1. Problems from Gilat, Ch. 1.10

1.2 a) Calculate

$$23\left(-8+\frac{\sqrt{607}}{3}\right)+\left(\frac{40}{8}+4.7^2\right)^2$$

```
INPUT

res2a = 23 * (-8 + sqrt(607)/3) + (40/8 + 4.7^2)^2
res2b = nthroot(509,3) - 4.5^2 + log(200)/1.5 + sqrt(75)

OUTPUT

res2a =  738.75
res2b = -0.073190
```

1.4 a) Calculate

$$\cos\left(\frac{5\pi}{6}\right)\sin^2\left(\frac{7\pi}{8}\right)+\frac{\tan\left(\frac{\pi}{6}\ln 8\right)}{\sqrt{7}+2}$$

```
INPUT

res4a = cos(5*pi/6) * sin(7*pi/8)^2 + tan(pi/6*log(8)) / (sqrt(7) + 2)
res4b = cos(3*pi/5)^2 + tan(pi*log(6)/5) / (8*7/2)

OUTPUT

res4a =  0.28462
res4b =  0.17038
```

1.6 a) Define the variables x and z as $x = 5.3$, and $z = 7.8$, then evaluate:

$$\frac{xz}{(x/z)^2}+14x^2-0.8z^2$$

```
INPUT

x = 5.3
z = 7.8

res6a = (x*z) / (x/z)^2 + 14*x^2 - 0.8*z^2
res6b = x^2*z - z^2*x + (x/z)^2 - sqrt(z/x)

OUTPUT

x =  5.3000
z =  7.8000

res6a =  434.13
res6b = -104.10
```

1.10 a) The following is a trignonometric identity:

$$\sin(3x) = 3\sin x - 4\sin^3 x$$

Verify that the identity is correct by calculating each side of the equation, substituting $x = 7\pi/20$.

```
INPUT

x = 7*pi/20

lhsa = sin(3*x)
rhsa = 3*sin(x) - 4*sin(x)^3

res10a = lhsa - rhsa

format long
lhsa
rhsa

format short

OUTPUT

x =  1.0996
lhsa = -0.15643
rhsa = -0.15643
res10a = -4.4409e-16
lhsa = -0.156434465040231
rhsa = -0.156434465040230
```

1.16) The distance d from a point (x_0, y_0) to a line $Ax + By + C = 0$ is given by:

$$d = \frac{|Ax_0 + By_0 + C|}{\sqrt{A^2 + B^2}}$$

Determine the distance of the point $(-3, 4)$ from the line $2x - 7y - 10 = 0$. First define the variables A, B, C, x_0 and y_0, and then calculate d. (Use the `abs` and `sqrt` functions).

```
INPUT

A = 2
B = -7
C = -10
x0 = 3
y0 = -4

d = abs(A*x0 + B*y0 + C) / sqrt(A^2 + B^2)

OUTPUT

A =  2
B = -7
C = -10
x0 =  3
y0 = -4
d =  3.2967
```

2. Problems from Gilat, Ch. 2.11

2.1 Create a row vector that has the elements 6, $8 \cdot 3$, 81, $e^{2.5}$, $\sqrt{65}$, $\sin(\pi/3)$ and 23.05.

```
INPUT

res1 = [6 8*3 81 exp(2.5) sqrt(65) sin(pi/3) 23.05]

OUTPUT

res1 =

    6.00000   24.00000   81.00000   12.18249    8.06226    0.86603   23.05000
```

2.2 Create a column vector that has the elements 44, 9, $\ln(51)$, 2^3, 0.1 and $5\tan(25°)$.

```
INPUT

res2 = [44; 9; log(51); 2^3; 0.1; 5*tand(25)]

OUTPUT

res2 =

   44.00000
    9.00000
    3.93183
    8.00000
    0.10000
    2.33154
```

2.4 Create a column vector in which the first element is 18, the elements decrease with increments of -4, and the last element is -22. (Recall that a column vector can be created by the transpose of a row vector.)

```
INPUT

res4 = [18:-4:-22]'

OUTPUT

res4 =

   18
   14
   10
    6
    2
   -2
   -6
  -10
  -14
  -18
  -22
```

2.8 Create a vector, name it `Afirst`, that has 13 elements in which the first is 3, the increment is 4 and the last element is 51. Then, using the colon symbol, create a new vector, call it `Asecond`, that has seven elements. The first four elements are the the first four elements of the vector `Afirst`, and the last three are the last three elements of the vector `Afirst`.

```
INPUT

Afirst = 3:4:51

Asecond(1:7) = [Afirst(1:4) Afirst(11:13)]

OUTPUT

Afirst =

   3   7  11  15  19  23  27  31  35  39  43  47  51

Asecond =

    3    7   11   15   43   47   51
```

2.9 Create the matrix shown below by using the vector notation for creating vectors with constant spacing and/or the `linspace` command when entering the rows.

$$B = \begin{bmatrix} 0 & 4 & 8 & 12 & 16 & 20 & 24 & 28 \\ 69 & 68 & 67 & 66 & 65 & 64 & 63 & 62 \\ 1.4 & 1.1 & 0.8 & 0.5 & 0.2 & -0.1 & -0.4 & -0.7 \end{bmatrix}$$

```
INPUT

B = [0:4:28; 69:-1:62; linspace(1.4,-0.7,8)]

OUTPUT

B =

 Columns 1 through 7:

   0.00000    4.00000    8.00000   12.00000   16.00000   20.00000   24.00000
  69.00000   68.00000   67.00000   66.00000   65.00000   64.00000   63.00000
   1.40000    1.10000    0.80000    0.50000    0.20000   -0.10000   -0.40000

 Column 8:

  28.00000
  62.00000
  -0.70000
```

2.10 Using the colon symbol, create a 3×5 matrix (assign to a variable named `msame`) in which all of the elements are the number 7.

```
INPUT

msame(1:3, 1:5) = 7

OUTPUT

msame =

   7   7   7   7   7
   7   7   7   7   7
   7   7   7   7   7
```

2.14 Create the following matrix, A:

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \end{bmatrix}$$

```
INPUT

A = reshape(1:15, 5, 3)'

OUTPUT

A =

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

Use the matrix A to:

a) Create a five-element row vector named `va` that contains the elements of the first row of A.

```
INPUT

va = A(1,:)

OUTPUT

va =

     1     2     3     4     5
```

b) Create a three-element row vector named `vb` that contains the elements of the third column of A.

```
INPUT

vb = A(:,3)

OUTPUT

vb =

     3
     8
    13
```

c) Create an eight-element row vector names `vc` that contains the elements of the second row of A and the fourth column of A.

INPUT

```
vc = [A(2,:) A(:,4)']
```

OUTPUT

```
vc =

     6     7     8     9    10     4     9    14
```

d) Create a six-element row vector named `vd` that contains the elements of the first and fifth columns of A.

INPUT

```
vd = [A(:,1)' A(:,5)']
```

OUTPUT

```
vd =

     1     6    11     5    10    15
```

2.18 Using the `zeros`, `ones` and `eye` commands, create the following arrays:

a)

$$\begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

INPUT

```
a = [zeros(2,3) ones(2,3)]
```

OUTPUT

```
a =

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

b)

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

INPUT

```
b = [ones(4,1) eye(4) zeros(4,1)]
```

OUTPUT

```
b =

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

c)

$$\begin{bmatrix} 1 & 1 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}$$

INPUT

```
c = [ones(1,2); zeros(2,2); ones(1,2)]
```

OUTPUT

```
c =

   1   1
   0   0
   0   0
   1   1
```

3. Writing simple `octave/MATLAB` functions and scripts

3b) `threeoutargs`:

Create a `MATLAB` function **`threeoutargs`** which has two input arguments, x and y, and which returns *three* output arguments which are $x + y$, $x - y$ and $(x + y)/2$, respectively. Ensure that you save the definition of your function as the file `threeoutargs.m`.

Sample implementation

```
function [res1 res2 res3] = threeoutargs(x, y)
   res1 = x + y;
   res2 = x - y;
   res3 = (x + y) / 2;
end
```

3c) `sintaylor`

The Taylor series expansion for $\sin x$ is given by

$$\sin x = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \cdots$$

Create a `MATLAB` function **`sintaylor`** with a header as follows

```
function res = sintaylor(x, nmax, epsi)
```

and which computes an approximation of $\sin(x)$ using the following truncated version of the series

$$\sin x = \sum_{n=0}^{n_{\max}} \frac{(-1)^n}{(2n+1)!} x^{2n+1}$$

`sintaylor` should return as soon as either one of the conditions have been met

- All of the terms in the truncated series have been evaluated.
- An individual term in the series has an absolute value that is $\leq$ `epsi` (but include that term in the sum)

Save your code in the file `sintaylor.m`

Sample implementation

```
function res = sintaylor(x, nmax, epsi)
%% sintaylor(x, nmax, epsi)
%%
%% Evaluates Taylor series for sin(x) about x=0 using
%% a maximum of nmax + 1 terms, or until the current
%% term in the expansion has an absolute value <= epsi
%%
   res = 0;
   for n = 0:nmax
      term = ((-1)^n/factorial(2*n + 1)) * x^(2*n + 1);
      res = res + term;
      if abs(term) <= epsi
         break;
      end
   end
end
```