```
>> syms a b c x                    Define a, b, c, and x as symbolic variables.
>> S=sqrt(a*x^2 + b*x + c)         Create the symbolic expression
S =                                √(ax² + bx + c), and assign it to
(a*x^2+b*x+c)^(1/2)
>> pretty(S)                       The pretty command displays
                  2          1/2   the expression in a math format.
              (a x  + b x + c)
```

11.3 *SOLVING ALGEBRAIC EQUATIONS*

A single algebraic equation can be solved for one variable, and a system of equations can be solved for several variables with the `solve` function.

Solving a single equation:

An algebraic equation can have one or several symbolic variables. If the equation has one variable, the solution is numerical. If the equation has several symbolic variables, a solution can be obtained for any of the variables in terms of the others. The solution is obtained by using the `solve` command, which has the form

`h = solve(eq)` or `h = solve(eq,var)`

- The argument `eq` can be the name of a previously created symbolic expression, or an expression that is typed in. When a previously created symbolic expression `S` is entered for `eq`, or when an expression that does not contain the = sign is typed in for `eq`, MATLAB solves the equation `eq` = 0.
- An equation of the form $f(x) = g(x)$ can be solved by typing the equation (including the = sign) as a string for `eq`.
- If the equation to be solved has more than one variable, the `solve(eq)` command solves for the default symbolic variable (see Section 11.1.3). A solution for any of the variables can be obtained with the `solve(eq,var)` command by typing the variable name for `var`.
- If the user types `solve(eq)`, the solution is assigned to the variable `ans`.
- If the equation has more than one solution, the output `h` is a symbolic column vector with a solution at each element. The elements of the vector are symbolic objects. When an array of symbolic objects is displayed, each row is enclosed with square brackets (see the following examples).

The following examples illustrate the use of the `solve` command.

```
>> syms a b x y z
>> h=solve(exp(2*z)-5)
h =
log(5)/2
>> S=x^2-x-6
S =
x^2-x-6
>> k=solve(S)
k =
 -2
  3
>> solve('cos(2*y)+3*sin(y)=2')
ans =
     pi/2
     pi/6
 (5*pi)/6
>> T= a*x^2+5*b*x+20
T =
a*x^2+5*b*x+20
>> solve(T)
ans =
 -(5*b+5^(1/2)*(5*b^2-16*a)^(1/2))/(2*a)
 -(5*b-5^(1/2)*(5*b^2-16*a)^(1/2))/(2*a)
>> M = solve(T,a)
M =
-(5*b*x+20)/x^2
```

Define `a`, `b`, `x`, `y`, and `z` as symbolic variables.

Use the `solve` command to solve $e^{2z} - 5 = 0$.

The solution is assigned to `h`.

Create the symbolic expression $x^2 - 6 - x$, and assign it to `S`.

Use the `solve(S)` command to solve $x^2 - 6 - x$.

The equation has two solutions. They are assigned to `k`, which is a column vector with symbolic

Use the `solve` command to solve $\cos(2y) + 3\sin(y) = 2$. (The equation is typed as a string in the command.)

The solution is assigned to `ans`.

Create the symbolic expression $ax^2 + 5bx + 20$, and assign it to `T`.

Use the `solve(S)` command to solve $T = 0$.

The equation $T = 0$ is solved for the variable x, which is the default variable.

Use the `solve(eq,var)` command to solve $T = 0$.

The equation $T = 0$ is solved for the variable a.

- It is also possible to use the `solve` command by typing the equation to be solved as a string, without having the variables in the equation first created as symbolic objects. However, if the solution contains variables (when the equation has more than one variable), the variables do not exist as independent symbolic objects. For example:

```
>> ts=solve('4*t*h^2+20*t-5*g')
ts =
(5*g)/(4*h^2+20)
```

The expression $4th^2 + 20t - 5g$ is typed in the `solve` command.

The variables t, h, and g were not created as symbolic variables before the expression was typed in the `solve` command.

MATLAB solves the equation $4th^2 + 20t - 5g = 0$ for t.

The equation can also be solved for a different variable. For example, a solution for *g* is obtained by:

```
>> gs=solve('4*t*h^2+20*t-5*g','g')

gs =
(4*t*h^2)/5 + 4*t
```

Solving a system of equations:

The `solve` command can also be used for solving a system of equations. If the number of equations and the number of variables are the same, the solution is numerical. If the number of variables is greater than the number of equations, the solution is symbolic for the desired variables in terms of the other variables. A system of equations (depending on the type of equations) can have one or several solutions. If the system has one solution, each of the variables for which the system is solved has one numerical value (or expression). If the system has more than one solution, each of the variables can have several values.

The format of the `solve` command for solving a system of *n* equations is:

```
output = solve(eq1,eq2,....,eqn)
```

or

```
output = solve(eq1,eq2,...,eqn,var1,var2,...,varn)
```

- The arguments `eq1,eq2,...,eqn` are the equations to be solved. Each argument can be a name of a previously created symbolic expression, or an expression that is typed in as a string. When a previously created symbolic expression `S` is entered, the equation is `S = 0`. When a string that does not contain the = sign is typed in, the equation is `expression = 0`. An equation that contains the = sign must be typed as a string.
- In the first format, if the number of equations *n* is equal to the number of variables in the equations, MATLAB gives a numerical solution for all the variables. If the number of variables is greater than the number of equations *n*, MATLAB gives a solution for *n* variables in terms of the rest of the variables. The variables for which solutions are obtained are chosen by MATLAB according to the default order (Section 11.1.3).
- When the number of variables is greater than the number of equations *n*, the user can select the variables for which the system is solved. This is done by using the second format of the `solve` command and entering the names of the variables `var1,var2,...,varn`.

The `output` from the `solve` command, which is the solution of the system, can have two different forms. One is a cell array and the other is a structure. A cell array is an array in which each of the elements can be an array. A structure is an array in which the elements (called fields) are addressed by textual field designators. The fields of a structure can be arrays of different sizes and types. Cell arrays and structures are not presented in detail in this book, but a short explanation is given below so that the reader will be able to use them with the `solve` command.

When a cell array is used in the output of the `solve` command, the command has the following form (in the case of a system of three equations):

```
[varA, varB, varC] = solve(eq1,eq2,eq3)
```

- Once the command is executed, the solution is assigned to the variables `varA`, `varB`, and `varC`, and the variables are displayed with their assigned solution. Each of the variables will have one or several values (in a column vector) depending on whether the system of equations has one or several solutions.
- The user can select any names for `varA`, `varB`, and `varC`. MATLAB assigns the solution for the variables in the equations in alphabetical order. For example, if the variables for which the equations are solved are x, u, and t, the solution for t is assigned to `varA`, the solution for u is assigned to `varB`, and the solution for x is assigned to `varC`.

The following examples show how the `solve` command is used for the case where a cell array is used in the output:

```
>> syms x y t
>> S=10*x+12*y+16*t;
>> [xt yt]=solve(S, '5*x-y=13*t')
xt =
2*t
yt =
-3*t
```

Define `x`, `y`, and `t` as symbolic variables.

Assign to `S` the expression $10x + 12y + 16t$.

Use the `solve` command to solve the system: $10x + 12y + 16t = 0$
$5x - y = 13t$

Output in a cell array with two cells named `xt` and `yt`.

The solutions for x and y are assigned to `xt` and `yt`, respectively.

In the example above, notice that the system of two equations is solved by MATLAB for x and y in terms of t, since x and y are the first two variables in the default order. The system, however, can be solved for different variables. As an example, the system is solved next for y and t in terms of x (using the second form of the `solve` command:

```
>> [tx yx]=solve(S,'5*x-y=13*t',y,t)
tx =
x/2
yx =
-(3*x)/2
```

The variables for which the system is solved (y and t) are entered.

The solutions for the variables for which the system is solved are assigned in alphabetical order. The first cell has the solution for t, and the second cell has the solution for y.

When a structure is used in the output of the `solve` command, the command has the form (in the case of a system of three equations)

```
AN = solve(eq1,eq2,eq3)
```

- `AN` is the name of the structure.
- Once the command is executed the solution is assigned to `AN`. MATLAB displays the name of the structure and the names of the fields of the structure, which are the names of the variables for which the equations are solved. The size and the type of each field is displayed next to the field name. The content of each field, which is the solution for the variable, is not displayed.
- To display the content of a field (the solution for the variable), the user has to type the address of the field. The form for typing the address is: `structure_name.field_name` (see example below).

As an illustration the system of equations solved in the last example is solved again using a structure for the output.

```
>> syms x y t
>> S=10*x+12*y+16*t;
>> AN=solve(S,'5*x-y=13*t')
AN =
   x: [1x1 sym]
   y: [1x1 sym]
>> AN.x
ans =
2*t
>> AN.y
ans =
-3*t
```

Use the `solve` command to solve the system: $10x + 12y + 16t = 0$, $5x - y = 13t$

MATLAB displays the name of the structure `AN` and the names of its fields `x` and `y` (size and type), which are the names of the variables for which the equations are solved.

Type the address of the field `x`.

The content of the field (the solution for x) is displayed.

Type the address of the field `y`.

The content of the field (the solution for y) is displayed.

Sample Problem 11-1 shows the solution of a system of equations that has two solutions.

Sample Problem 11-1: Intersection of a circle and a line

The equation of a circle in the $x\,y$ plane with radius R and its center at point (2, 4) is given by $(x-2)^2 + (y-4)^2 = R^2$. The equation of a line in the plane is given by $y = \frac{x}{2} + 1$. Determine the coordinates of the points (as a function of R) where the line intersects the circle.

Solution

The solution is obtained by solving the system of the two equations for x and y in terms of R. To show the difference in the output between using cell array and structure output forms of the `solve` command, the system is solved twice. The