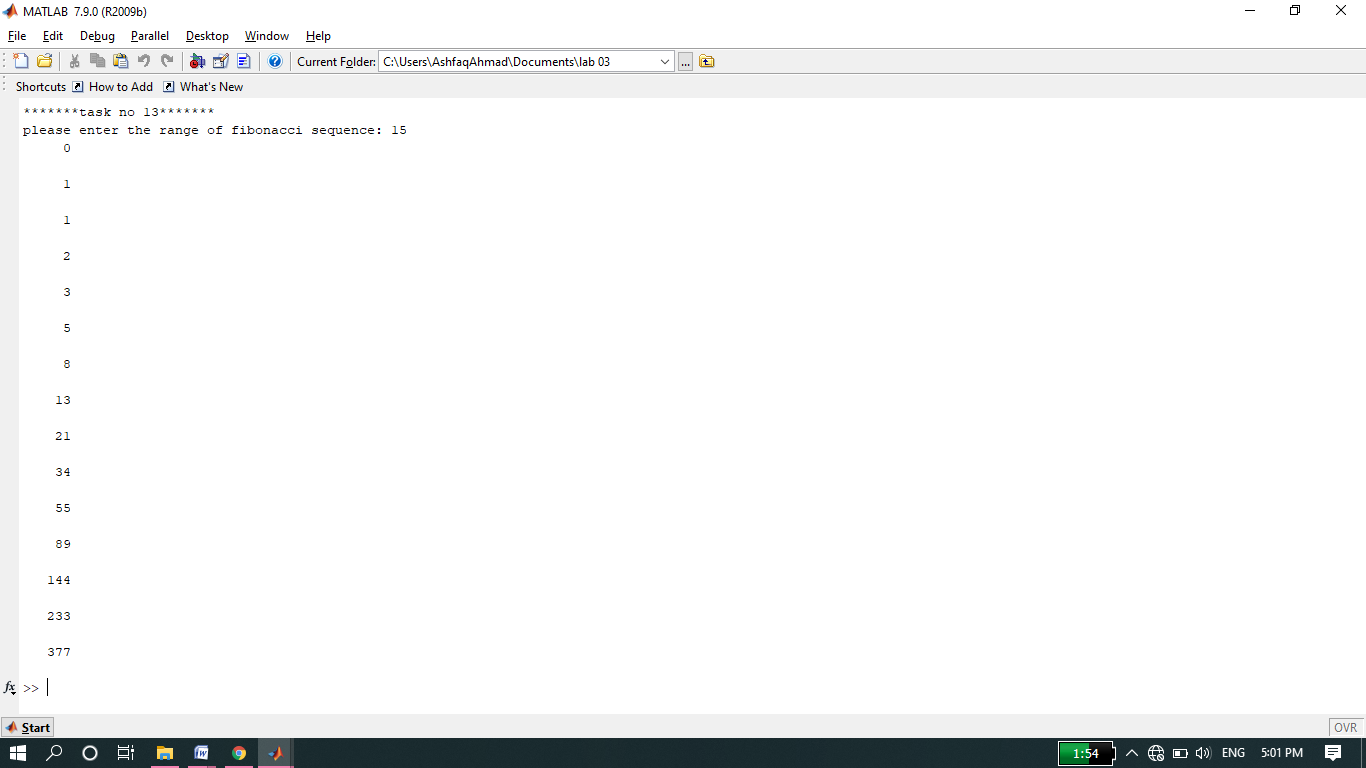
first=second;

second=next;

i=i+1;

end

**Screenshot of Output:**

****

**-------------------------TASK 14--------------------------**

* **Design function of two inputs and two outputs to determine height in cm and mass in kg from height in inch and weight in pound(lb.).**

**Source code:**

function[H,M]=conversion(h,m)

disp('\*\*\*\*\*\*\*Task no 14\*\*\*\*\*\*\*');

disp('\*\*\*\* part (A)\*\*\*\*\*\*\*');

h=input('please enter height in inches: ');

m=input('please enter mass in pounds: ');

disp('Now height in centimeter and mass in kilogram are: ');

H=h\*2.54

M=m\*0.453

disp('\*\*\*\* part(B)\*\*\*\*\*\*\*');

h=input('Now my height in inches: ');

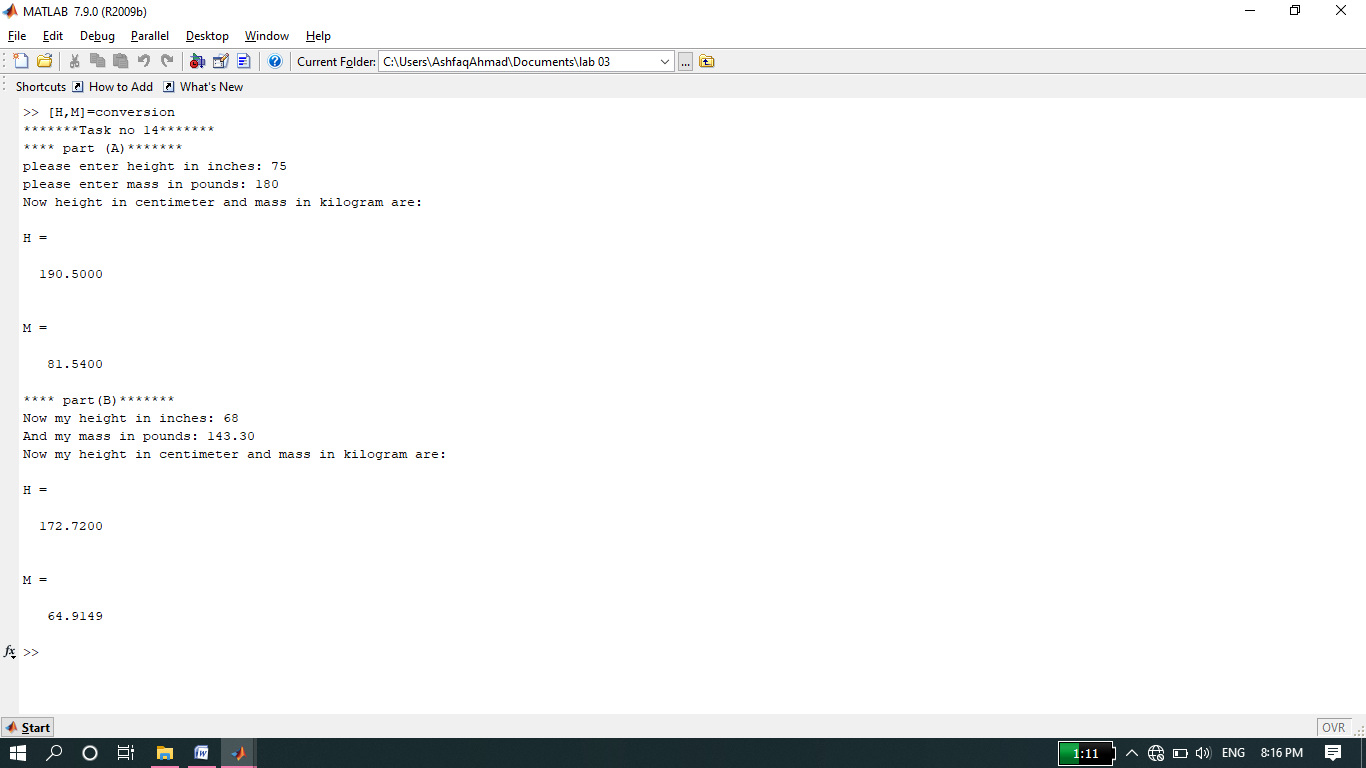
m=input('And my mass in pounds: ');

disp('Now my height in centimeter and mass in kilogram are: ');

H=h\*2.54;

M=m\*0.453;

**Screenshot of Output:**

****

**NOTE:**

* **1 inch=2.54 cm and 1 pound=0.453 kg**

**-------------------------TASK 15--------------------------**

* **File handling with Matlab. Create a file of different format in matlab.**

**Source code:**

clc

clear all

close all

disp('\*\*\*\*\*\*\*task no 15\*\*\*\*\*\*\*\*');

op=fopen('weekdays.txt','wt');

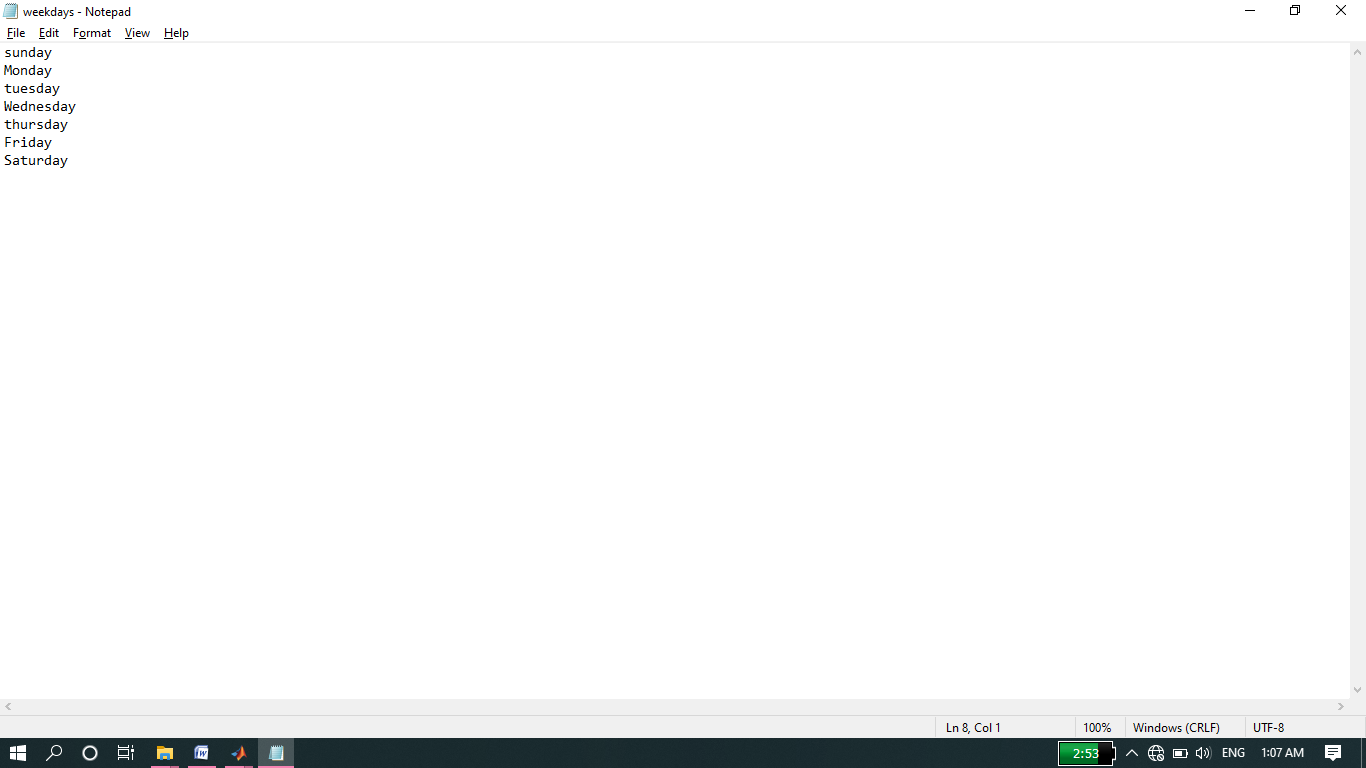
fprintf(op,'sunday\nMonday\ntuesday\nWednesday\n')

fprintf(op,'thursday\nFriday\nSaturday\n');

fclose(op);

clc

**Screenshot of Output:**

****

**-------------------------TASK 16--------------------------**

* **Selection sorting algorithm.**

**Source code:**

function[ret]= sort

for i=1:1:8;

ret(i)=input('please enter array elements: ');

end

disp('ret= ');

disp(ret);

disp('Now array in sorted form: ');

for i=1:1:7;

for j=i+1:8

if ret(j)<ret(i)

temp=ret(j);

ret(j)=ret(i);

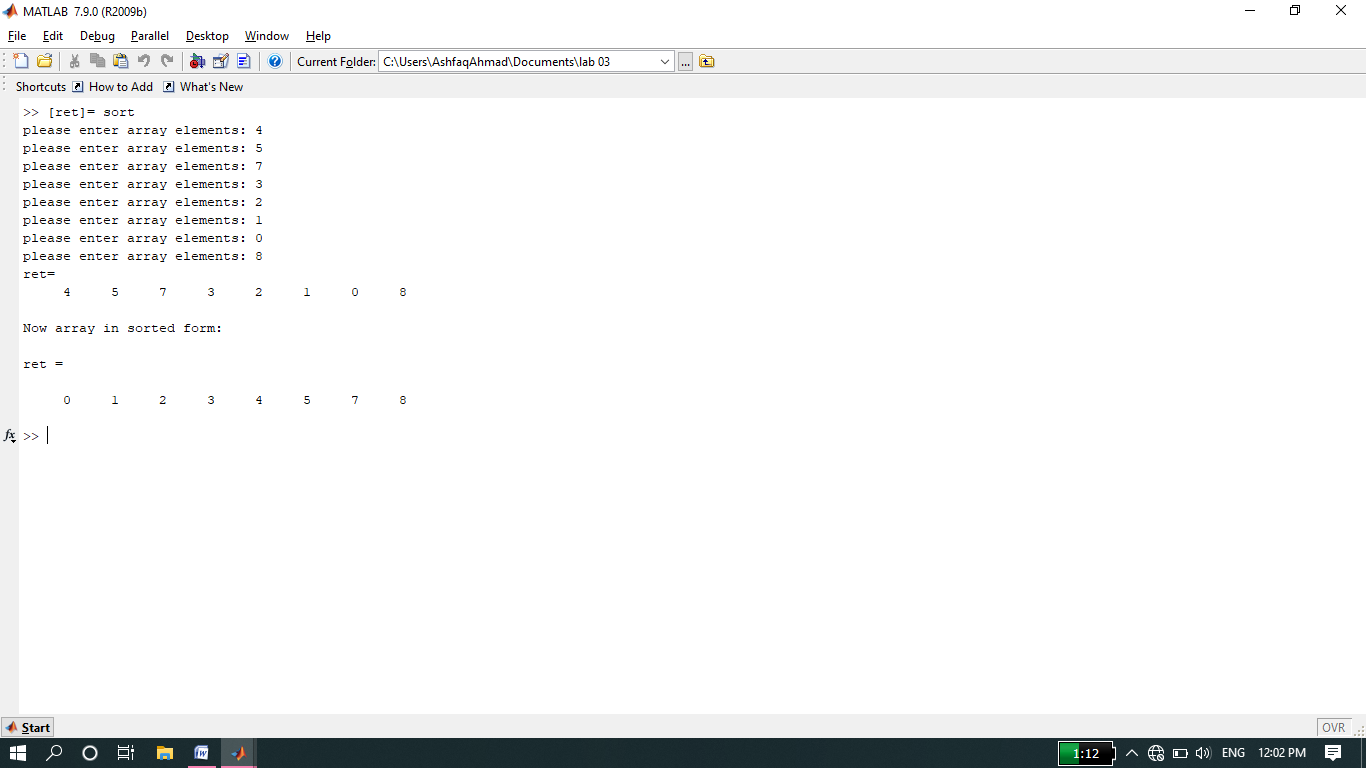
ret(i)=temp;

end

end

end

**Screenshot of Output:**

****

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