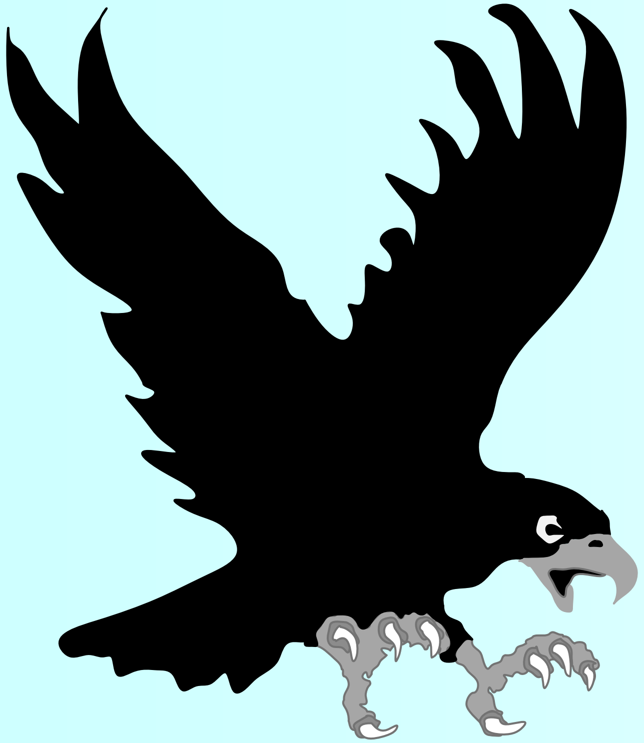


Programlama -1

“Symbolic Mathematics”

Dr. Cahit Karakuş, 2020

“Symbolic” toolbox allows you to:

- İfadeleri sembolik veri türleriyle sembolik biçimde girilir.
- Sembolik ifadeleri genişletir veya basitleştirir.
- Sembolik kökleri, limitleri, minimumları, maksimumları vb. bulur.
- Sembolik işlevleri ayırt eder ve bütünleştirir.
- Taylor fonksiyon serileri oluşturur (diğer araçların yanı sıra).
- Cebirsel ve diferansiyel denklemleri sembolik olarak çözer.
- Eşzamanlı denklemleri çözer (hatta bazı doğrusal olmayan).
- Değişken hassas aritmetik yapar.
- Sembolik fonksiyonların grafik gösterimlerini oluşturur.

Manipulating symbolic expressions

- numden()
- expand()
- factor()
- collect()
- simplify()
- simple()
- poly2sym()

numden()

- The numden() command is used to separate the numerator and denominator of a quotient.
- Bir bölümün payını ve paydasını ayırmak için numden() komutu kullanılır. Aşağıdaki ifadede numden() kullanın:

```
y=2*(x + 3)^2/(x^2 + 6*x + 9)
[numerator, denominator]=numden(y)
```

```
numerator= 2*(x + 3)^2
denominator= (x^2 + 6*x + 9)
```

expand()

- expand() is used to expand an expression by expanding the products of factors in an expression.

- Expand the numerator of y:

```
expand(numerator)
```

```
ans= 2*x^2 + 12*x + 18
```

factor()

- The factor() command is used to factor an expression into a product of terms.
- factor() komutu, bir ifadeyi bir terimler çarpımına ya da çarpanlarınaa ayırmak için kullanılır.

Example: Factor the expression for the denominator.

```
factor(denominator)
```

```
ans=
(x + 3)^2
```

simplify()

- The simplify() command uses the Maple simplification algorithm to simplify each part of an expression.
- Enter the expression:

```
b=sym('3*a - (a + 3)*(a - 3)^2');
simplify(b)
```

```
ans= 12*a - a^3 + 3*a^2 - 27
```

simple()

- The simple() command executes many techniques to simplify an expression; its `ans` is the simplest expression.

```
simple(b)
```

```
ans = 3*a - (a + 3)*(a - 3)^2
```

Note: The simple() command displays the results of all simplification methods; only the final result, i.e., the answer (**ans**) is shown here.

poly2sym()

- The poly2sym() function uses an array of coefficients to create a polynomial:

```
a=[1,3,2];
b=poly2sym(a)
```

```
b=
x^2 + 3*x + 2
```

- The function sym2poly() is the inverse of poly2sym().

solve()

- The solve() function sets an expression equal to zero, then solves the equation for its roots.

```
E1=x^2 - 9
solve(E1)
```

```
ans=
3
-3
```

```
clear all
close all
a=[1 3 2]
b=poly2sym(a)
c=solve(b)
```

Hands on

```
solve('a*x^2 + b*x + c')
```

```
ans =
1/2/a*( -b + (b^2 - 4*a*c)^(1/2))
1/2/a*( -b - (b^2 - 4*a*c)^(1/2))
```

Note: The expression is in single quotes; if the symbolic variables in the expression have not been defined previously, then the single quotes are necessary.

Hands on

```
solve('a*x^2 + b*x + c', 'a')
```

```
ans =
-(b*x + c)/x^2
```

Note that this solves the expression for “a”.

Systems of equations

- You can use solve() to find the solution of a system of equations.

```
one=sym('3*x + 2*y - z = 10');
two=sym('-x + 3*y + 2*z = 5');
three=sym('x - y - z = -1');
[x,y,z]=solve(one,two,three)
```

```
clear all
close all
syms x y z

d1='3*x+2*y-z=10'
d2='-x+3*y+2*z=5'
d3='x-y-z=-1'
[x,y,z]=solve(d1,d2,d3)
```

x=	y=	z=
-2	5	-6

Note that the solve function produces symbolic output. You can change the output to numerical values with the double() command.

Substitution with subs()

- The subs() command allows you to substitute a symbol with another symbol or assign a number to a variable.

```
syms a b c x y;
quadratic = a*x^2 + b*x + c;
yquadratic = subs(quadratic,x,y)
```

```
yquadratic =
a*y^2 + b*y + c
```

Multiple substitutions

- You can use the subs() command to do multiple substitutions in one command. This is done by grouping the variables and their substitutes (other variables or numerical values) in braces.

```
subs(symbolic_function, {substitutant}, {substitute})
```

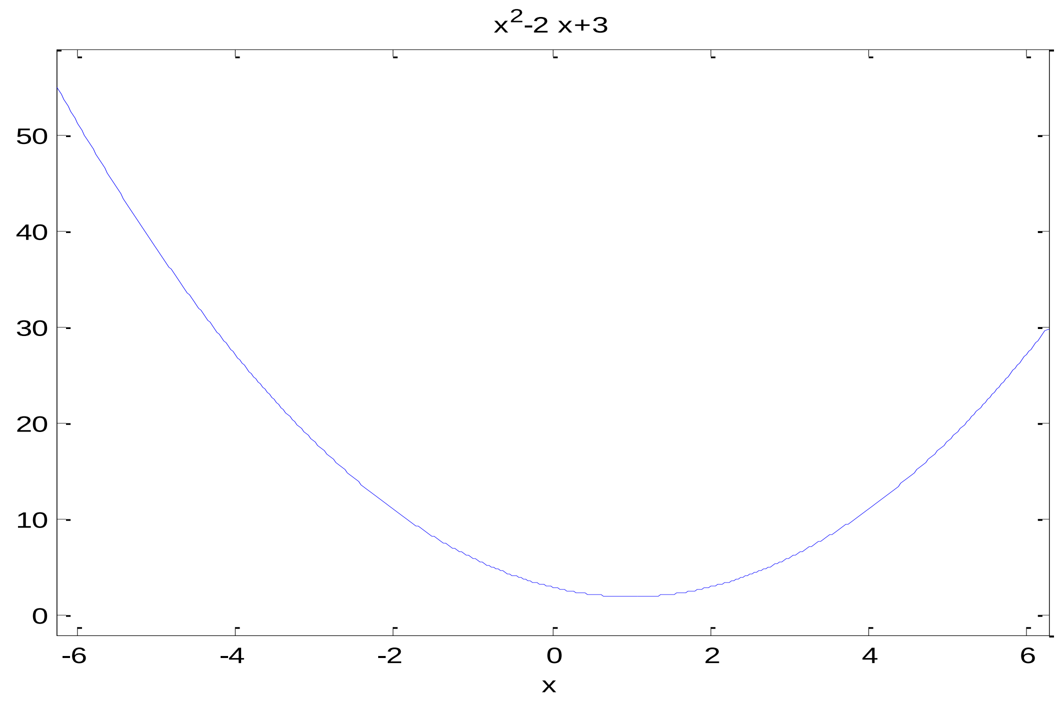
```
subs(quadratic, {a,b,c,x}, {1,2,3,4})
```

```
ans =
27
```

Plotting symbolical functions

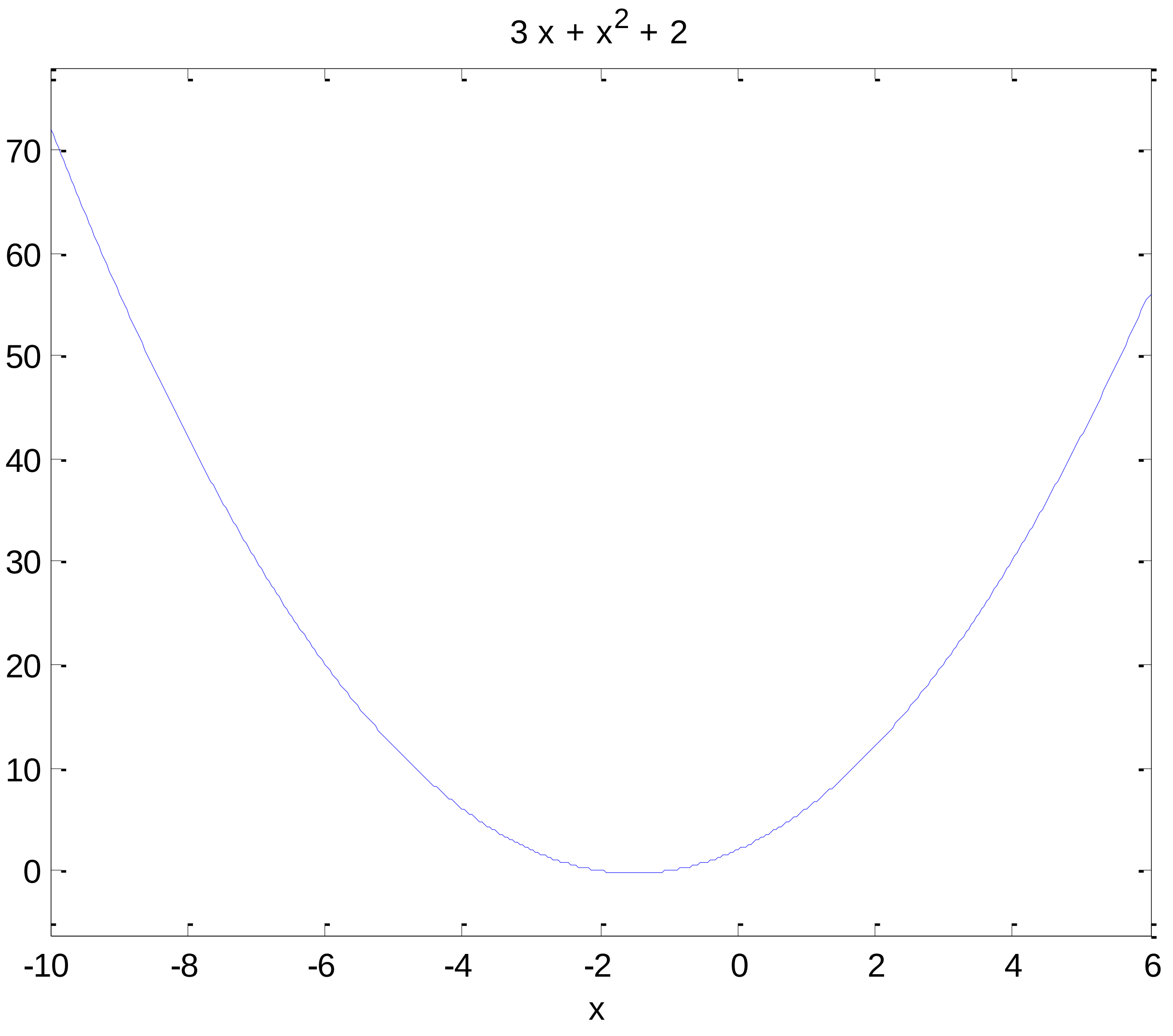
- Plotting symbolic functions in MATLAB is done with the ezplot() set of commands.
- The syntax of ezplot() when y is a function of x is:
  `ezplot(y)`

```
y = sym('x^2 - 2*x + 3');
ezplot(y)
```



EZPLOT

```
clear all
close all
a=[1 3 2]
b=poly2sym(a)
c=solve(b)
ezplot(b,[-10,6])
```



Differentiation

- MATLAB allows you to differentiate symbolic functions with the diff() command.
- The syntax of diff(), when f is a symbolic function of the variable x and n is the order of the derivative to be determined, is:

```
diff(f,'x' ,n)
```

```
diff(y,'x',1)
```

```
ans =
2*x-2
```

Try a second derivative. Your result should be 2.

Integration

- MATLAB allows you to integrate symbolic functions with the int() command. Either definite integrals, with assigned numeric or symbolic bounds, or indefinite integrals (note that when you compute indefinite integrals, the constant of integration does *not* appear explicitly).

```
int(y,'x',a,b)
```

```
int(y,'x')
```

```
ans =
1/3*x^3-x^2+3*x
```

Try a definite integral for 1<x<2. The result should be 7/3.

Integration constant

- If you want to display the integration constant when applying the int() function, you can do the following:

```
syms x y a
y = 2*x;
int(y,'a','x')
ans = x^2-a^2
```

Exercises

- Use the symbolic toolbox to solve the following system of equations:

  $x + 2y - z = 4$

  $3x + 8y + 7z = 20$

  $2x + 7y + 9z = 23$

- The velocity of a car is $v = t^2 - 3t + 5$. Find the displacement for $1 < t < 5$ and the acceleration at $t=1.5$. Plot the equations for distance, velocity, and acceleration on one graph.

Summary

- Creating Symbolic Expressions
  - sym('x'), syms x, expressions i.e. e=sym('m*c^2')
- Manipulation
  - numden, expand, factor, collect, simplify, simple, poly2sym
- Solutions
  - solve, subs
- Plotting
  - ezplot
- Differentiation
  - diff(y, 'x', n)
- Integration
  - int(y, 'x', a, b)