Lab Report # 03

CSE301 - L Signals & Systems Lab

Submitted by: AWAIS SADDIQUI

Registration No: 21PWCSE1993

Class Section: “A”

Submitted to:

Dr. Durr-e-Nayab

Department of Computer System Engineering

UET Peshawar

LAB ASSESSMENT RUBRICS

Marking Criteria	Exceeds expectation (5-4)	Meets expectation (3-2)	Does not meet expectation (1)	Score
1. Realization of Experiment	Program compiles (noerrors and no warnings). Program always works correctly and meets the specification(s). Completed between 71-100% of the requirements.	Program compiles (no errors and some warnings). Some details of the program specification are violated, program functions incorrectly for some inputs. Completed between 41-70% of the requirements.	Program fails to or compile with lots of warnings. Program only functions correctly in very limited cases or not at all. Completed less than 40% of the requirements.	30%
2. Ability to apply required code utility or data structure	Able to apply required data type or data structure and produce correct results. Familiarize and selects proper functions for simulation of given problem using software tools like MATLAB.	Able to apply required data type or data structure but does not produce correct results. Need guidance to select proper functions for simulation of given problem using software tools like MATLAB.	Unable to identify required data type or data structure. Incapable of selecting proper functions for simulation of given problem using software tools like MATLAB.	20%
3. Documentation	Clearly and effectively documented including descriptions of all variables/functions. Specific purpose is noted for each function, control structure, input requirements and output results.	Basic documentation including descriptions of all variables/functions. Specific purpose is noted for each function and control structure.	No documentation included.	10%

4. Ability to run/debug	Executes Matlab codes without errors, excellent user prompts, good use of symbols, spacing in output. Thorough and organized testing has been completed and output from test cases is included.	Executes Matlab codes without errors. User prompts are understandable, minimum use of symbols or spacing in output. Some testing has been completed.	Does not execute Matlab codes due to errors. User prompts are misleading or nonexistent. No testing has been completed.	20%
5. Results compilation	Show processed results effectively by conducting simple computations and plotting using collected data	Show processed results effectively by conducting simple computations and plotting using collected data with minor error	Unable to show processed results effectively by conducting simple computations and plotting using collected data with minor error	10%
6. Efficiency	Excellent use of CPU and Memory.	Good but not smart use of CPU and Memory.	Inefficient use of CPU and Memory.	10%
7. Lab Performance (Team work and Lab etiquettes)	Actively engages and cooperates with other group members in an effective manner. Respectfully and carefully observes safety rules and procedures	Cooperates with other group members in a reasonable manner. Observes safety rules and procedures with minor deviation.	Distracts or discourages other group members from conducting the experiment. Disregards safety rules and procedures.	10%

Instructor:

Name: ___________________________

Signature: ________________________

MAKING FUNCTIONS:

A function is a reusable piece of code that can be called from program to accomplish some specified functionality. A function takes some input arguments and returns some output. To create a function that adds two numbers and stores the result in a third variable

Script Vs Function:

A script is simply a collection of Matlab commands in an m-file. Upon typing the name of the file (without the extension), those commands are executed as if they had been entered at the keyboard. Functions are used to create user-defined Matlab commands.

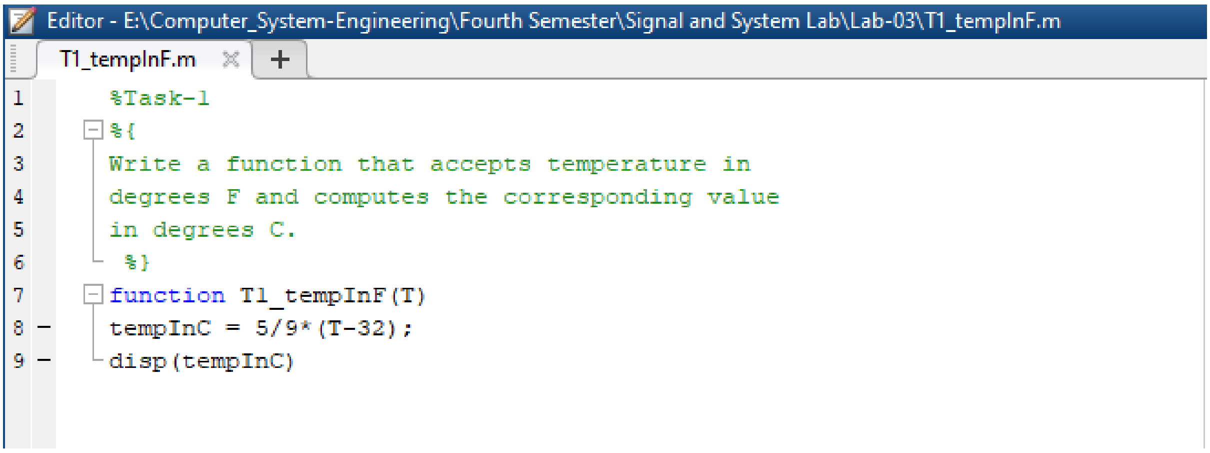
Objectives of the Lab:

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

Task 01:

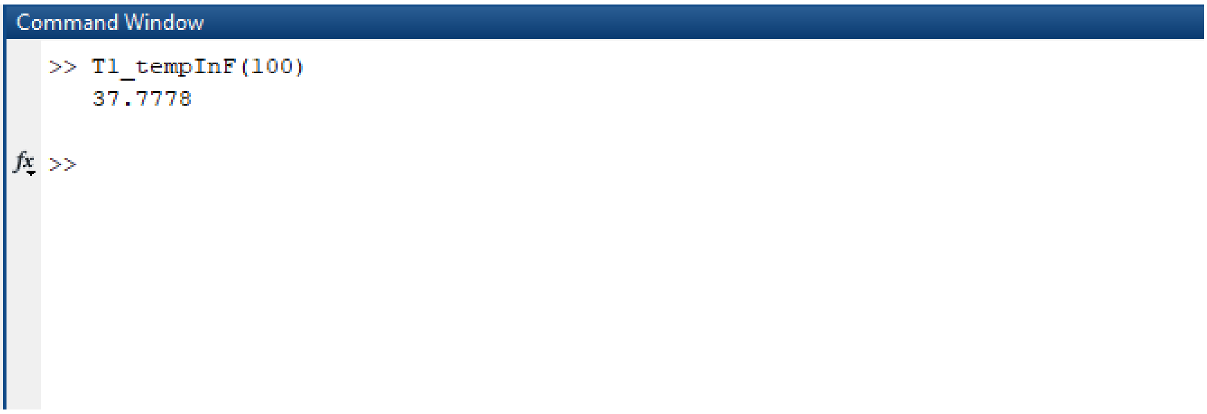
Write a function that accepts temperature in degrees F and computes the corresponding value in degrees C.

Code:


Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\T1_tempInF.m
T1_tempInF.m
+

```
%Task-1
%{
Write a function that accepts temperature in
degrees F and computes the corresponding value
in degrees C.
 %}
function T1_tempInF(T)
tempInC = 5/9*(T-32);
disp(tempInC)
```

Output:

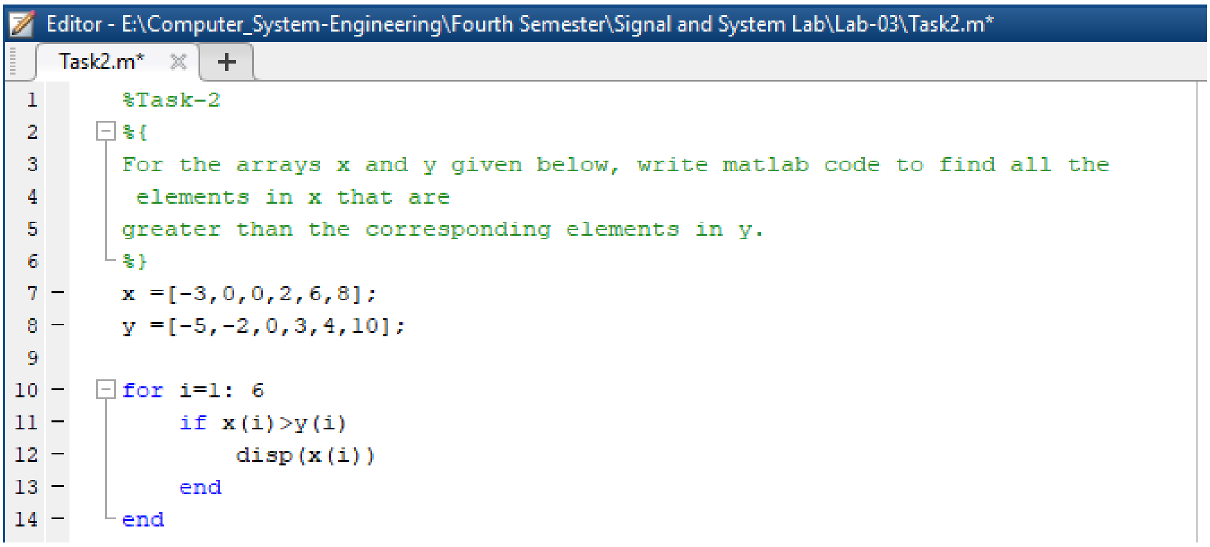


```
>> T1_tempInF(100)
   37.7778

>>
```

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.

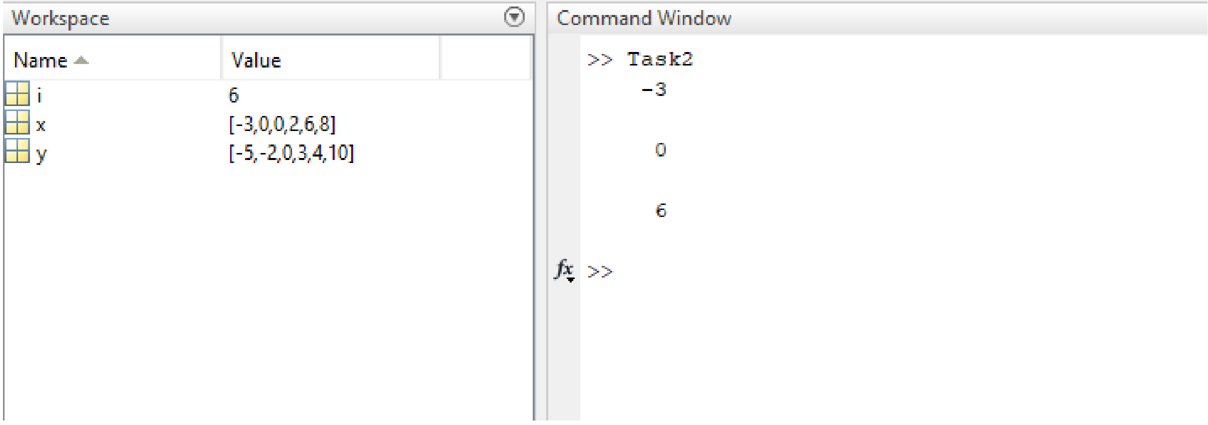
Code:



Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\Task2.m*
Task2.m*
+

```
%Task-2
%{
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];

for i=1: 6
    if x(i)>y(i)
        disp(x(i))
    end
end
```

Output:



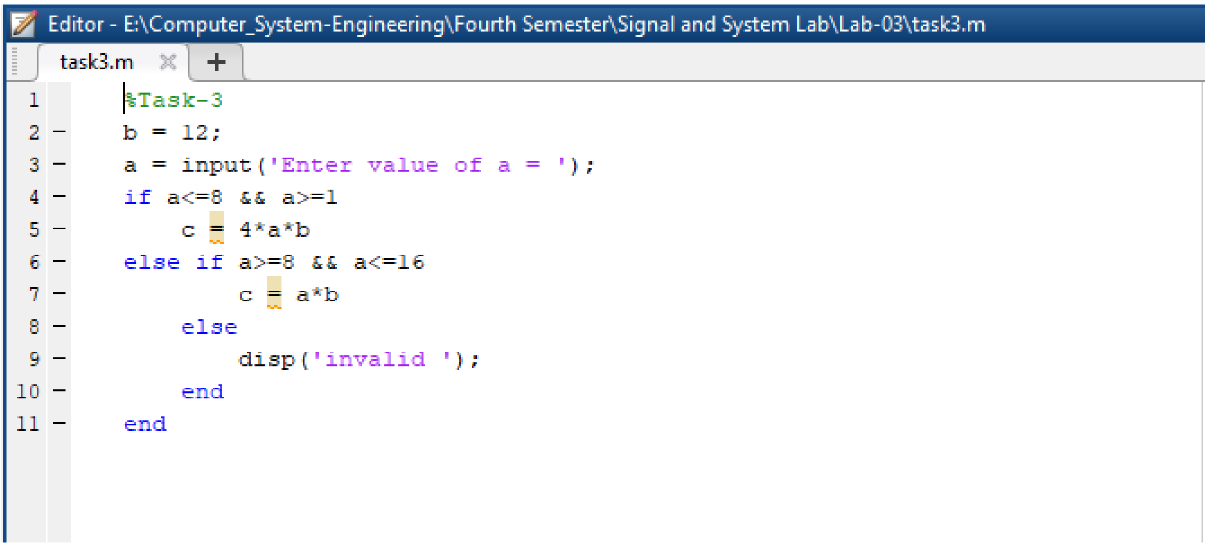
Name ▲	Value
i	6
x	[-3,0,0,2,6,8]
y	[-5,-2,0,3,4,10]

```
>> Task2
    -3

     0

     6

>>
```

Task 03:

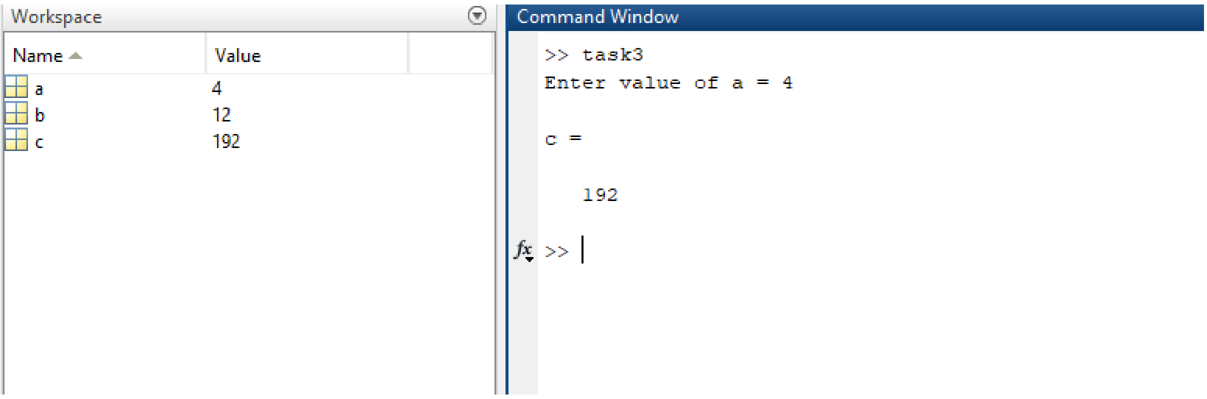
For 0 < a ≤ 16, find the values of C

Code:


Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\task3.m
task3.m
+

```
%Task-3
b = 12;
a = input('Enter value of a = ');
if a<=8 && a>=1
    c = 4*a*b
else if a>=8 && a<=16
        c = a*b
    else
        disp('invalid ');
    end
end
```

Output:



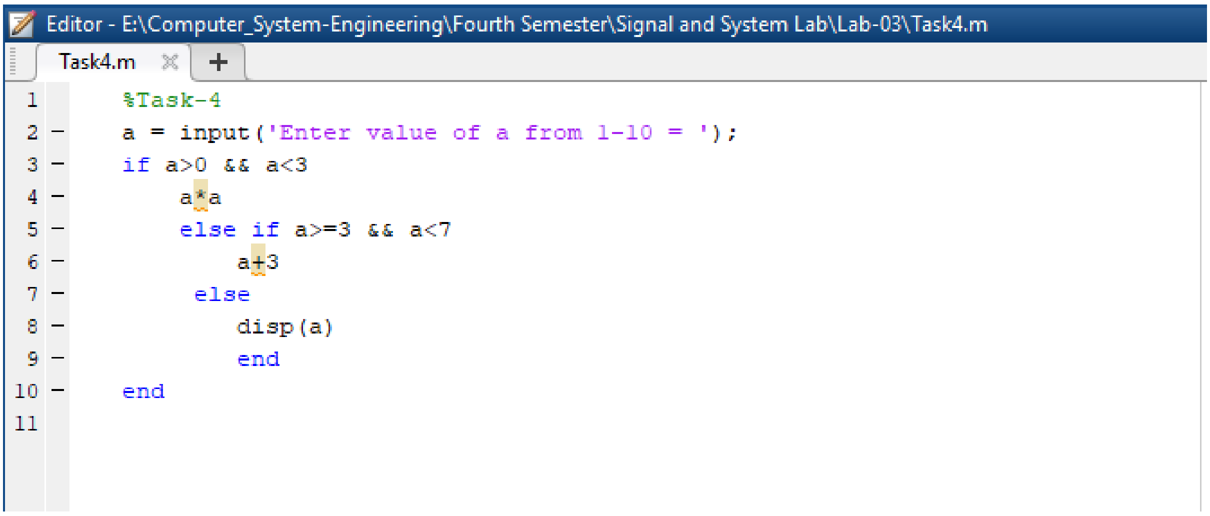
Name ▲	Value
a	4
b	12
c	192

```
>> task3
Enter value of a = 4

c =

   192

>> 
```

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:

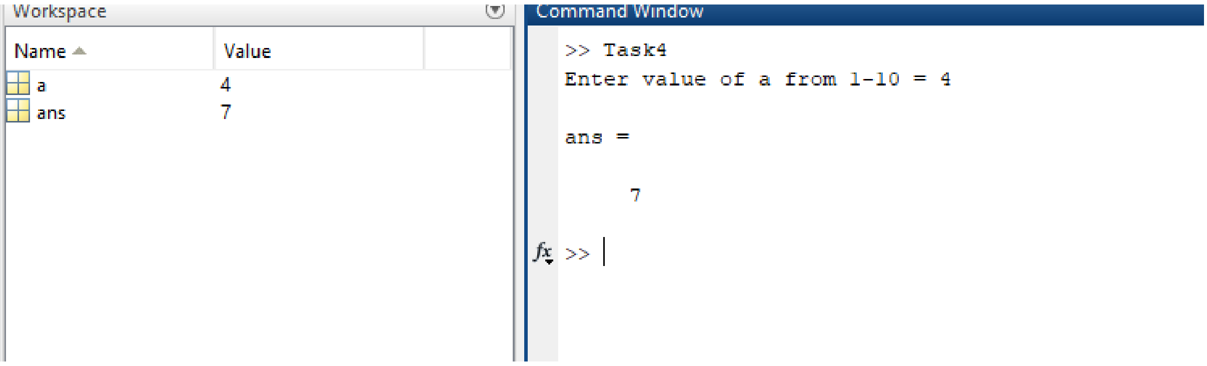
Code:



Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\Task4.m
Task4.m
+

```
%Task-4
a = input('Enter value of a from 1-10 = ');
if a>0 && a<3
    a*a
    else if a>=3 && a<7
        a+3
     else
        disp(a)
        end
end
```

Output:



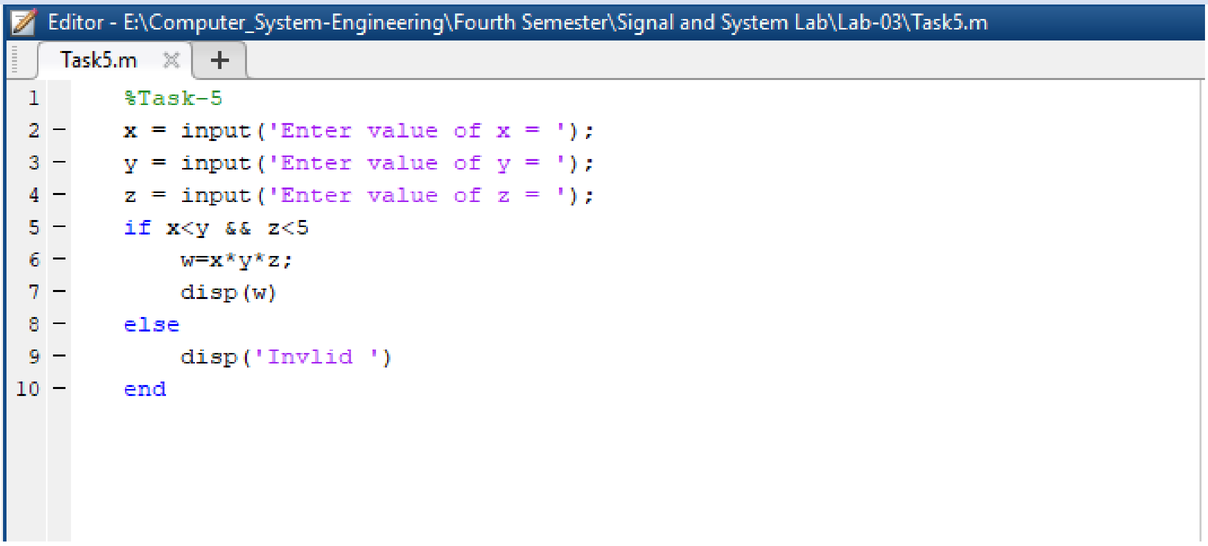
Name	Value
a	4
ans	7

```
>> Task4
Enter value of a from 1-10 = 4

ans =

     7

>> |
```

Task 05:

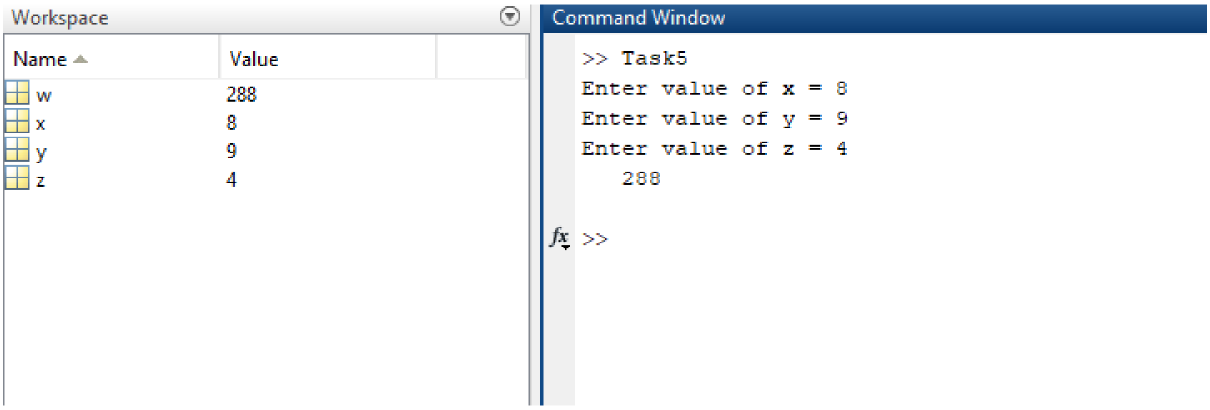
Rewrite the following statements to use only one if statement

Code:


Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\Task5.m
Task5.m
+

```
%Task-5
x = input('Enter value of x = ');
y = input('Enter value of y = ');
z = input('Enter value of z = ');
if x<y && z<5
    w=x*y*z;
    disp(w)
else
    disp('Invlid ')
end
```

Output:



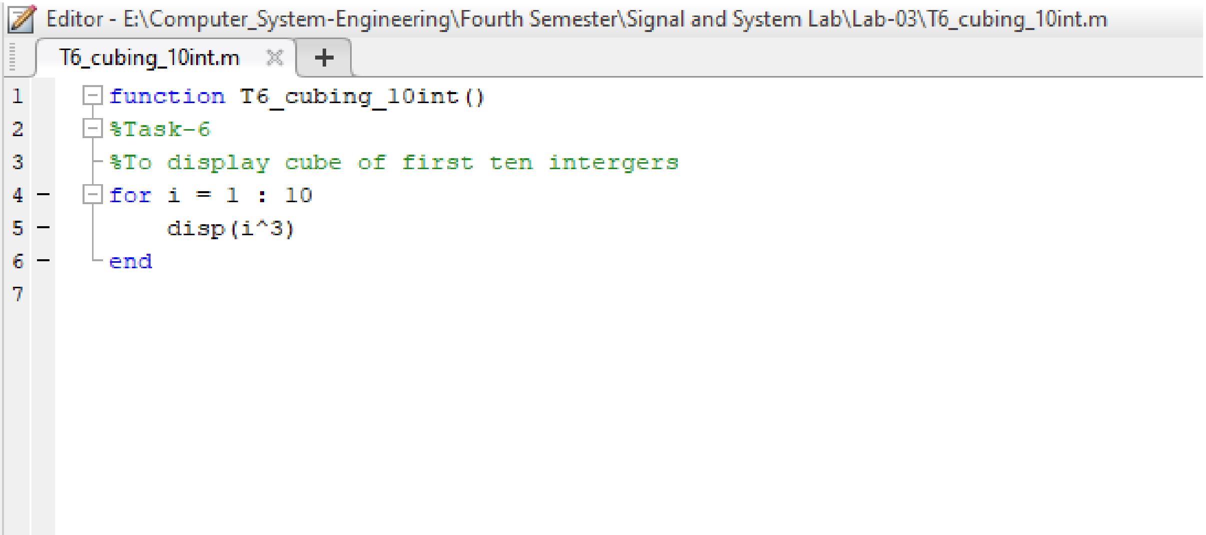
Name	Value
w	288
x	8
y	9
z	4

```
>> Task5
Enter value of x = 8
Enter value of y = 9
Enter value of z = 4
   288

>>
```

Task 06:

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

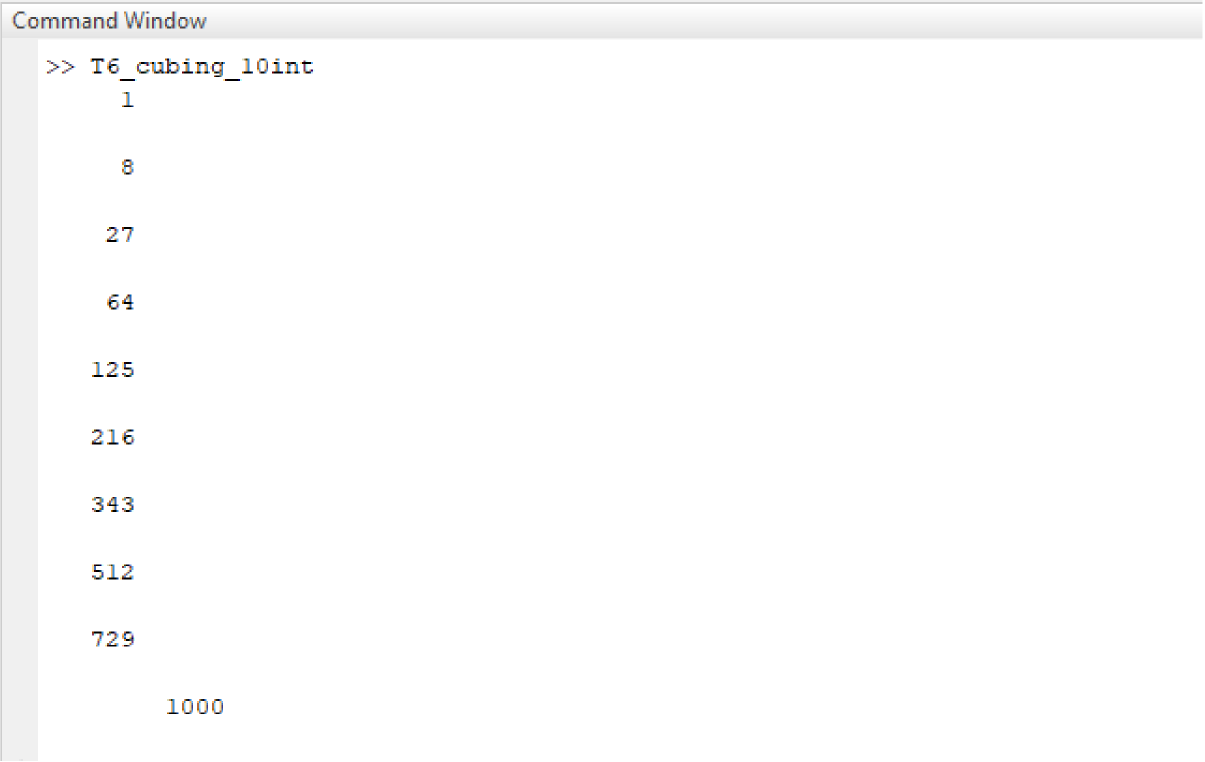
Code:



Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\T6_cubing_10int.m
T6_cubing_10int.m
+

```
function T6_cubing_10int()
%Task-6
%To display cube of first ten intergers
for i = 1 : 10
    disp(i^3)
end
```

Output:

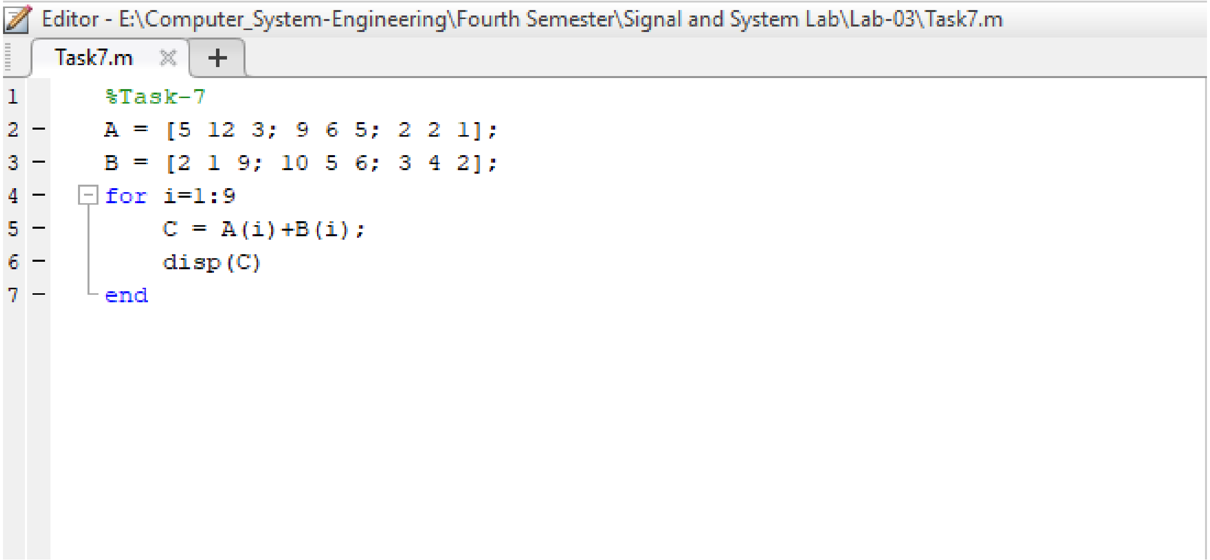


```
>> T6_cubing_10int
     1

     8

    27

    64

   125

   216

   343

   512

   729

        1000
```

Task 07:

Add the following two matrices using for loop.

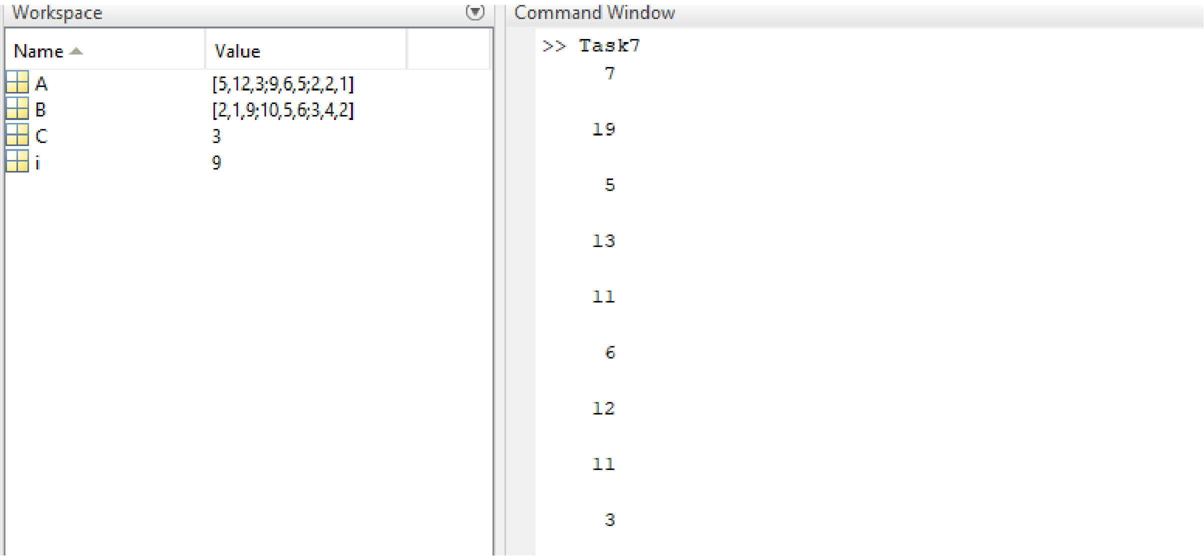
Code:



Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\Task7.m

```
%Task-7
A = [5 12 3; 9 6 5; 2 2 1];
B = [2 1 9; 10 5 6; 3 4 2];
for i=1:9
    C = A(i)+B(i);
    disp(C)
end
```

Output:



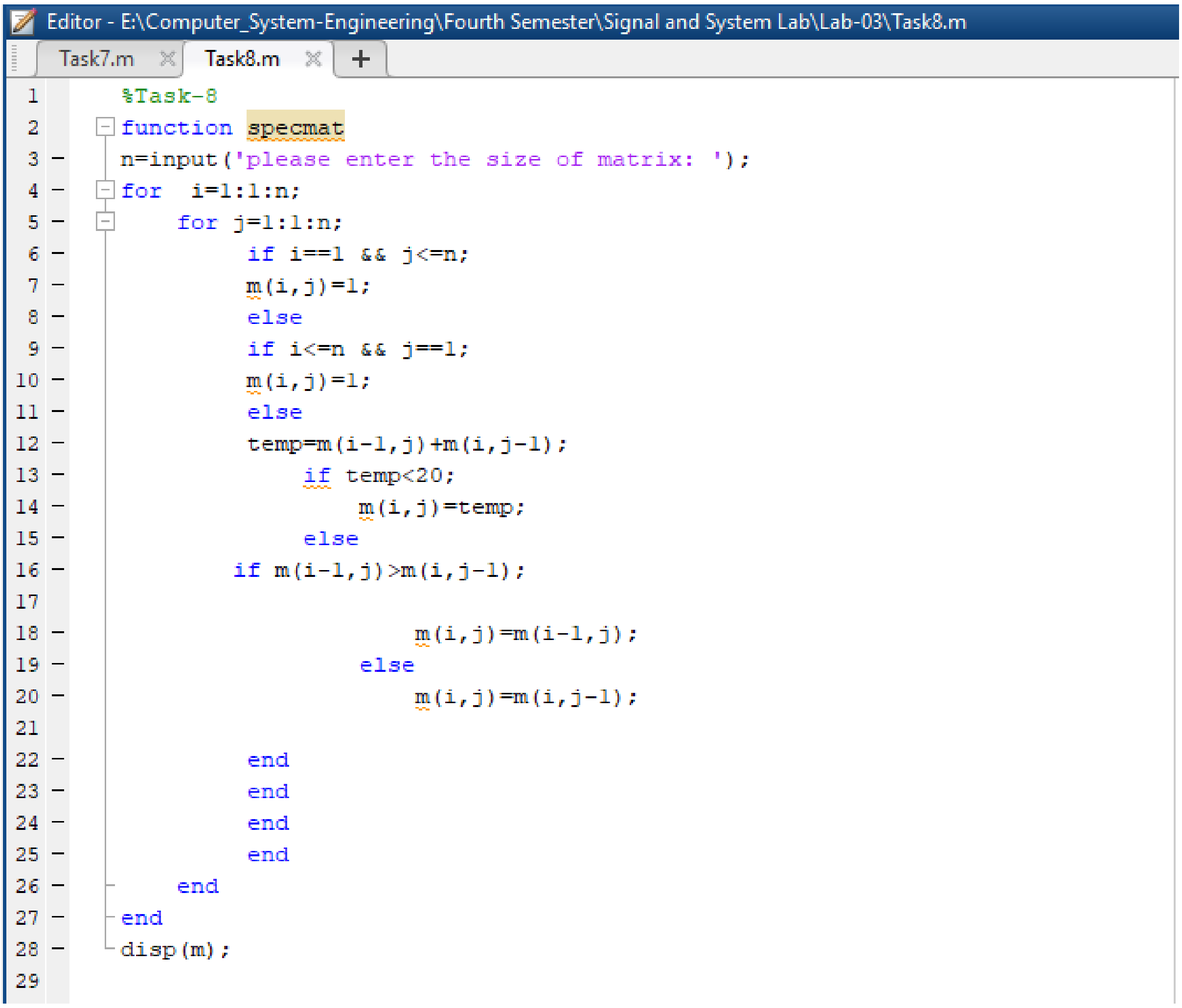
Name	Value
A	[5,12,3;9,6,5;2,2,1]
B	[2,1,9;10,5,6;3,4,2]
C	3
i	9

```
>> Task7
     7

    19

     5

    13

    11

     6

    12

    11

     3
```

Task 08:

Write MATLAB function that creates a special square matrix that has ones in the first row and first column, and whose remaining elements are the sum of two elements i.e. the element above and the element to the left, if the sum is less than 20. Otherwise, the element is the maximum of those two element values

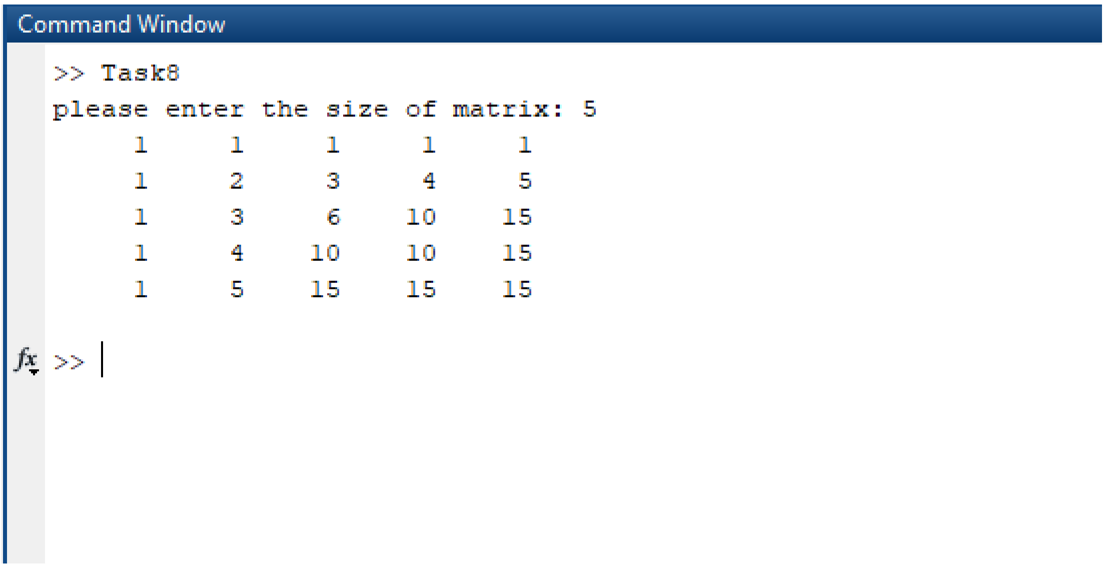
Code:



Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\Task8.m
Task7.m
Task8.m
+

```
%Task-8
function specmat
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);
```

Output:

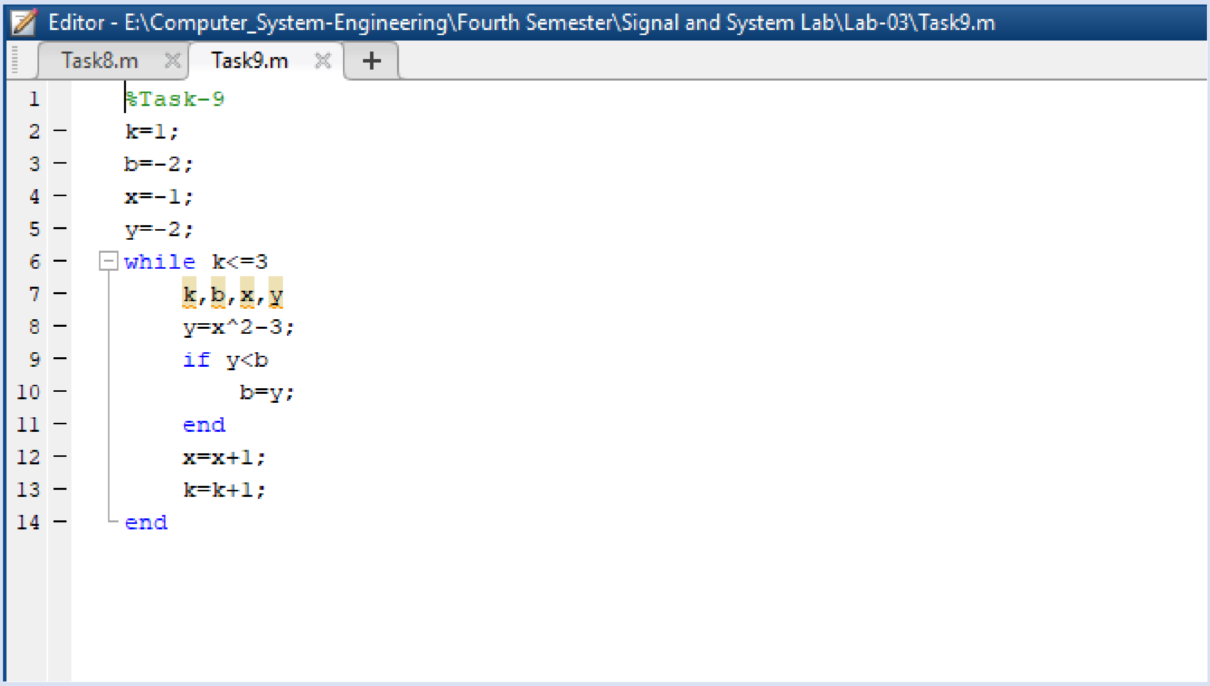


```
>> Task8
please enter the size of matrix: 5
      1      1      1      1      1
      1      2      3      4      5
      1      3      6     10     15
      1      4     10     10     15
      1      5     15     15     15

>> 
```

Task 09:

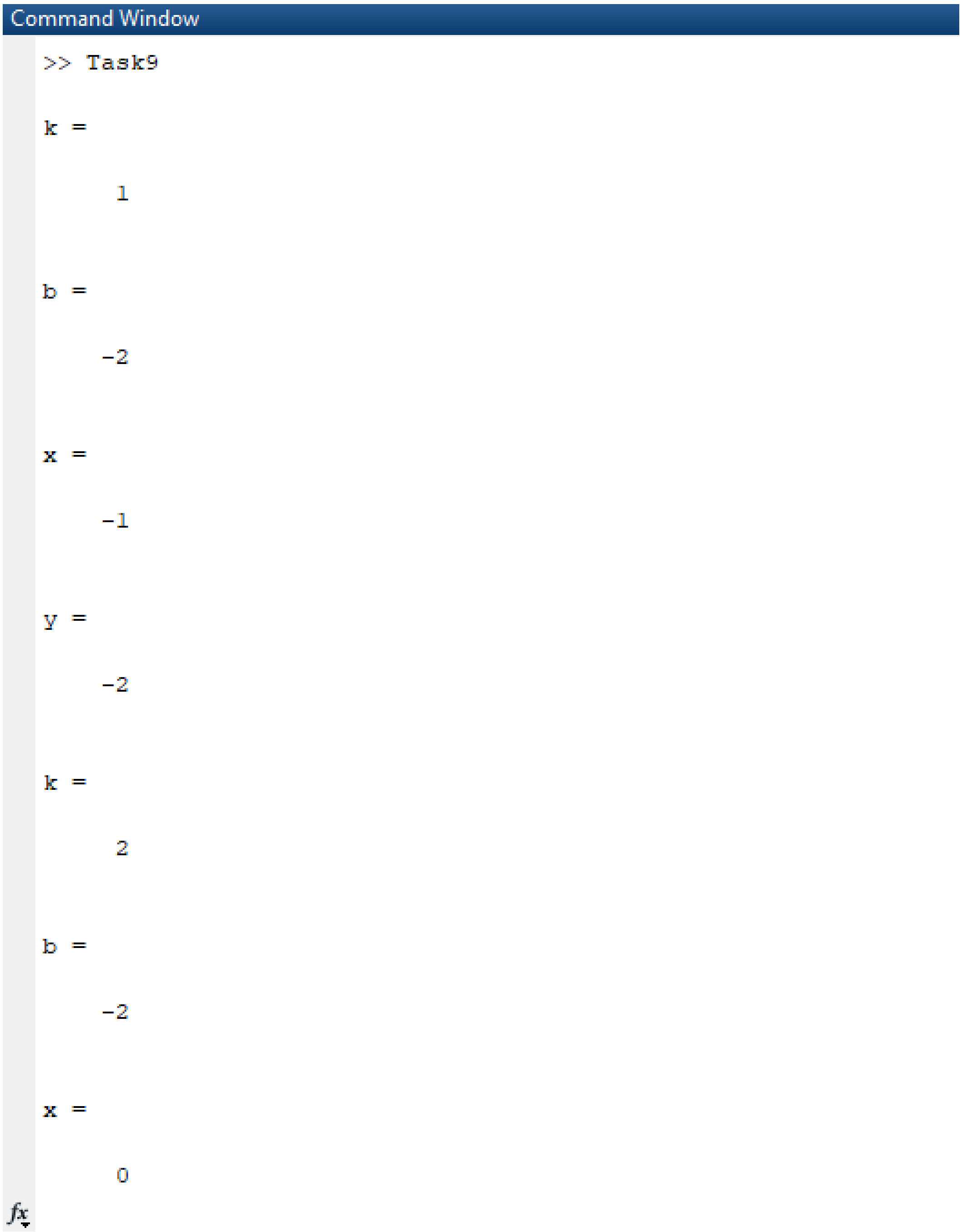
Consider the following script file. Fill in the lines of the following table with the values that would be displayed immediately after the while statement if you ran the script file. Write in the values the variables have each time the while statement is executed. You might need more or fewer lines in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

Code:


Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\Task9.m
Task8.m
Task9.m
+

```
%Task-9
k=1;
b=-2;
x=-1;
y=-2;
while k<=3
    k,b,x,y
    y=x^2-3;
    if y<b
        b=y;
    end
    x=x+1;
    k=k+1;
end
```

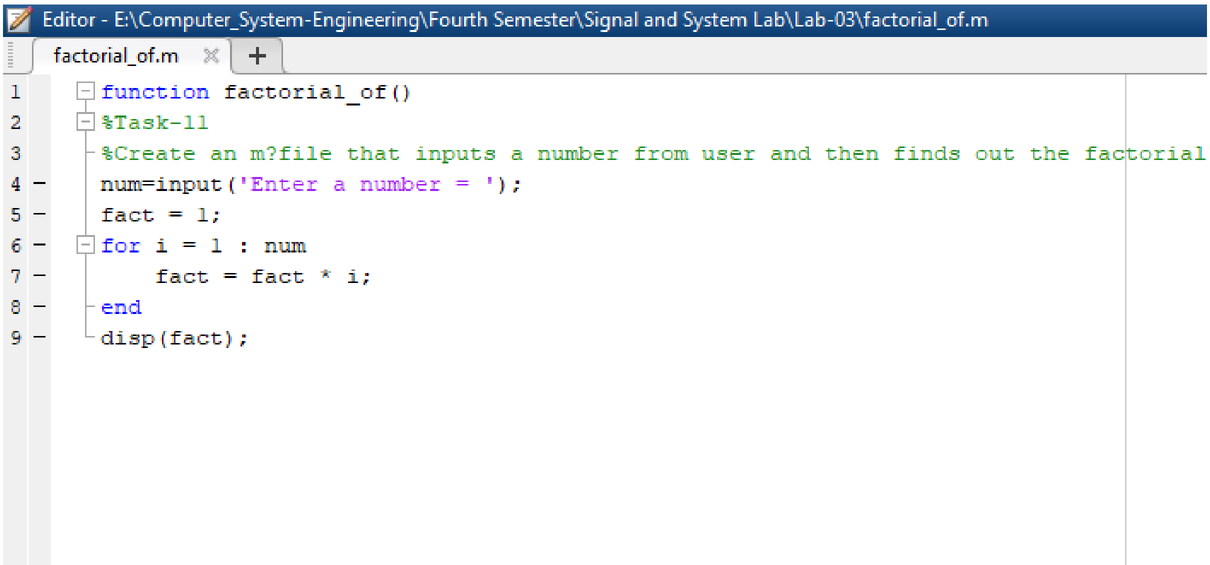
Output:



```
>> Task9

k =

     1


b =

    -2


x =

    -1


y =

    -2


k =

     2


b =

    -2


x =

     0
```

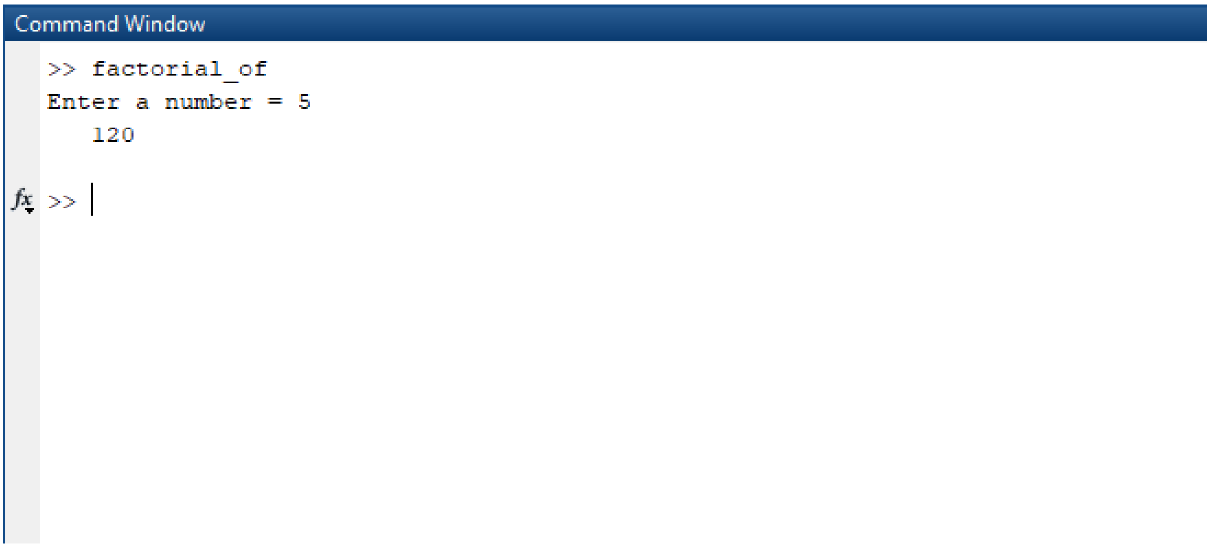
Task 10:

Create an m-file that inputs a number from user and then finds out the factorial of that number. Code:


Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\factorial_of.m
factorial_of.m
+

```
function factorial_of()
%Task-11
%Create an m?file that inputs a number from user and then finds out the factorial
num=input('Enter a number = ');
fact = 1;
for i = 1 : num
    fact = fact * i;
end
disp(fact);
```

Output:

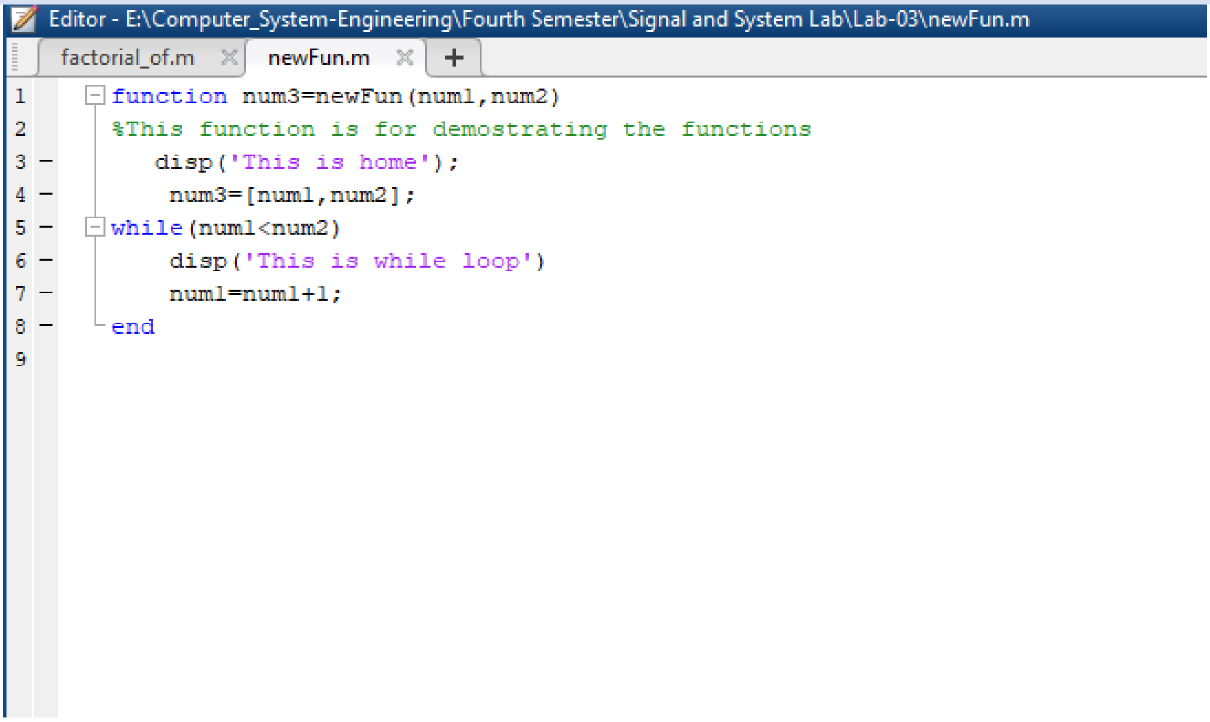


```
>> factorial_of
Enter a number = 5
   120

>> 
```

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.

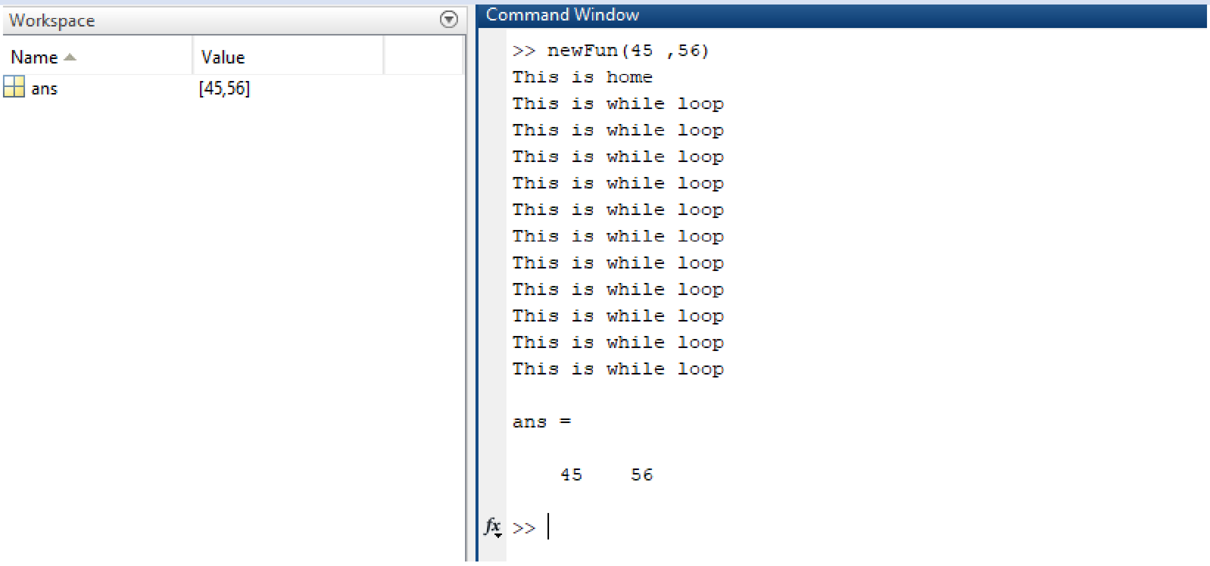
Code:



Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\newFun.m

```
function num3=newFun(num1,num2)
%This function is for demostrating the functions
   disp('This is home');
    num3=[num1,num2];
while(num1<num2)
    disp('This is while loop')
    num1=num1+1;
end
```

Output:

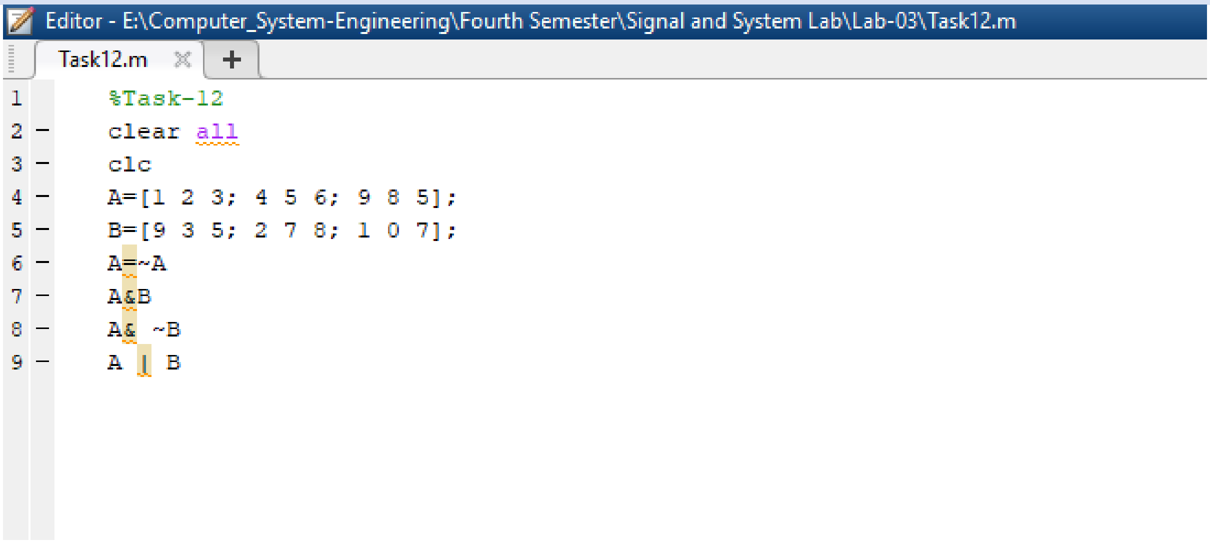


Name	Value
ans	[45,56]

```
>> newFun(45 ,56)
This is home
This is while loop
This is while loop
This is while loop
This is while loop
This is while loop
This is while loop
This is while loop
This is while loop
This is while loop
This is while loop
This is while loop

ans =

    45    56

>> 
```

Task 12:

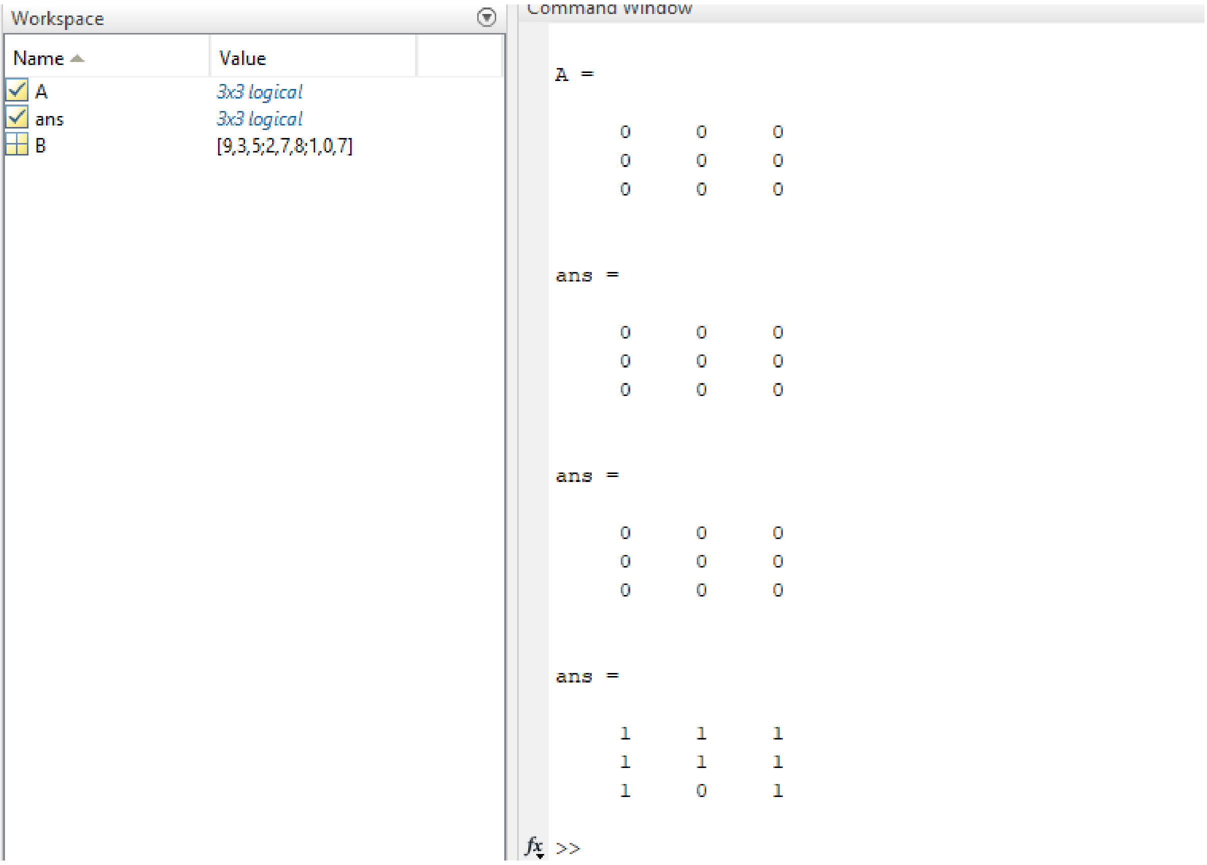
Code:



Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\Task12.m

```
%Task-12
clear all
clc
A=[1 2 3; 4 5 6; 9 8 5];
B=[9 3 5; 2 7 8; 1 0 7];
A=~A
A&B
A& ~B
A | B
```

Output:



Workspace

Name ▲	Value
☑ A	3x3 logical
☑ ans	3x3 logical
▦ B	[9,3,5;2,7,8;1,0,7]

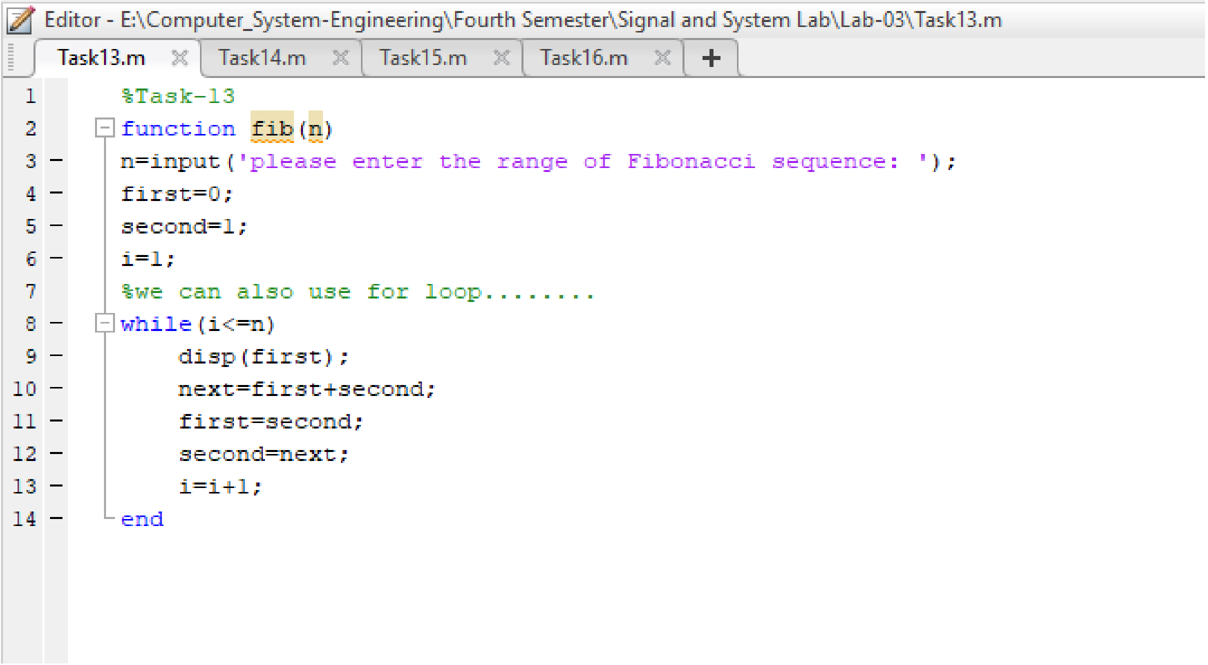
Command Window

```
A =

     0     0     0
     0     0     0
     0     0     0


ans =

     0     0     0
     0     0     0
     0     0     0


ans =

     0     0     0
     0     0     0
     0     0     0


ans =

     1     1     1
     1     1     1
     1     0     1

fx >>
```

Task 13:

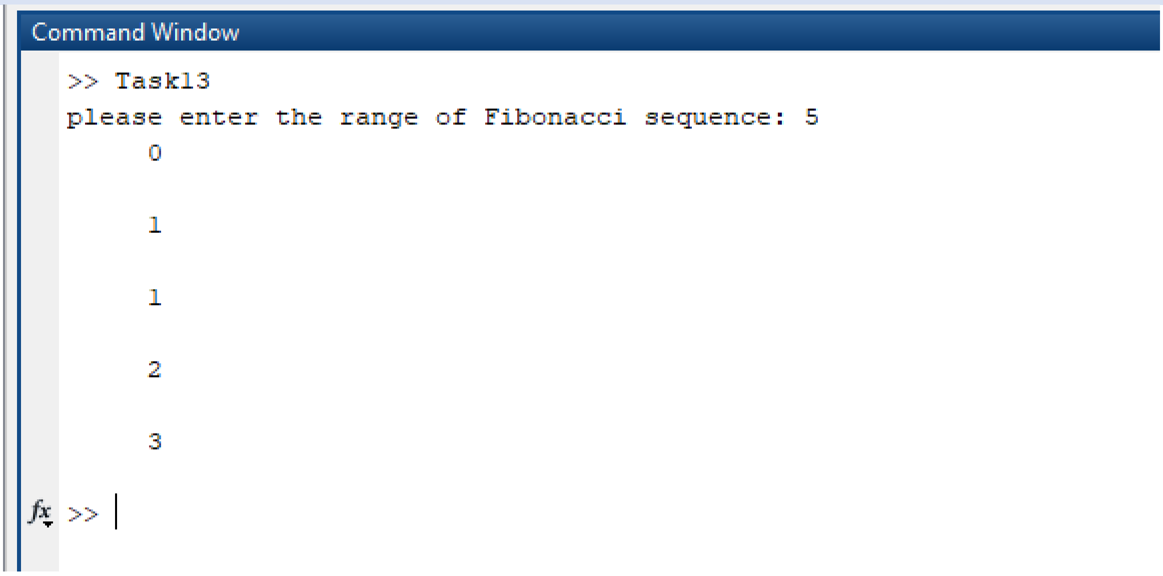
Design a function Fib(N) that takes N as an input and generates a Fibonacci sequence for N. Fibonacci sequence is a tile of squares whose side lengths are successive or each number is the sum of the previous number.

Code:


Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\Task13.m
Task13.m
Task14.m
Task15.m
Task16.m
+

```
%Task-13
function fib(n)
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
```

Output:

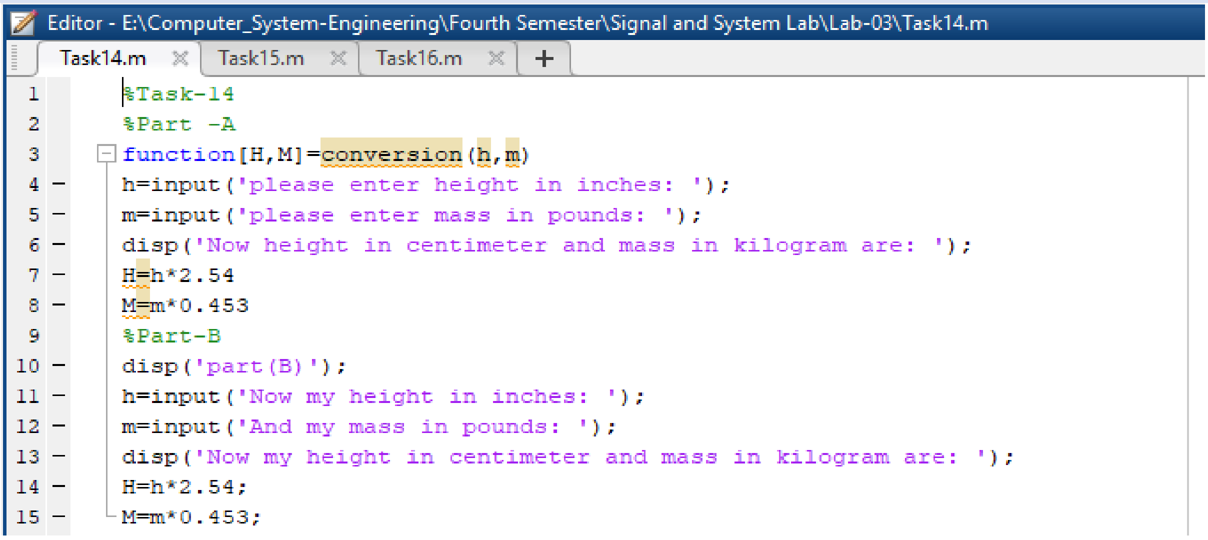


```
>> Task13
please enter the range of Fibonacci sequence: 5
     0

     1

     1

     2

     3

fx >> 
```

Task 14:

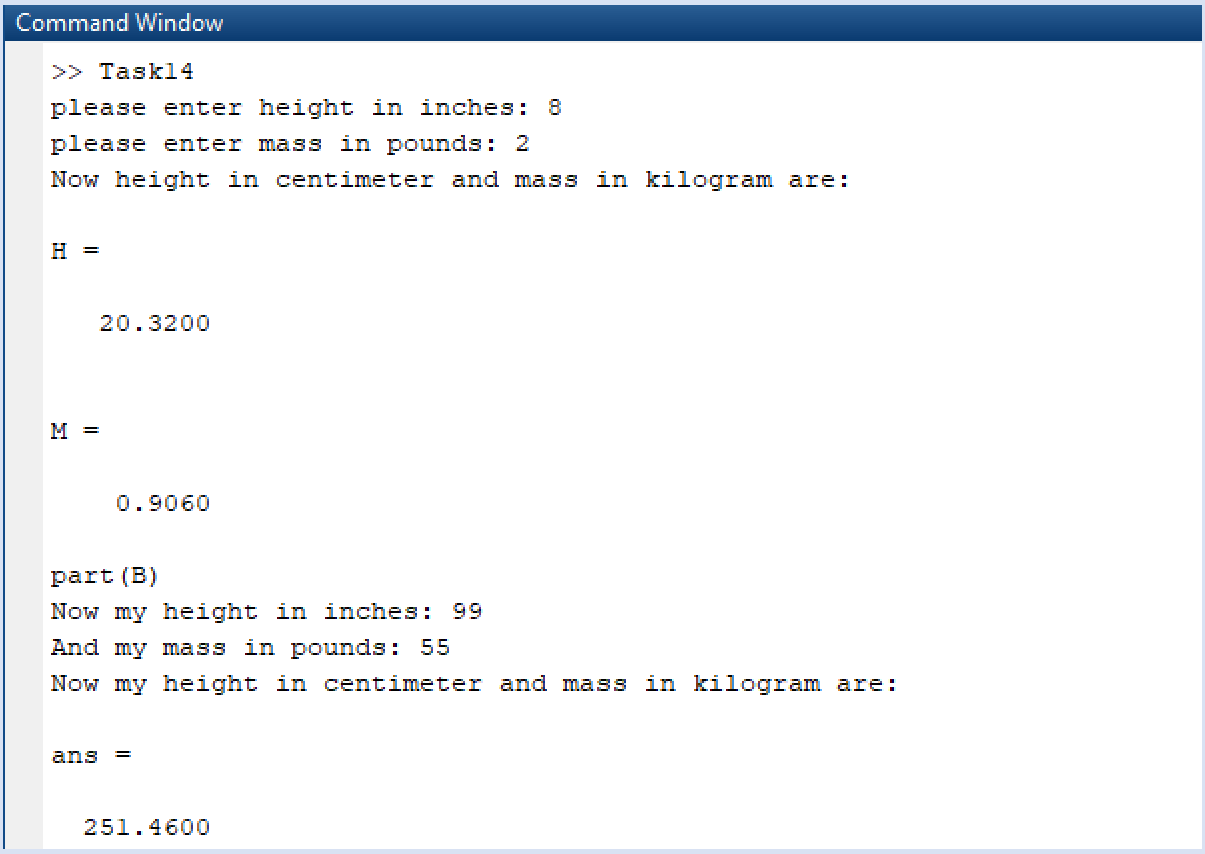
Write a user-defined MATLAB function “Calculate”, with two input and two output arguments that determines the height in centimeters (cm) and mass in kilograms (kg) of a person from his height in inches (in.) and weight in pounds

Code:


Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\Task14.m
Task14.m
Task15.m
Task16.m
+

```
%Task-14
%Part -A
function[H,M]=conversion(h,m)
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
%Part-B
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;
```

Output:

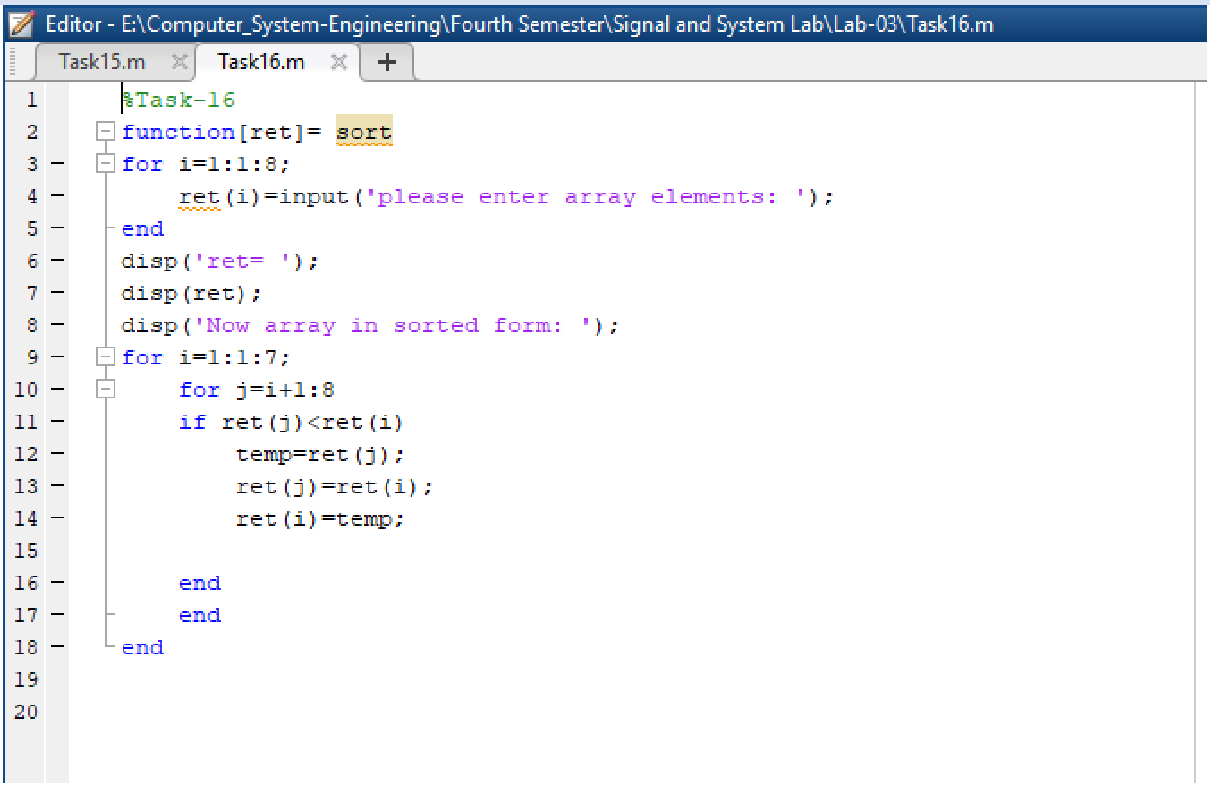


```
>> Task14
please enter height in inches: 8
please enter mass in pounds: 2
Now height in centimeter and mass in kilogram are:

H =

   20.3200


M =

    0.9060

part(B)
Now my height in inches: 99
And my mass in pounds: 55
Now my height in centimeter and mass in kilogram are:

ans =

  251.4600
```

Task 15:

File handling in MatLab. Create files of different formats in MatLab. Use the following commands to create a text file using MatLab commands

Code:

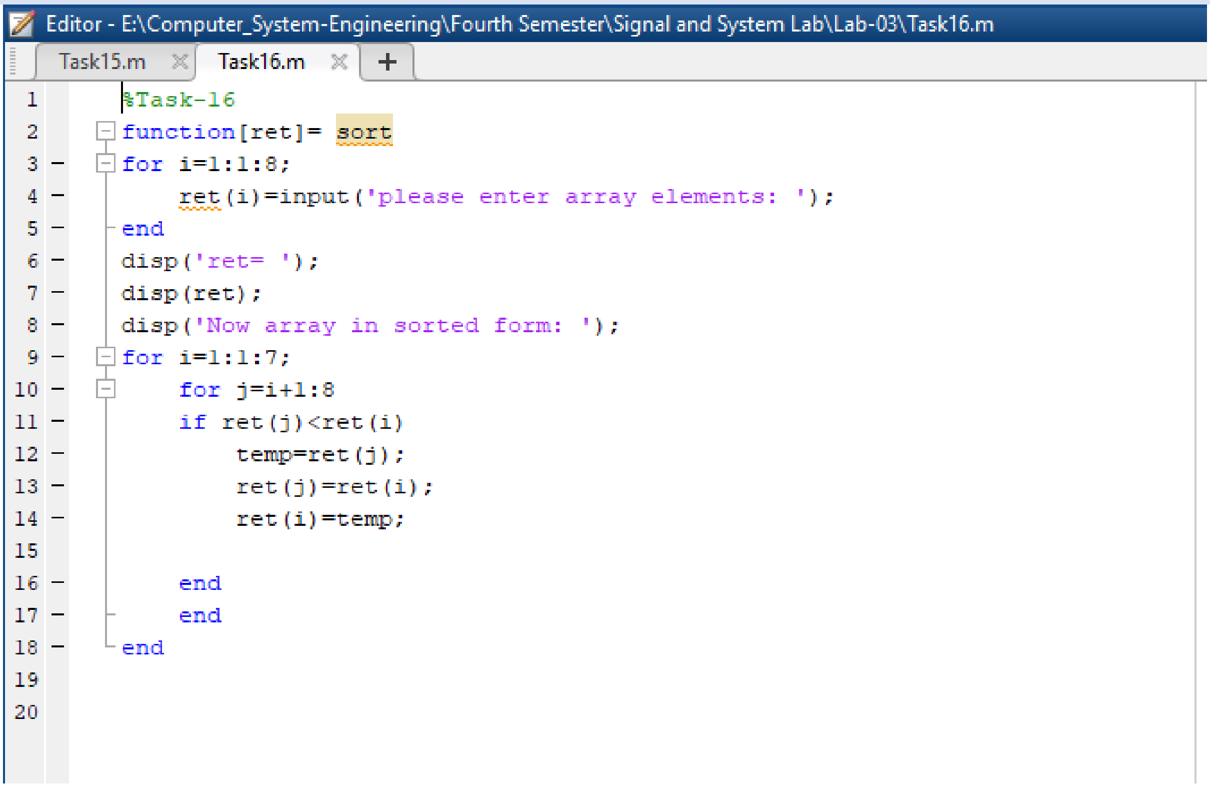

Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\Task16.m
Task15.m
Task16.m
+

```
%Task-16
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
```

Task 16:

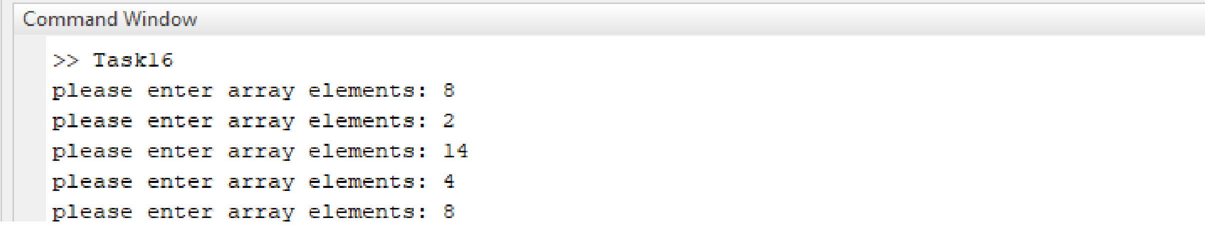
Implement any Sorting and Searching algorithm of your choice by creating user-defined functions in MatLab.

Code:


Editor - E:\Computer_System-Engineering\Fourth Semester\Signal and System Lab\Lab-03\Task16.m
Task15.m
Task16.m
+

```
%Task-16
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
```

Output:



```
>> Task16
please enter array elements: 8
please enter array elements: 2
please enter array elements: 14
please enter array elements: 4
please enter array elements: 8
```