CENG236 Numerical Computations
Labwork 2

1 Generating Vectors

1.1 Colon Notation

```
>> 1:5
1   2   3   4   5

>> 1:0.5:5
1.0000    1.5000    2.0000    2.5000    3.0000    3.5000    4.0000    4.5000    5.0000
```

The examples above generate vectors that start from 1 and end at 5. The difference between them is how the generated numbers increase: the first example increases the numbers by 1 explicitly, while the second example increases the numbers by 0.5.

1.2 linspace Function

```
>> a = linspace(0,5)
a = 0.00000   0.05051   0.10101   0.15152   0.20202   0.25253 [...]
    [...] 4.84848   4.89899   4.94949   5.00000

>> b = linspace(0,5,10)
b = 0.00000   0.55556   1.11111   1.66667   2.22222   2.77778
    3.33333   3.88889   4.44444   5.00000
```

linspace function generates linearly spaced vectors. It generates 100 equally spaced numbers when used with 2 parameters, or N equally spaced numbers when used with 3 parameters.

```
>> a = linspace(-5*pi, 10*pi, 201);
>> b = sin(a);
>> plot(a, b)
```

2 Symbolic Variables and Expressions

So far, all the variables we have used are of type **double**. These variables, because of limitations, can only store a limited amount of digits after decimal seperator, therefore they are not exact. For example:

```
>> sin(pi)
ans = 1.2246e-16
```

The answer to $sin(\pi)$ must be 0, but since in this case `pi` variable is a floating point number, the answer is an estimation of $sin(\pi)$.

Unlike floating point numbers, symbolic numbers are exact representations. Instead of using floating point numbers, we can get the correct answer with symbolic numbers:

```
>> sin(sym(pi))
ans = 0
```

Algebraic manipulations such as factoring and expanding can be accomplished with symbolic variables:

```
>> syms x
>> p = (x+2)*(x-1)
p = (x + 2)*(x - 1)

>> expand(p)
ans = x^2 + x - 2

>> factor(ans)
ans = [ x + 2, x - 1]

>> factor(sym(236))
ans = [ 2, 2, 59]

>> syms a b c
>> expand(log(a^2*b^3/c^5))
ans = log((a^2*b^3)/c^2)
>> expand(log(a^2*b^3/c^5),'IgnoreAnalyticConstraints',true)
ans = 2*log(a) + 3*log(b) - 5*log(c)
```

2.1 Symbolic Functions

Symbolic expressions can be evaluated as functions in MATLAB:

```
>> syms x
>> f=x/(1+x^2)
```

Now you can evaluate this function f for any given value of x with **subs** function:

```
>> subs(f, x, 2)
ans = 2/5
```

You can also use **subs** function with vectors. The returning value will be the same size as the input vector.

```
>> a = linspace(-10,10,101);
>> b = subs(f, x, a);
>> plot(a, b)
```

You can get roots of a given equation by **solve** function:

```
>> syms x
>> solve(x^2-5*x+6)
ans = 2 3
```

2.2 Exercise

Convert the given formulae to symbolic functions and plot them in the given ranges.

- $(x-1)*(x-2)*(x-3)$, Range: [-10 10]
- $-x^3+3x^2-4$, Range: [-5 5]
- $x^4-4x^3-2x^2+12x-3$, Range: [-3 5]

3 Inline Functions

The **inline** function converts a given **string with any number of variables** into a function that can be evaluated later. For example, the function $f(a,b)=\sqrt{a^2+b^2}$ can be turned into an inline function by:

```
>> f = inline('sqrt(a^2 + b^2)')
```

In this example, f is defined to be an inline function which takes two parameters a and b. Then it can either be called with **feval** function, or by its name:

```
>> f(3, 4)
ans =  5
>> feval(f, 5, 12)
ans =  13
```

3.0.1 Exercise

Write MATLAB commands that convert the given formulae into inline functions and then evaluate their results with different values.

- $\frac{x}{1+x^2}$
- $\frac{x^2-y^2}{x^2+y^2}$
- $e^{1+sin(x)}$

3.1 Plotting Inline Functions

Lets assume we are given $f(x)=x*sin(x)$ function. We can plot it by:

```
>> a = linspace(-10*pi, 10*pi, 500);
>> f = inline('x.*sin(x)')
>> plot(a, f(a))
```

Do note that we have used `.*` operator instead of `*` operator. Why?

3.1.1 Exercise

Plot the first and third functions you have defined in the previous exercise between the ranges [0 10] and [0 4π].

4 Differences and Derivatives

diff function calculates differences and derivatives of given values and functions.

```
>> a = [1 1 4 9 13 14]
>> diff(a)
ans = 0 3 5 4 1

>> syms x
>> f = x^3 + 3*x^2 + 5
>> diff(f)
ans = 3*x^2 + 6*x
>> diff(ans)
ans = 6*x + 6

>> g=sin(x)
>> diff(g)
ans = cos(x)
>> diff(ans)
ans = -sin(x)
```

5 Integrals

int function computes integrals of given expressions.

```
>> syms x
>> int(-2*x/(1 + x^2)^2)
ans = 1/(x^2 + 1)

>> syms x z
>> int(x/(1 + z^2), x)
ans = x^2/(2*(z^2 + 1))
>> int(x/(1 + z^2), z)
ans = x*atan(z)
```