Computer science

second Stage

Lec3

<u>Control flow</u>

Lect.Asst.Lubna Ali jalil

Control flow

In programming, control flow refers to the order in which individual instructions, statements, or function calls are executed or evaluated. By default, MATLAB executes code line by line from top to bottom. However, control flow structures allow us to:

• Make decisions based on conditions

• Repeat actions multiple times

• Skip or redirect parts of the code

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

1- Conditional Statements: if, else, elseif

MATLAB provides conditional statements to execute code only when specific conditions are satisfied.

The ''if...end'' structure MATLAB supports the variants of "if" construct.

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

➤ if ....end

Syntax

if condition

   statements

end

Example

    discr = 5;

    if discr < 0

      disp('Warning: discriminant is negative, roots are imaginary');

    end

➤ if ....else .... end

   Syntax

   if
   condition

   statements

   else

   statements

    end

   Example:

   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

```
if condition
   statements
elseif another_condition
   statements
else
   alternative_statements
end
```

Example:

```
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:

- if: Executes a block of code if the condition is true.
- else: Executes an alternative block if the condition is false.
- elseif: Allows checking multiple conditions in sequence.

- 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, These operators are used to compare values and form conditions.

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

note : Used when the number of iterations is known in advance.

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
```

Switch statement

A switch block conditionally executes one set of statements from several choices. Each choice is covered by a case statement.

The switch block tests each case until one of the cases is true. A case is true when

- For numbers, **eq(case_expression,switch_expression)**.
- For strings, **strcmp(case_expression,switch_expression)**.
- For objects that support the **eq(case_expression,switch_expression)**.
- For a cell array case_expression, at least one of the elements of the cell array matches switch_expression, as defined above for numbers, strings and objects.

When a case is true, MATLAB executes the corresponding statements and then exits the switch block.

The **otherwise** block is optional and executes only when no case is true.

Syntax

The syntax of switch statement in MATLAB is –

```
switch <switch_expression>
    case <case_expression>
        <statements>
    case <case_expression>
        <statements>
        ...
        ...
    otherwise
        <statements>
end
```

Example

Create a script file and type the following code in it

```
grade = 'B';
 switch(grade)
 case 'A'
   fprintf('Excellent!\n' );
 case 'B'
   fprintf('Well done B\n' );
 case 'C'
   fprintf('Well done c\n' );
 case 'D'
   fprintf('You passed\n' );
 case 'F'
   fprintf('Better try again\n' );
 otherwise
   fprintf('Invalid grade\n' );
 end
```

Homework

1. What does control flow refer to in programming?
2. Which of the following is NOT a MATLAB control flow structure?
3. What is the purpose of the if statement in MATLAB?
4. Which keyword is used to handle alternative conditions in MATLAB?
5. Which of the following is true about elseif in MATLAB?
A) It must be written as two words
B) It requires a semicolon
C) It is written as one word
D) It cannot be used with if
6. What does the relational operator == mean in MATLAB?
7. Which loop is used when the number of iterations is known in advance?
8. What is the correct syntax for a for loop in MATLAB?
9. Which loop continues until a condition becomes false?
10. What happens if a loop condition is not well-defined?
11. Which command stops an infinite loop in MATLAB?
12. What does a switch block do in MATLAB?
13. Which keyword is used when no case in a switch block is true?
14. In a switch block, when does MATLAB exit the block?

شكرا لحسن الأصغاء