MA2252 Introduction to Computing

Lecture 7: Branching Statements

Sharad Kumar Keshari

School of Computing and Mathematical Sciences

University of Leicester

Learning outcomes

At the end of lecture, students will be able to understand and create

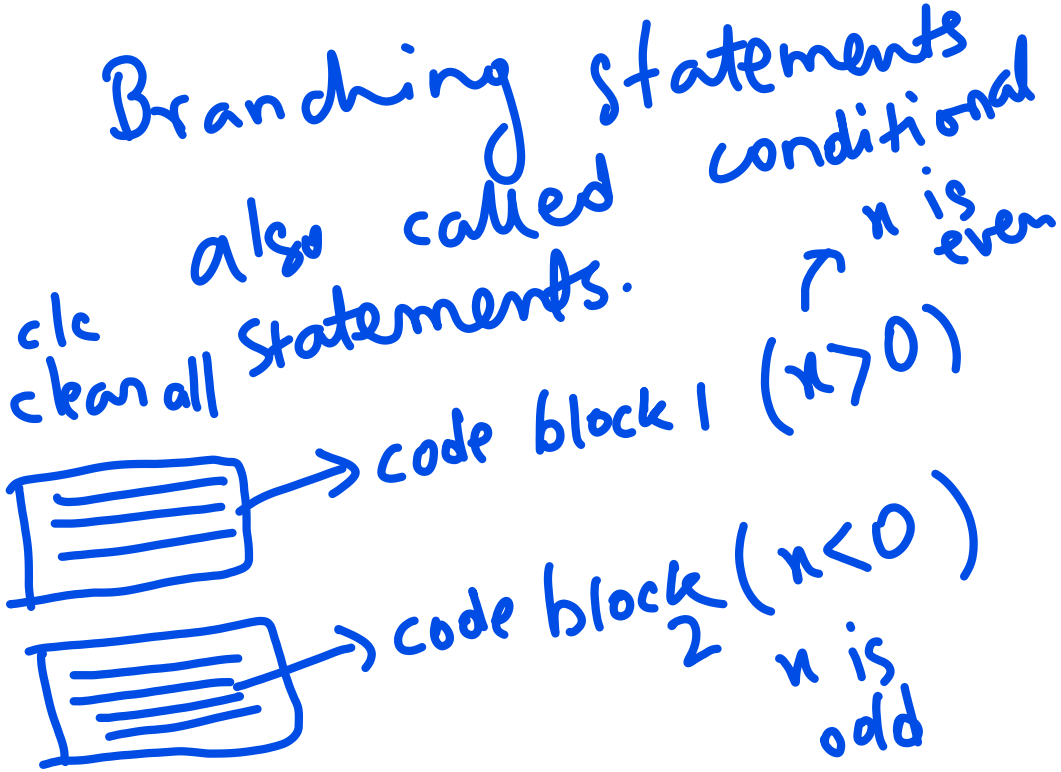
- If-Statements
- Switch statements

Branching Statements

Branching statements are used in programming to execute a section of code under specific conditions.

Examples:

1. If-Statements
2. Switch Statements



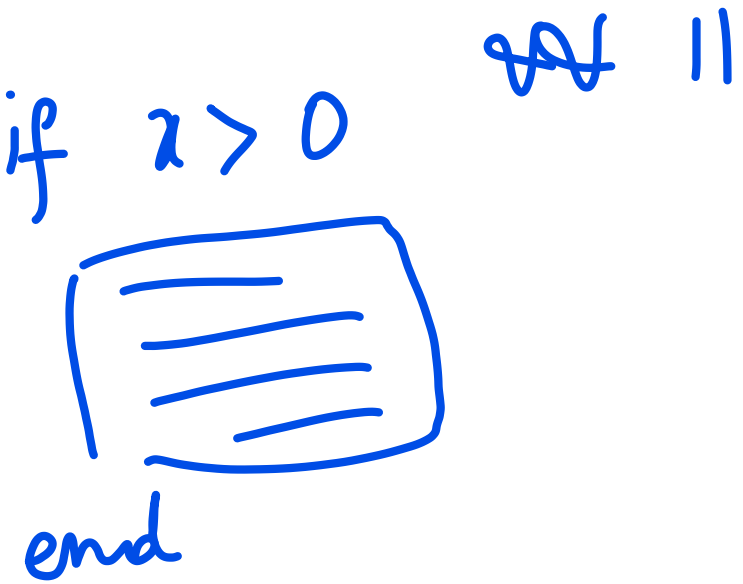
If-Statements

If statements can be constructed in different ways and they always end with end keyword.

1. Using if keyword

   Construction:

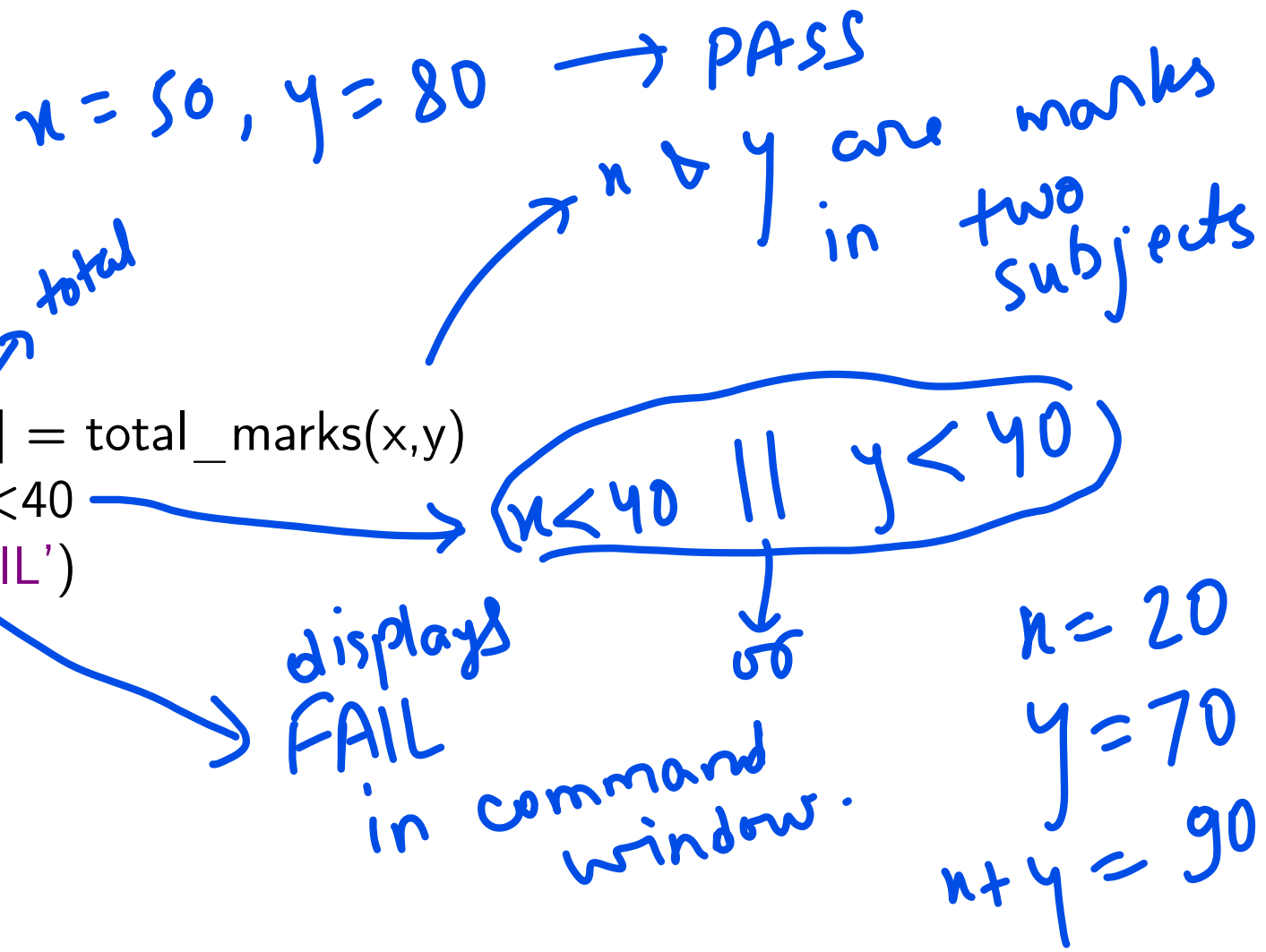
   if logical expression
   code block
   end



If-Statements (contd.)

Example:

```
function [t] = total_marks(x,y)
if x<40||y<40
   disp('FAIL')
end
t=x+y;
end
```

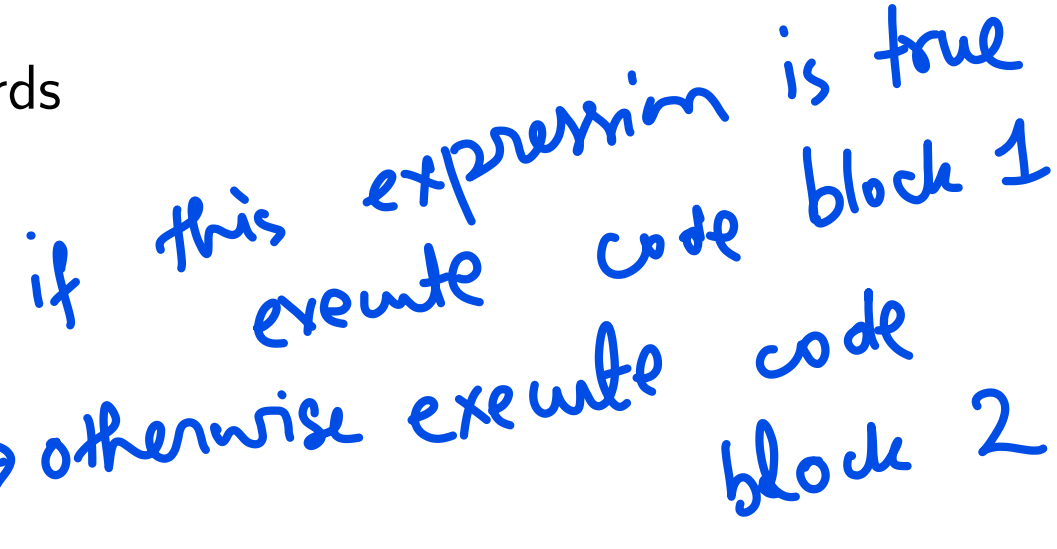


If-Statements (contd.)

2. Using if and else keywords

Construction:

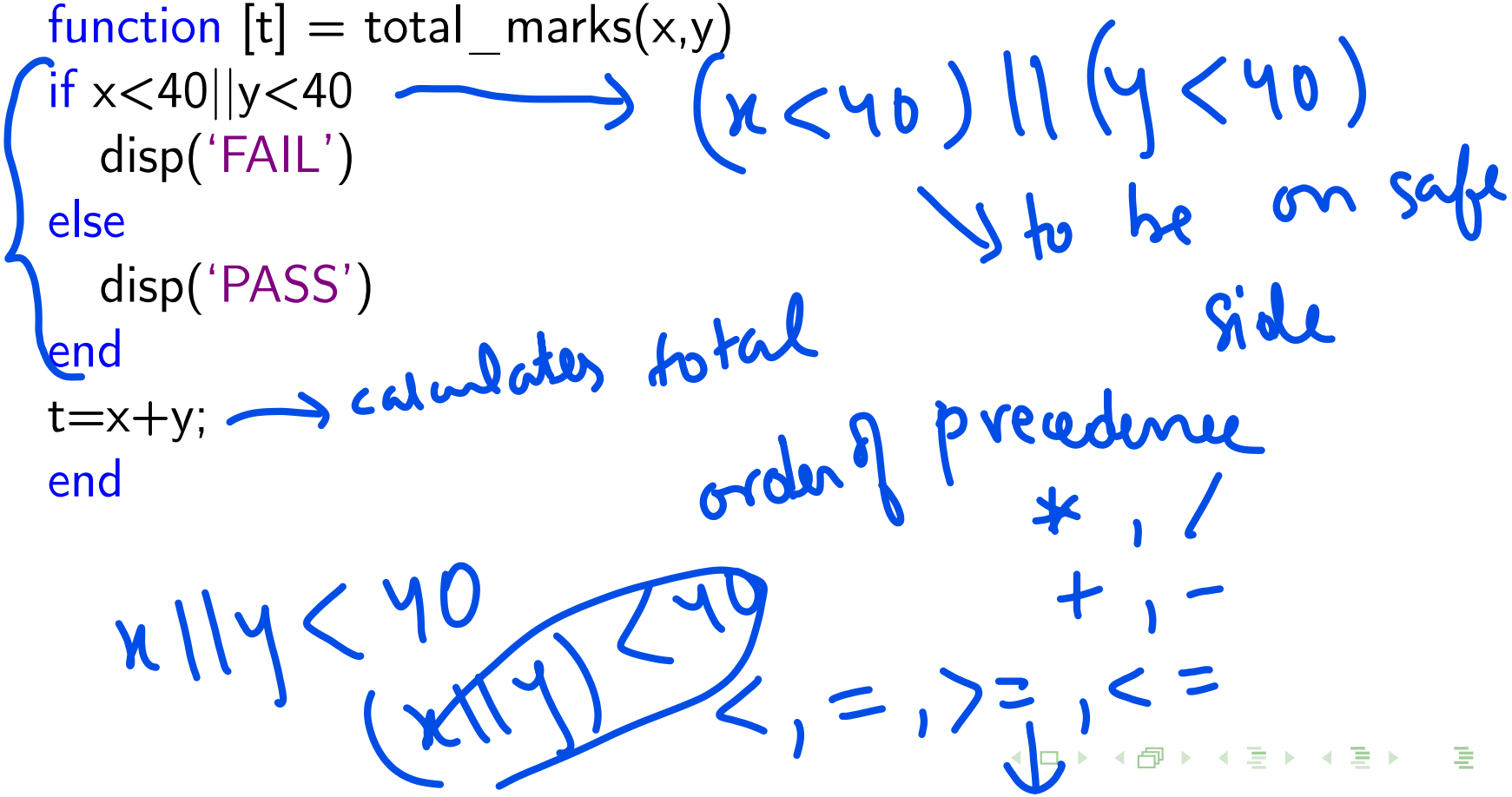
```
if logical expression
        code block 1
else
        code block 2
end
```



If-Statements (contd.)

Example:

```
function [t] = total_marks(x,y)
if x<40||y<40
    disp('FAIL')
else
    disp('PASS')
end
t=x+y;
end
```



If-Statements (contd.)

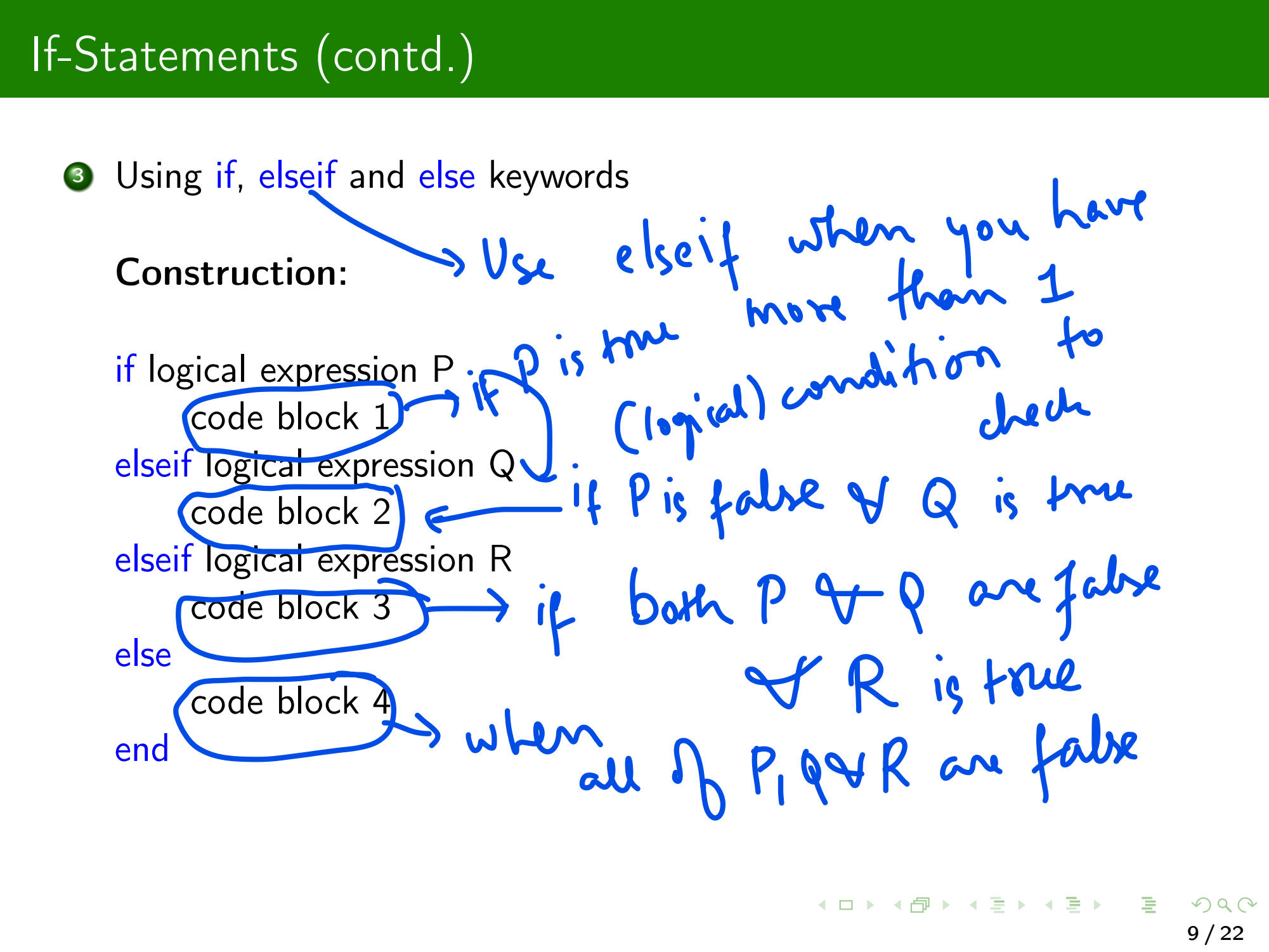
Demo

If-Statements (contd.)

3. Using if, elseif and else keywords

Construction:

```
if logical expression P
    code block 1
elseif logical expression Q
    code block 2
elseif logical expression R
    code block 3
else
    code block 4
end
```



If-Statements (contd.)

Example

```
function [t] = total_marks(x,y)
if x<0||y<0
   disp('marks cannot be negative')
elseif x>100||y>100
   disp('marks cannot exceed 100')
elseif x<40||y<40
   disp('FAIL')
else
   disp('PASS')
end
t=x+y;
end
```

P

Q

R

calculating the total

$x = 200$, $y = 30$ ✗ not legal marks

$x = -5$, $y = 10$ ✗ invalid marks

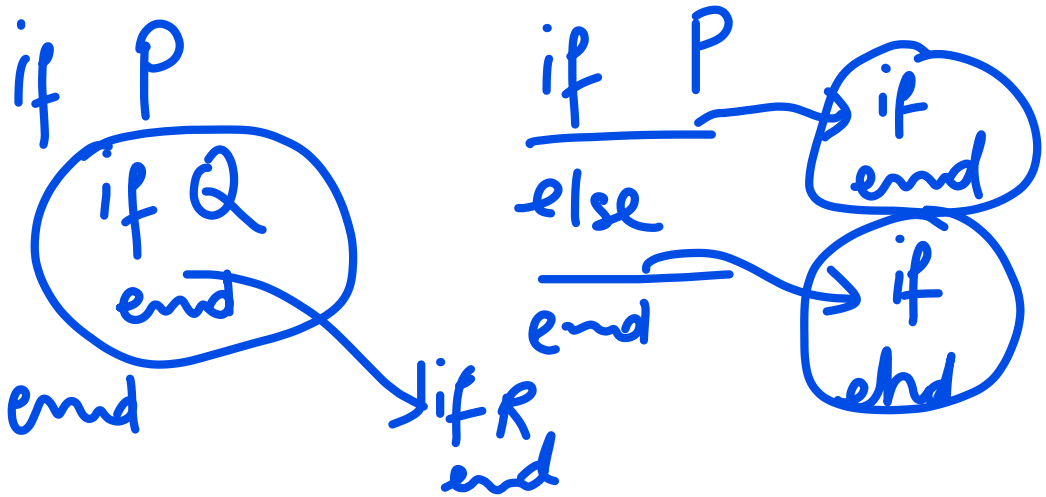
If-Statements (contd.)

Demo

If-Statements (contd.)

Nested If-Statement

If-statement nested/contained within another if-statement.



If-Statements (contd.)

Example:

```
function [t] = total_marks(x,y)
if x<0||y<0
   disp('marks cannot be negative')
   t=sprintf('cannot calculate total marks');
elseif x>100||y>100
   disp('marks cannot exceed 100')
   t=sprintf('cannot calculate total marks');
```

If-Statements (contd.)

```
else
t=x+y;
   if x<40||y<40
      disp('FAIL')
   else
      disp('PASS')
   end
end
end
```

total is calculated in the else part

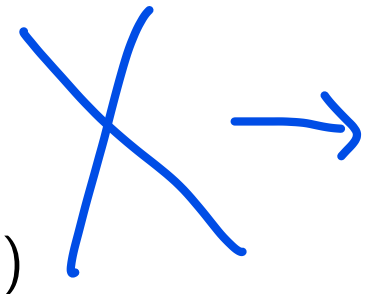
this is nested inside the parent if statement

If-Statements (contd.)

Demo

Activity

```
function [weight] = myweight(x)
if weight>70
   disp('Eat healthy')
elseif weight>80
   disp('Exercise more')
elseif weight>100
   disp('Call the doctor')
else
   disp('Relax!')
end
```



110 > 70 True

are not checked by MATLAB

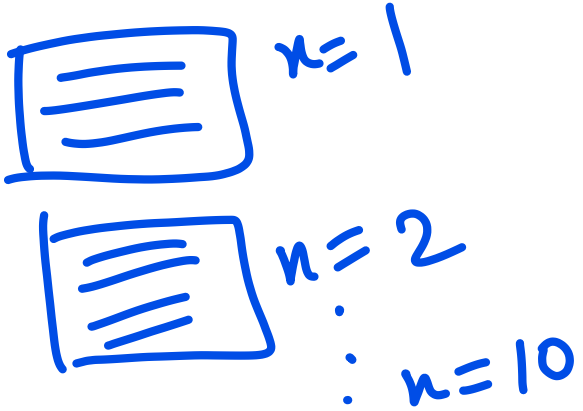
if weight > 100
call the doctor
else if weight > 80
else if weight > 70

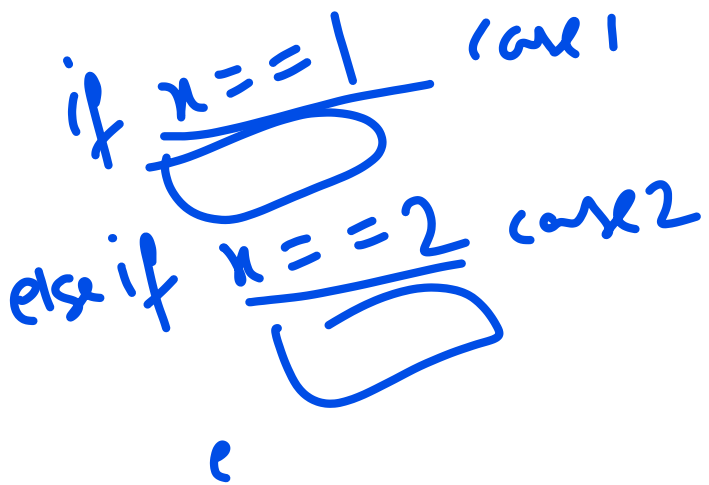
My weight is 110 kg. What suggestion this code will give? To answer, please go to mentimeter link in the chat.

Switch Statements

learn from the books S. Attaway & Otto-Denies

Switch statements are used to check if an expression is equal to one of possible values. Each value is called a case.

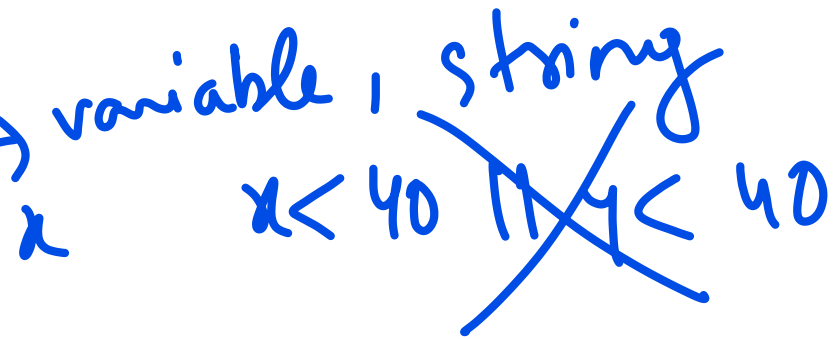




Switch Statements (contd.)

Construction:

```
switch expression
   case value 1
      code block 1
   case value 2
      code block 2
   case value 3
      code block 3
   otherwise
      code block 4
end
```



Switch x
case 1 n == 1
case 2

just like else in if

Switch Statements (contd.)

Example:

x is marks out of 10

```
function grade = mygrade_switch(x)
%this function calculates grade based on marks from 0 to 10 using switch
switch x
   case 10
      grade='A+';    %assign A+ if marks=10
   case 9
      grade='A';
   case{7,8}
      grade='B';
   otherwise
      grade='C'
end
end
```

7 or 8

6, 5, 4, 3, 2, 1, 0

else if else to state the validity of marks

Switch Statements (contd.)

Demo

If vs Switch

1. If-statements are useful when you deal with logical conditions in your code.
2. Switch becomes handy when you deal with one expression taking multiple possible values.
3. Switch statements can also be written using if-statements but vice-versa is not true in general.

End of Lecture 7

Please provide your feedback here