Computer since

First Stage

Lec3

Control flow

MS.c Mortada Sabri

MS.c Esraa Hussei

Control flow

MATLAB has four control flow structures: the if statement, the for loop, the while loop, and the switch statement.

The ‘‘if...end’’ structure MATLAB supports the variants of “if” construct.

- if .... end
- if ....else ……... end
- if .... elseif ……….. else …….. end

Syntax

```
if expression
   statements
elseif expression
   statements
else
   statements
end
```

Example

- if .... end

```
discr = 5;
if discr < 0
  disp(’Warning: discriminant is negative, roots are imaginary’);
end
```

- ➢ if ....else ……... end

```
 discr = 5;
  if discr < 0
    disp(’Warning: discriminant is negative, roots are imaginary’);
  else
    disp(’Roots are real, but may be repeated’)
 end
```

- ➢ if .... elseif ……….. else …….. end

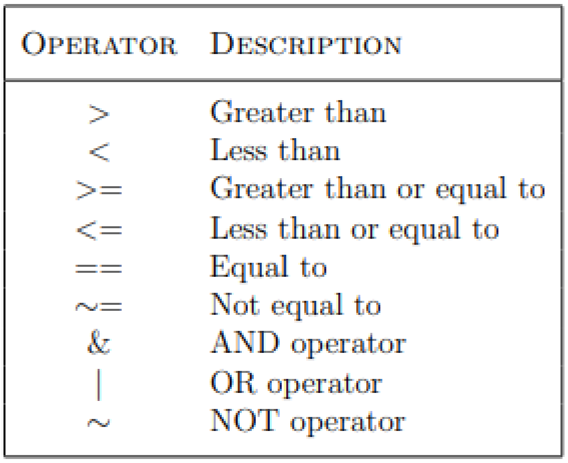
```
discr = 5;
if discr < 0
   disp(’Warning: discriminant is negative, roots are imaginary’);
elseif discr == 0
   disp(’Discriminant is zero, roots are repeated’)
else disp(’Roots are real’)
end
```

It should be noted that:

- elseif has no space between else and if (one word)
- no semicolon (;) is needed at the end of lines containing if, else, end
- Indentation of if block is not required, but facilitate the reading.
- the end statement is required

Relational and logical operators

A relational operator compares two numbers by determining whether a comparison is true or false



OPERATOR	DESCRIPTION
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to
==	Equal to
~=	Not equal to
&	AND operator
|	OR operator
~	NOT operator

Note that the "equal to" relational operator consists of two equal signs (==) (with no space between them), since = is reserved for the assignment operator.

Loop types

A loop statement allows us to execute a statement or group of statements multiple times. The drawing shows the general form of a loop statement for most programming languages.

For loop

A for loop is a repetition control structure that allows you to efficiently write a loop that needs to execute a specific number of times.

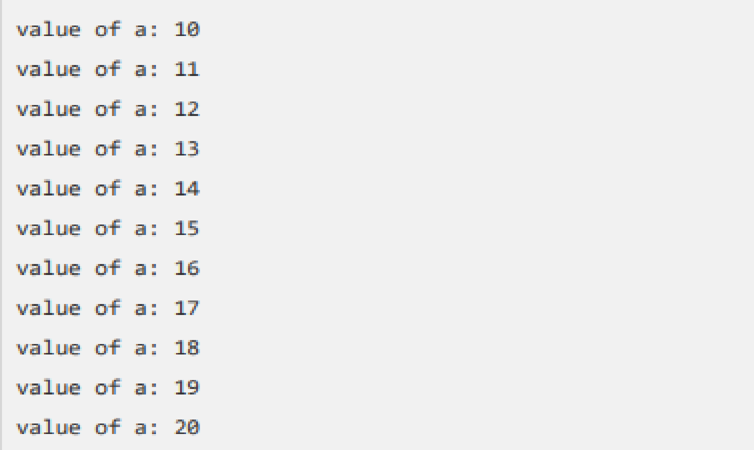
The syntax of a for loop in MATLAB is as following:

```
for variable = expression
statements
end
```

Example

```
for a = 10:20
   fprintf('value of a: %d\n', a);
 end
```

When the code above is executed, the result will be



```
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15
value of a: 16
value of a: 17
value of a: 18
value of a: 19
value of a: 20
```

The ''while...end'' loop

This loop is used when the number of passes is not specified. The looping continues until a stated condition is satisfied. The while loop has the form:

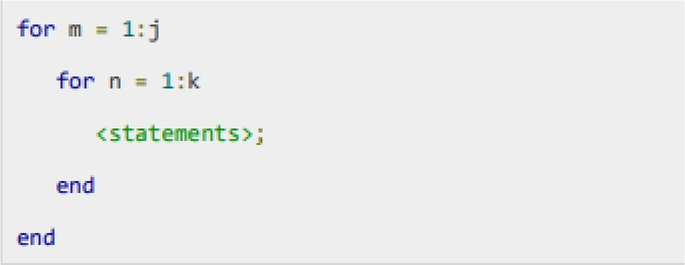
```
 while expression
 statements
 end
```

The statements are executed as long as expression is true.

```
 x = 1
while  x <= 10
 x = 3*x
 end
```

The Nested Loops

Matlab also allows to use one loop inside another loop. The syntax for a nested for loop statement in MATLAB is as follows:



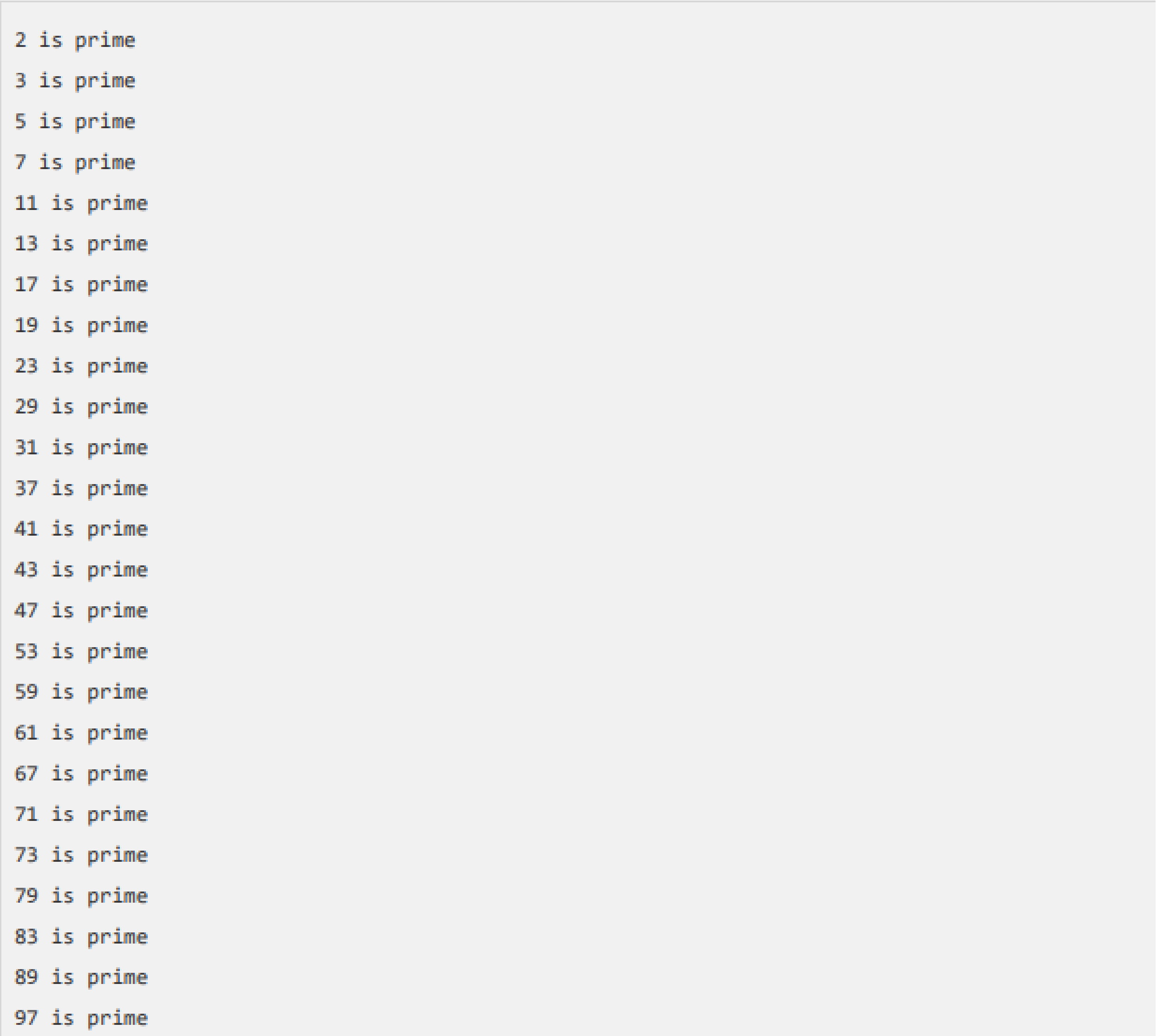
```
for m = 1:j
   for n = 1:k
      <statements>;
   end
end
```

It is important to note that if the condition inside the looping is not well defined, the looping will continue indefinitely. If this happens, we can stop the execution by pressing **Ctrl-C**.

Example

We can use a nested for loop to display all the prime numbers from 1 to 100.

```
for i=2:100
  for j=2:100
        if(~mod(i,j))
        break;
        % if factor found, not prime
        end
  end
        if(j > (i/j))
        fprintf('%d is prime\n', i);
        end
 end
```



```
2 is prime
3 is prime
5 is prime
7 is prime
11 is prime
13 is prime
17 is prime
19 is prime
23 is prime
29 is prime
31 is prime
37 is prime
41 is prime
43 is prime
47 is prime
53 is prime
59 is prime
61 is prime
67 is prime
71 is prime
73 is prime
79 is prime
83 is prime
89 is prime
97 is prime
```