

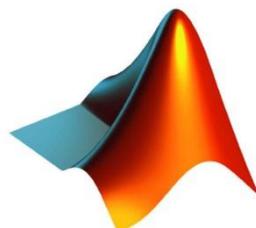
|  |                                  |
|--|----------------------------------|
| العلوم   | الكلية                           |
| الرياضيات  | القسم                            |
| Programming (MATLAB)   | المادة باللغة الانجليزية         |
| البرمجة بلغة (ماتلاب)  | المادة باللغة العربية            |
| الثانية  | المرحلة الدراسية                 |
| صفوت عبدالقادر حمد   | اسم التدريسي                     |
| Switch Statements  | عنوان المحاضرة باللغة الانجليزية |
| جملة الحالة ( Case )   | عنوان المحاضرة باللغة العربية    |
| الخامسة عشر  | رقم المحاضرة                     |
| MATLAB<br>A Practical Introduction to Programming and<br>Problem Solving | المصادر والمراجع                 |
| MATLAB<br>The Language of Technical Computing                            |                                  |
| MATLAB numerical computing   |                                  |



University Of Anbar  
College Of Science  
Math Department



# Programing Language (MATLAB)



MATLAB

Lecture 15

Switch Statements

By:

Safwat A. Hamad

For the 2<sup>nd</sup> stage Math Department

## 1. The Switch Statement

A *switch* statement can often be used in place of a nested *if-else* or an *if* statement with many *elseif* clauses. *Switch* statements are used when an expression is tested to see whether it is equal to one of several possible values.

The general form of the switch statement is:

```
switch switch_expression
    case caseexp1
        action1
    case caseexp2
        action2
    case caseexp3
        action3
% etc: there can be many of these
otherwise
    action n
end
```

The switch statement starts with the reserved word `switch` and ends with the reserved word `end`. The `switch_expression` is compared, in sequence, to the case expressions (`caseexp1`, `caseexp2`, etc.). If the value of the `switch_expression` matches `caseexp1`, for example, then `action1` is executed and the switch statement ends. If the value matches `caseexp3`, then `action3` is executed and, In general, if the value matches `caseexp $i$`  where  $i$  can be any integer from 1 to  $n$ , then `action  $i$`  is executed. If the value of the `switch_expression` does not match any of the case expressions, the action after the word `otherwise` is executed (the  $n$ th action, `action  $n$` ) if there is an `otherwise` (if not, no action is executed). It is not necessary to have an `otherwise` clause, although it is frequently useful. The `switch_expression` must be either a scalar or a character vector.

For the previous example, the switch statement can be used as follows:

`switchletgrade.m`

```

% First, error-check
if quiz < 0 || quiz > 10
grade = 'X';
else
% If here, it is valid so figure out the
% corresponding letter grade using a switch
switch quiz
    case 10
        grade = 'A';
    case 9
        grade = 'A';
    case 8
        grade = 'B';
    case 7
        grade = 'C';
    case 6
        grade = 'D';
    otherwise
        grade = 'F';
    end
end
end
end
  
```

As the same action of printing 'A' is desired for more than one grade, these can be combined as follows:

```

switch quiz
case {10,9}
    grade = 'A';
case 8
    grade = 'B';
% etc.
  
```

The curly braces around the case expressions 10 and 9 are necessary.

In this example, we error-checked first using an if-else statement. Then, if the grade was in the valid range, a switch statement was used to find the corresponding letter grade.

```

-----
-----
-----
-----

```

SAFWAT HAMAD