Stuparu Elena Natalia, 422D

TEMA 1

1. Numele implicit al rezultatului în workspace și command prompt

   ans

   ```
   >> x = 3 + 4
   x = 7
   >> ans = x
   ans = 7
   ```

2. Inf

   ```
   >> inf
   ans = Inf
   ```

3. NaN

   ```
   >> NaN
   ans = NaN
   ```

5. +, -, *, /, ^,

   ```
   >> 2+13
   ans = 15
   >> 7-4
   ans = 3
   >> 3*7
   ans = 21
   >> 8/2
   ans = 4
   >> 2^6
   ans = 64
   ```

6. <, <=, >, >=, ==, ~=

   ```
   >> 4<5
   ans = 1
   >> 4<3
   ans = 0
   ```

7. &, | , ~

   ```
   >> 1&0
   ans = 0
   ```

```
>> 1&1
ans = 1
>> 1|0
ans = 1
>> 0|0
ans = 0
>> ~0
ans = 1
>> ~1
ans = 0
```

8. I,j

```
>> i
ans =  0 + 1i
>> j
ans =  0 + 1i
```

9. abs(z), angle(z)

```
>> z = 5 + 10i
z = 5 + 10i
>>  abs(z)
ans = 11.180
>> angle(z)
ans = 1.1071
```

10. acos(x), asin(x), atan(x)

```
>> x = pi/4
x = 0.7854
>> acos(x)
ans = 0.6675
>> asin(x)
ans = 0.9033
>> atan(x)
ans = 0.6658
```

11. acosh(x), asinh(x), atang(x)

```
>> x = pi/3
x = 1.0472
>> acosh(x)
ans = 0.3060
>> asinh(x)
ans = 0.9144
>> atan(x)
ans = 0.8084
```

12. cos(x), sin(x),tan(x)

```
>> x = pi/3
x = 1.0472
>>  cos(x)
ans = 0.5000
>> sin(x)
ans = 0.8660
>>  tan(x)
ans = 1.7321
```

13. cosh(x), sinh(x), tanh(x)

```
>> x = pi/3
x = 1.0472
>> cosh(x)
ans = 1.6003
>> sinh(x)
ans = 1.2494
>> tanh(x)
ans = 0.7807
```

14. conj(z), imag(z), real(z)

```
>> z = 1+i
z =  1 + 1i
>> conj(z)
ans =  1 - 1i
>> imag(z)
ans = 1
>> real(z)
ans = 1
```

15. exp(x), log(x), log10(x)

```
>> exp(2)
ans = 7.3891
>> e^2
ans = 7.3891

>> log(e^2)
ans = 2

>> log10(100)
ans = 2
>> log10(1000)
```

```
ans = 3
```

16. ceil(x), floor(x)

```
>> x = 5.7
x = 5.7000
>> ceil(x)
ans = 6
>> floor(x)
ans = 5
```

17. fix(x), round(x)

```
>> x = 2.8
x = 2.8000
>> fix(x)
ans = 2
>> round(x)
ans = 3
```

18. A*B, A./B

```
>> A = [2,1,3;-2,2,1]
A =

   2   1   3
  -2   2   1

>> B = [2,1;3,2;-2,2]
B =

   2   1
   3   2
  -2   2

>> A*B
ans =

    1  10
    0   4
```

Sau:

```
>> A = [2,4,6,8]
A =

   2  4   6   8

>> B=[2,2,2,2]
```

```
B =

   2  2   2   2
>> A./B
ans =

   1  2   3   4
```

19. **A**.^**B**

```
>> A = [2,4,6,8]
A =

   2  4   6   8

>> B=[2,2,2,2]
B =

   2  2   2   2

>> A.^B
ans =

    4   16  36   64
```

20. x’, A’

```
>> A = [2,1,3;-2,2,1]
A =

   2   1   3
  -2   2   1

>> A'
ans =

   2  -2
   1   2
   3   1

>> a = [2,4,6,8]
a =

   2  4   6   8

>> a'
```

```
ans =

   2
   4
   6
   8
```

21. x.’, A.’

```
>> A = [2,3;-2,1]
A =

   2   3
  -2   1

>> A.'
ans =

   2  -2
   3   1
```

22. **x**=valoare_initial:valoare_pas:valoare_final

```
>> h = 1:3:20
h =

    1    4   7   10   13   16   19
```

23. **x**=linspace(valoare_inițial, valoare_final, n)

```
>> linspace(1,10,5)
ans =

    1.0000   3.2500    5.5000    7.7500   10.0000
```

24. A=[x1 ;x2]

```
>> x1 = [1,2,3]
x1 =

   1  2   3

>> x2 = [4,5,6]
x2 =

   4  5   6

>> A=[x1;x2]
```

```
A =

  1   2   3
  4   5   6
```

25. ones(N,M), zeros(N,M), eye(N,M)

```
>> ones(2,3)
ans =

  1   1   1
  1   1   1

>> zeros(2,3)
ans =

  0   0   0
  0   0   0

>> eye(3,3)
ans =

Diagonal Matrix

  1 0  0
  0   1   0
  0   0   1
```

26. A(i,j)

```
>> A = [2,1,3;-2,2,1;6,7,8]
A =

  2   1   3
 -2   2   1
  6   7   8

>> A(2,3)
ans = 1
```

27. A(I,:), A(i:j,:), **A**(i:k:j,:), **A**([i,j,k],:), **A**(:,j), **A**(:,i:j), **A**(:,i:k:j), **A**(:,[i,j,k])

```
>> A = [2,1,3;-2,2,1;6,7,8]
A =
  2   1   3
 -2   2   1
  6   7   8

>> A(2,:)
ans =

 -2   2  1

>> A(2:3,:)
ans =

 -2   2   1
  6   7   8

>> A(1:1:2,:)
ans =

  2   1   3
 -2   2   1

>> A([1,2,3],:)
ans =

  2   1   3
 -2   2   1
  6   7   8

>> A(:,2)
ans =

  1
  2
  7

>> A(:,2:3)
ans =

  1   3
  2   1
  7   8
```

28. who

```
>> who
Variables visible from the current scope:

A    B    a    ans  h    i    x  x1  x2 z
```

29. size(A)

```
>> A = [2,1,3;-2,2,1;6,7,8]
A =

   2   1   3
  -2   2   1
   6   7   8

>> size(A)
ans =

   3  3
```

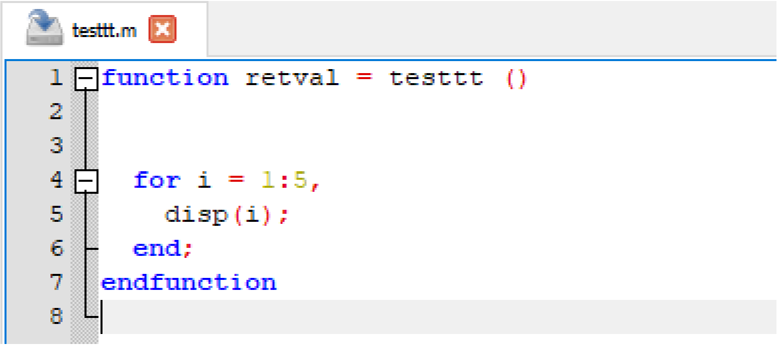
30. length(x)

```
>> x = [1,2,3,4,5]
x =

   1  2   3   4   5

>> length(x)
ans = 5
```

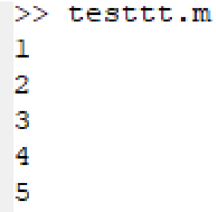
31. Bucla for

```
for i=val_initiala:pas:val_finala

    Instructiuni

            End
```

Stuparu Elena Natalia, 422D



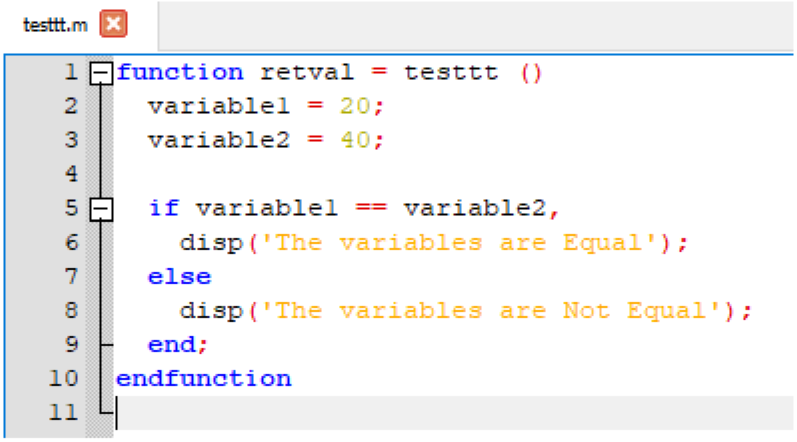
```
function retval = testtt ()


  for i = 1:5,
    disp(i);
  end;
endfunction

```



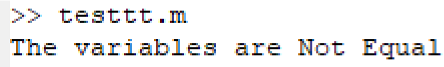
```
>> testtt.m
1
2
3
4
5
```

32. Secvență de cauzalitate if/ else/elseif

```
 if conditie
     instructiuni
else/elseif
     instructiuni
end
```



```
function retval = testtt ()
  variable1 = 20;
  variable2 = 40;

  if variable1 == variable2,
    disp('The variables are Equal');
  else
    disp('The variables are Not Equal');
  end;
endfunction

```

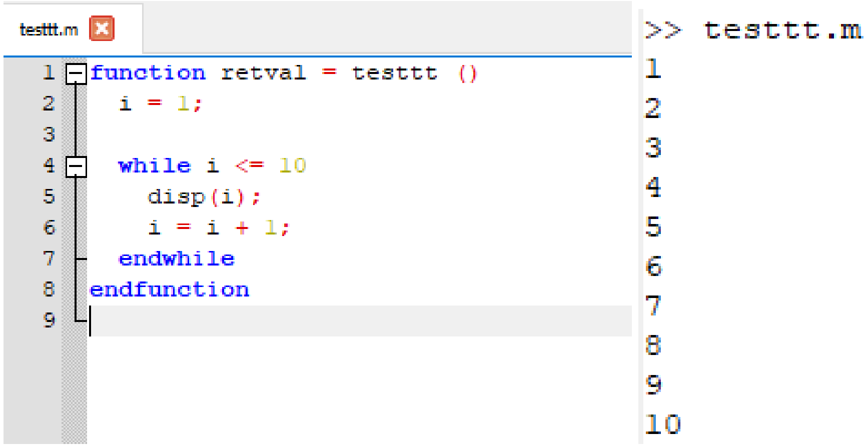


```
>> testtt.m
The variables are Not Equal
```

33. Bucla while

```
While (conditie)
     Instructiuni
end
```

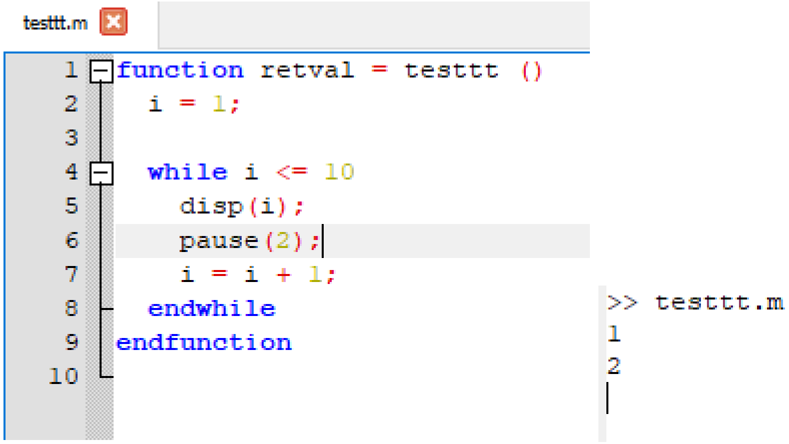
Stuparu Elena Natalia, 422D



```
function retval = testtt ()
  i = 1;

  while i <= 10
    disp(i);
    i = i + 1;
  endwhile
endfunction

```

```
>> testtt.m
1
2
3
4
5
6
7
8
9
10
```

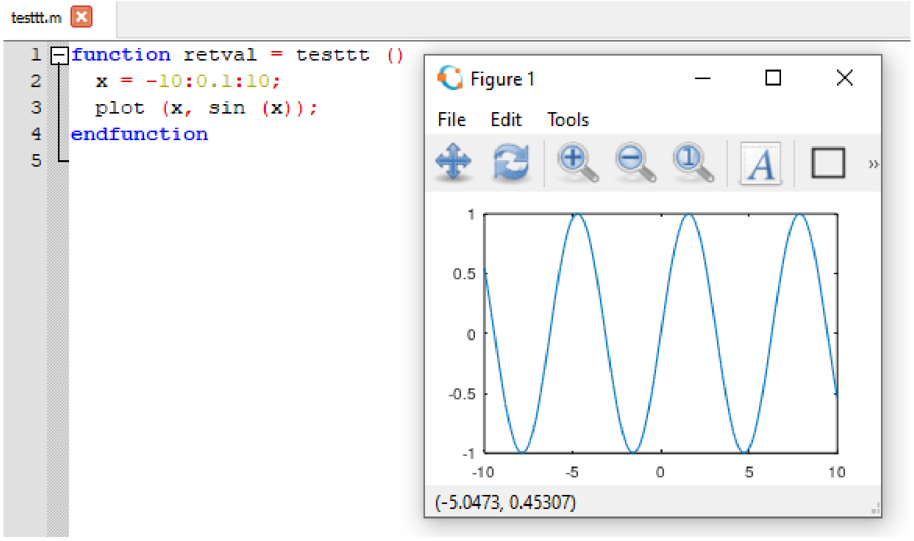
34. Pause, pause(n)



```
function retval = testtt ()
  i = 1;

  while i <= 10
    disp(i);
    pause(2);
    i = i + 1;
  endwhile
endfunction

```

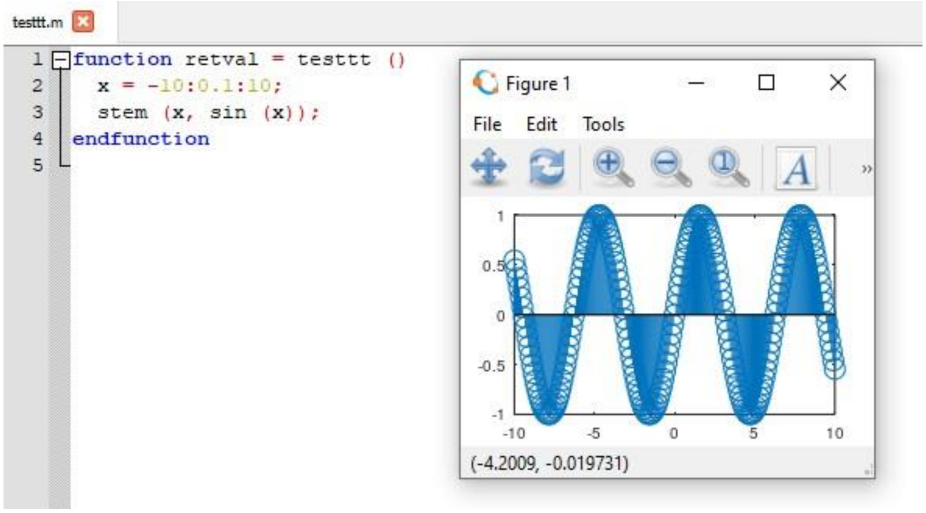
```
>> testtt.m
1
2
```

35. plot(x,y), stem(x,y)



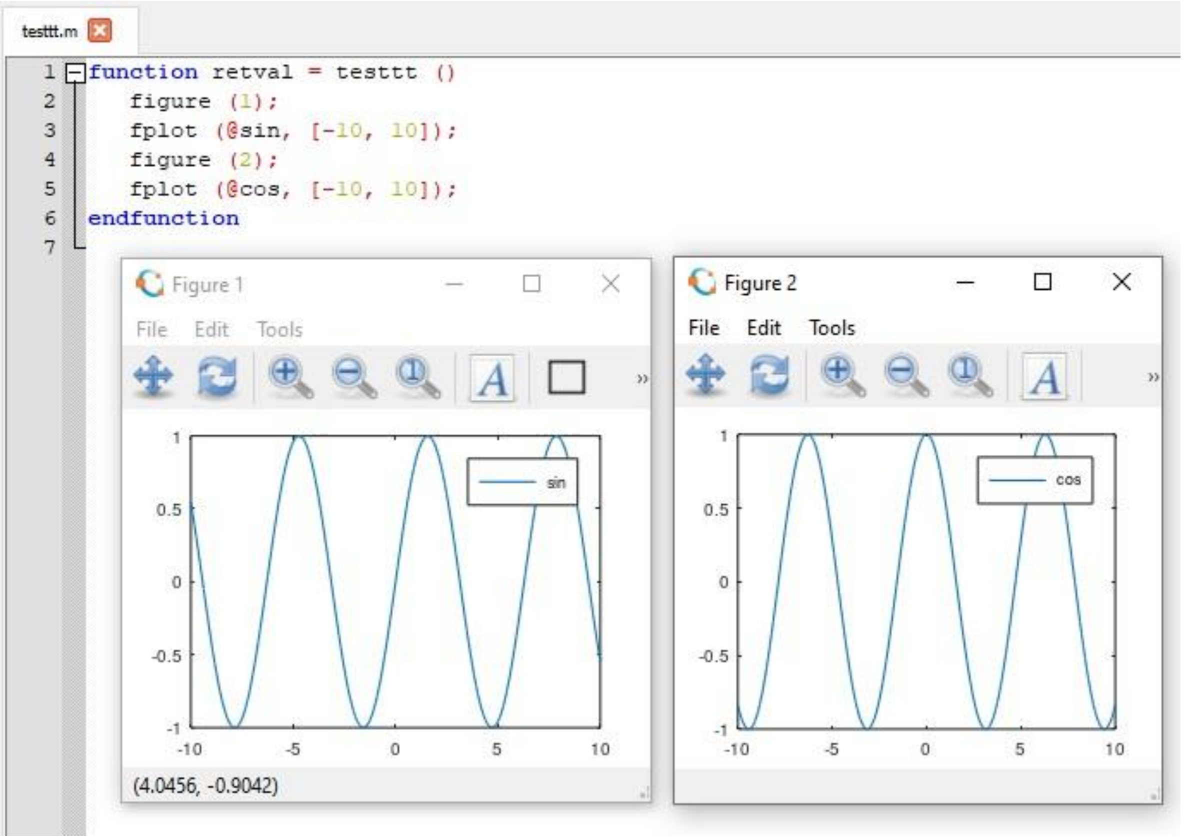
```
function retval = testtt ()
  x = -10:0.1:10;
  plot (x, sin (x));
endfunction

```

Stuparu Elena Natalia, 422D



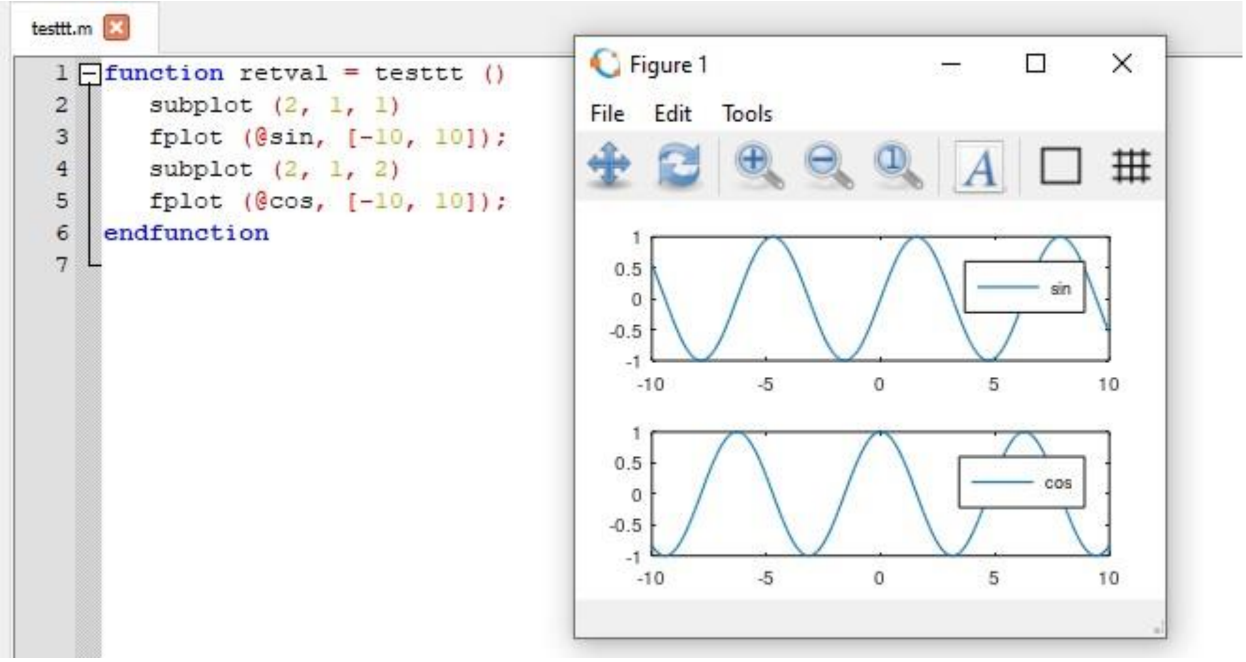
```
function retval = testtt ()
  x = -10:0.1:10;
  stem (x, sin (x));
endfunction
```

36. figure



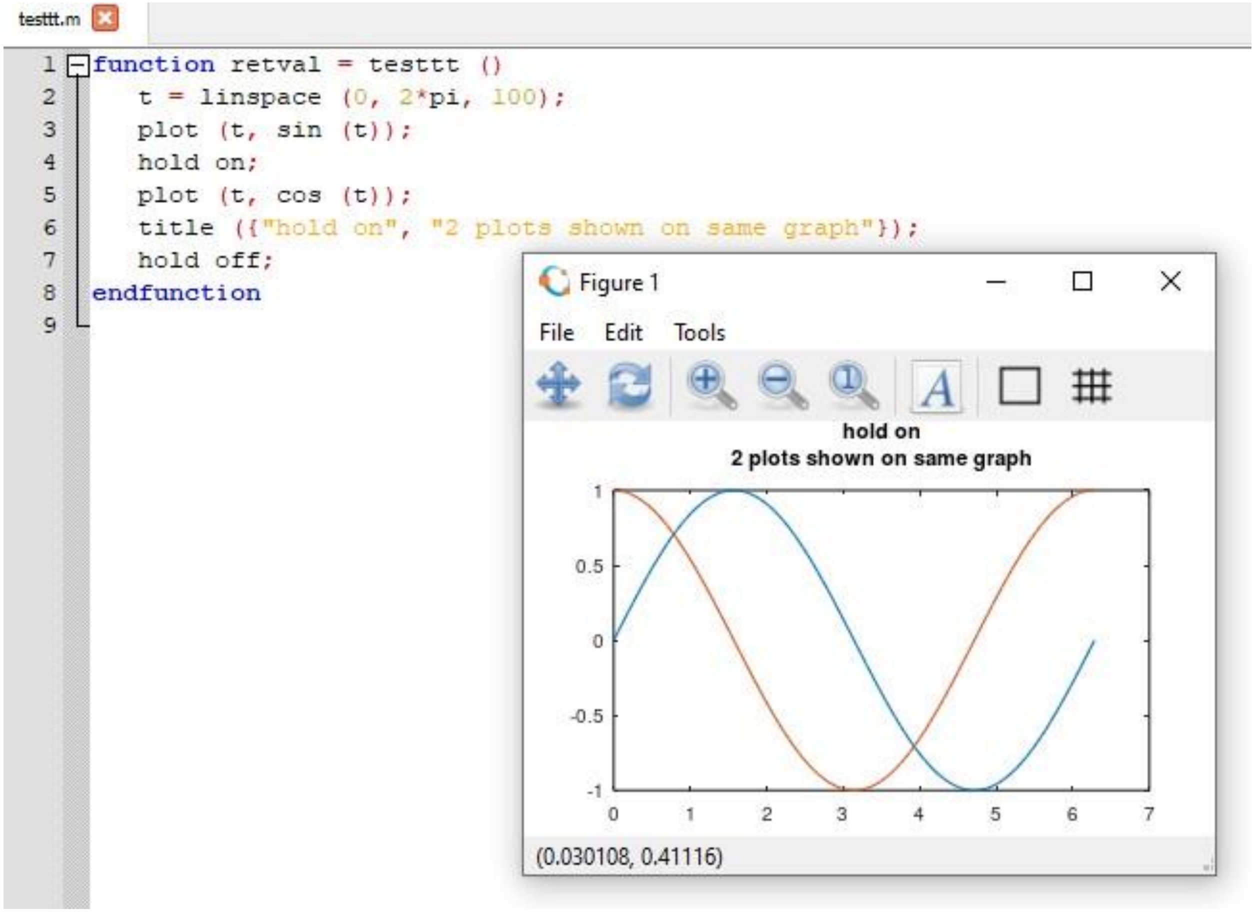
```
function retval = testtt ()
    figure (1);
    fplot (@sin, [-10, 10]);
    figure (2);
    fplot (@cos, [-10, 10]);
endfunction
```

37. Subplot



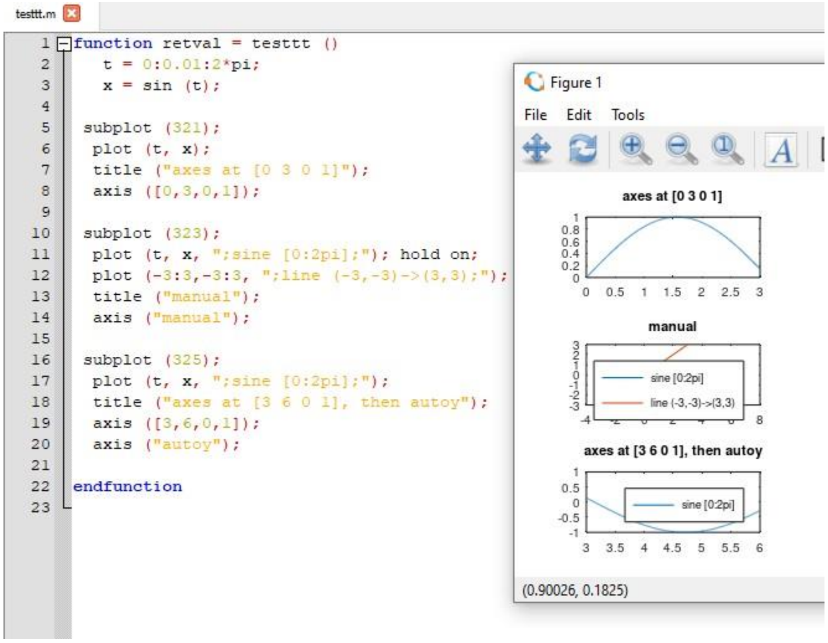
```
function retval = testtt ()
    subplot (2, 1, 1)
    fplot (@sin, [-10, 10]);
    subplot (2, 1, 2)
    fplot (@cos, [-10, 10]);
endfunction
```

38. hold on, hold off



```
function retval = testtt ()
    t = linspace (0, 2*pi, 100);
    plot (t, sin (t));
    hold on;
    plot (t, cos (t));
    title ({"hold on", "2 plots shown on same graph"});
    hold off;
endfunction
```

39. axis ([x_inf x_sup y_inf y_sup])



```
function retval = testtt ()
   t = 0:0.01:2*pi;
   x = sin (t);

 subplot (321);
  plot (t, x);
  title ("axes at [0 3 0 1]");
  axis ([0,3,0,1]);

 subplot (323);
  plot (t, x, ";sine [0:2pi];"); hold on;
  plot (-3:3,-3:3, ";line (-3,-3)->(3,3);");
  title ("manual");
  axis ("manual");

 subplot (325);
  plot (t, x, ";sine [0:2pi];");
  title ("axes at [3 6 0 1], then autoy");
  axis ([3,6,0,1]);
  axis ("autoy");

endfunction
```

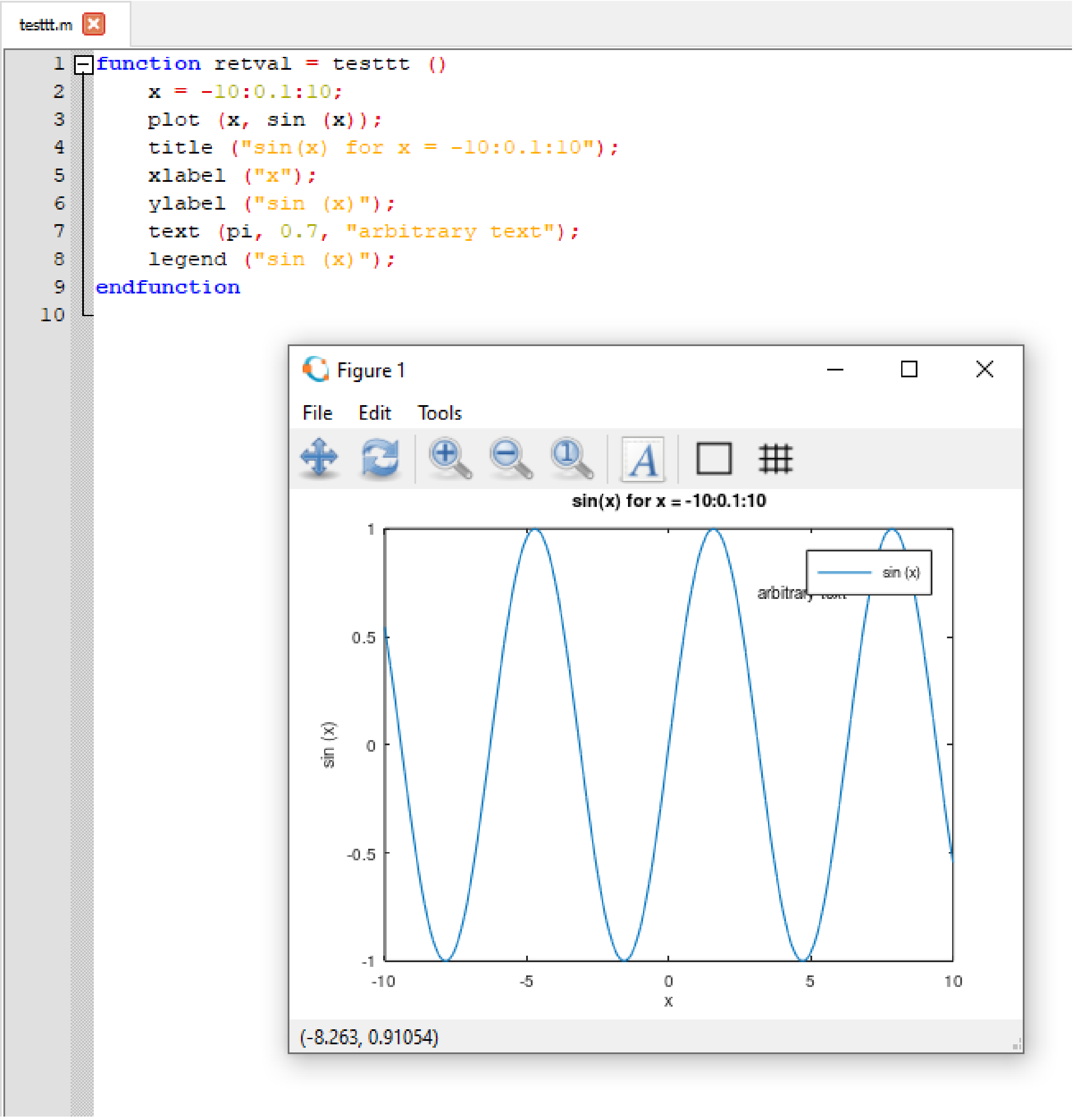
40. Titlurile axelor, titlul figurii, legenda figurii

xlabel(´nume_axa_OX´),

ylabel(nume_axa_OY),

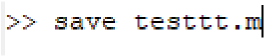
title(nume_figura),

legend (parametrii)



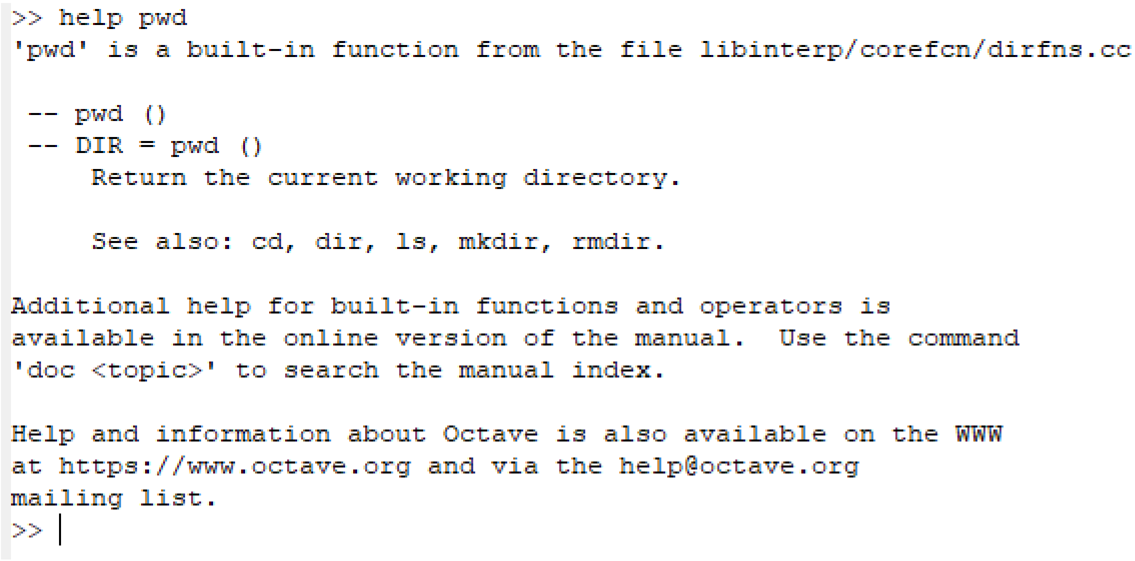
```
function retval = testtt ()
    x = -10:0.1:10;
    plot (x, sin (x));
    title ("sin(x) for x = -10:0.1:10");
    xlabel ("x");
    ylabel ("sin (x)");
    text (pi, 0.7, "arbitrary text");
    legend ("sin (x)");
endfunction
```

41. save, load



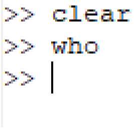
```
>> save testtt.m
```

42. help nume_functie



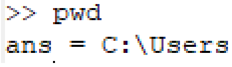
```
>> help pwd
'pwd' is a built-in function from the file libinterp/corefcn/dirfns.cc

 -- pwd ()
 -- DIR = pwd ()
     Return the current working directory.

     See also: cd, dir, ls, mkdir, rmdir.

Additional help for built-in functions and operators is
available in the online version of the manual.  Use the command
'doc <topic>' to search the manual index.

Help and information about Octave is also available on the WWW
at https://www.octave.org and via the help@octave.org
mailing list.
>> |
```

43. clear, clf



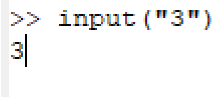
```
>> clear
>> who
>> |
```

44. cd, pwd



```
>> pwd
ans = C:\Users
```

45. input(“introduceti de la tastatura>>”)



```
>> input("3")
3|
```

Tema 1.2 : Scrieți un program care să genereze pentru 2 secunde o nota muzicală la alegere. Salvați comenzile într-un fișier denumit Nume_Prenume_Grupa_T1.2.

```
function retval = testtt ()
   Fs=8000;
   Ts=1/Fs;
   t=[0:Ts:2];
   F_A=440; %Frecventa notei este 440 Hz
   A=sin(2*pi*F_A*t);
   sound(A,Fs);
endfunction
```

Stuparu Elena Natalia, 422D

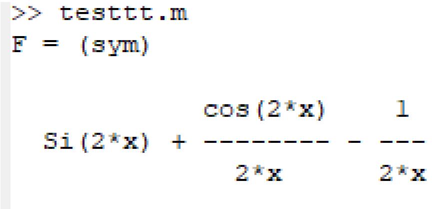
Tema 1.3

Scrieți un program Matlab/Octave în care să calculați integrala[1] funcției $x^2(t)$, unde:

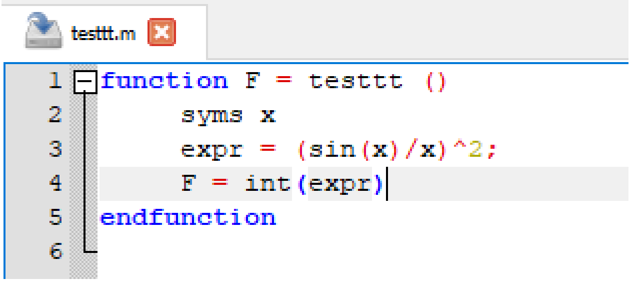
$$x(t) = \frac{sin(t)}{t}$$

Salvați comenzile într-un fișier denumit Nume_Prenume_Grupa_T1.3.

```
function F = testtt ()

    syms x

    expr = (sin(x)/x)^2;

    F = int(expr)

endfunction
```



```
>> testtt.m
F = (sym)

              cos(2*x)     1
  Si(2*x) + -------- - ---
                2*x       2*x
```



```
function F = testtt ()
        syms x
        expr = (sin(x)/x)^2;
        F = int(expr)
endfunction

```

Tema 1.4

```
          clear all;
          history -c;
          clc;

          function x = rampa (t)

          x=t*(t>=0);
          end
          rampa(3)
          function x = treapta (t)
            x=1*(abs(t)<0.5);
          end
          treapta(0.2)
          function x = grafic (t)
            x=(t<-1)(0)+(t>=-1 & t<=0)(-t-1)+(t>0 & t<=1)(2.5*t-1)+(t>1 & t<=2)(-0.5*t+2)+(t>2 & t<=3)(-t+3)+(t>3)(0);
            end
          grafic(0)
save file
```