Introduction to Computer Programming
Lecture 11.2:

Review : Control Flow & Loops

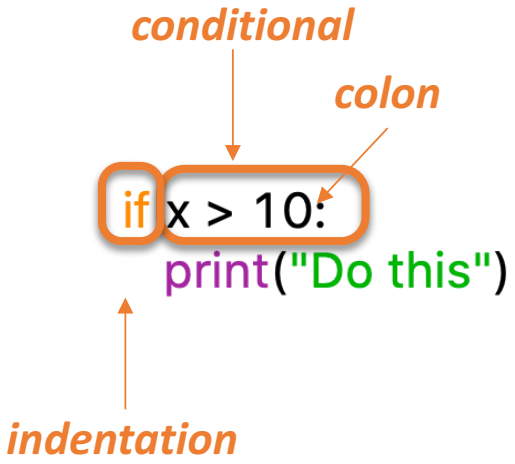
Hemma Philamore

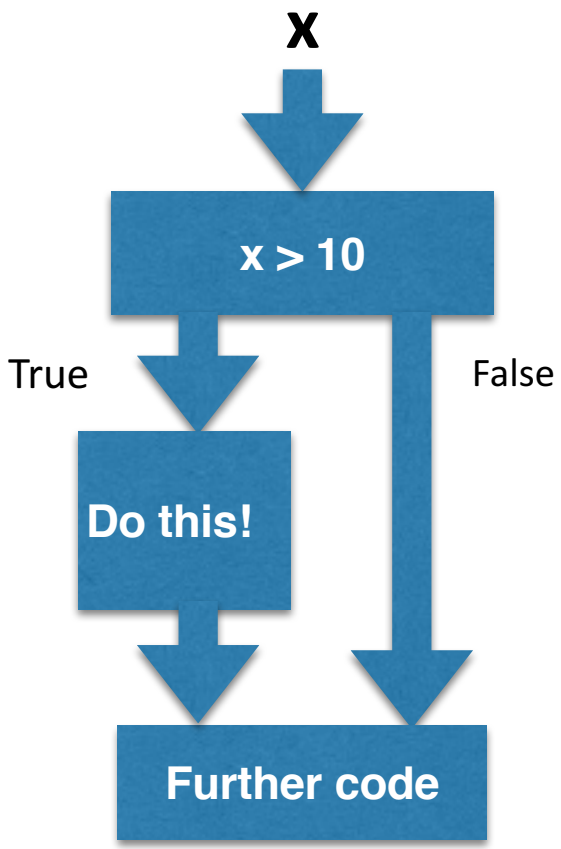
Department of Engineering Mathematics

Conditional Statements

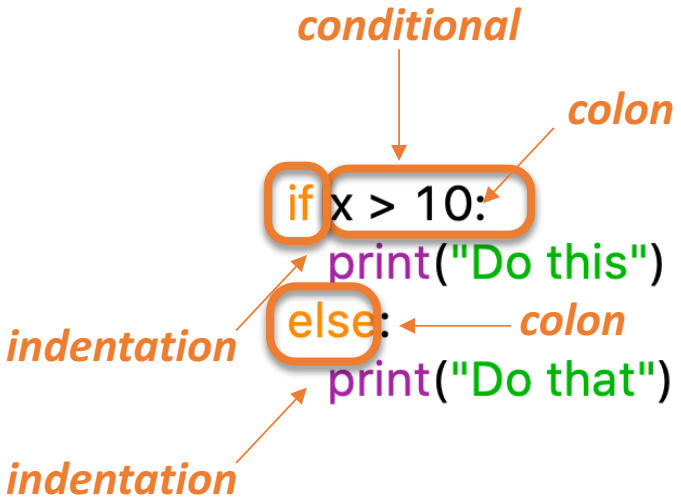
Making decisions in a program

If : Runs the block of code only if the condition is true.





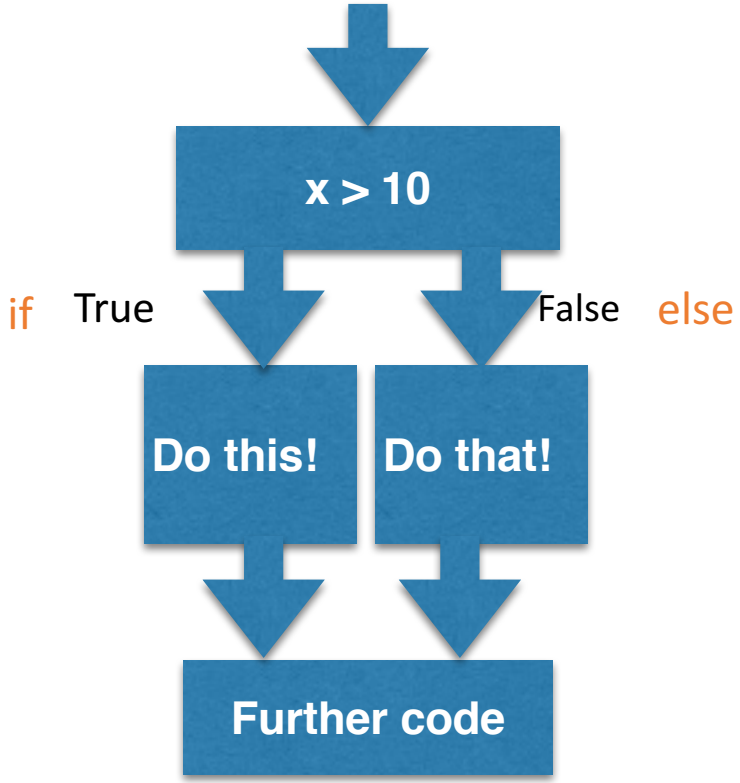
Conditional Statements



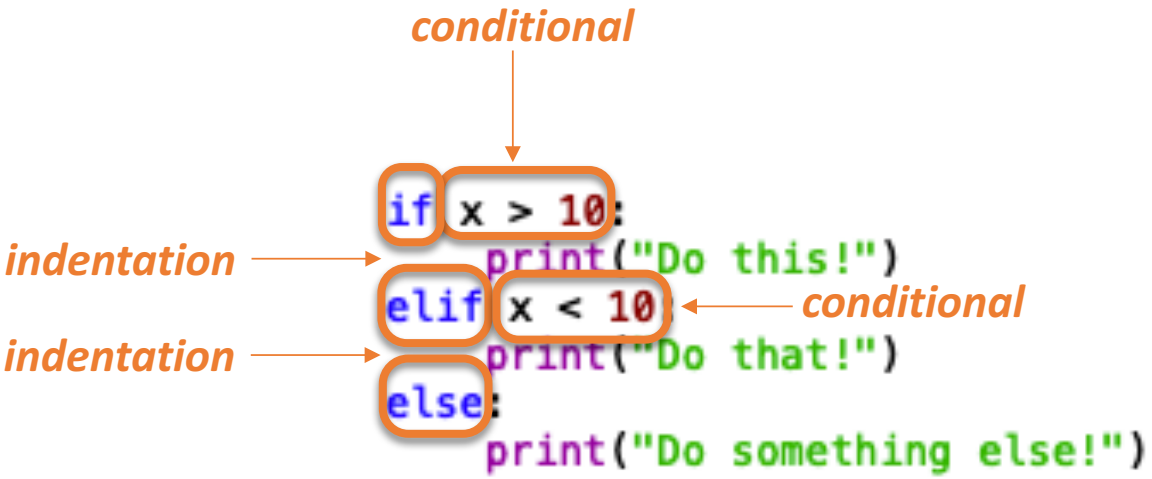
If... else...

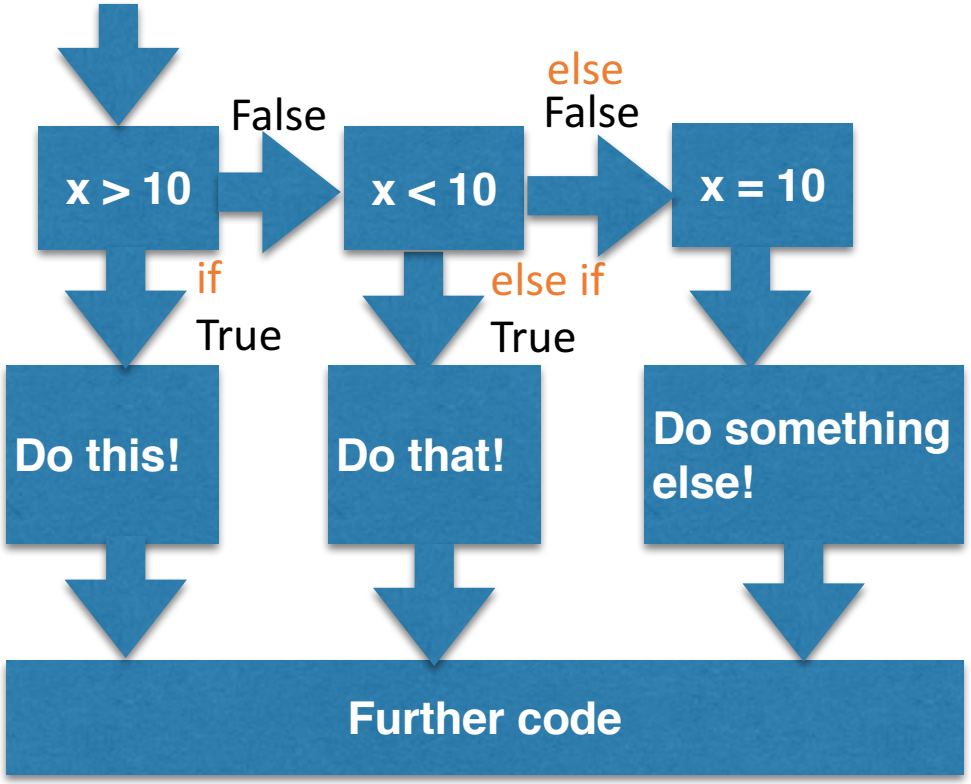
Runs the block of code under “if” only if the condition is true.

Runs the “else” code otherwise.



Conditional Statements





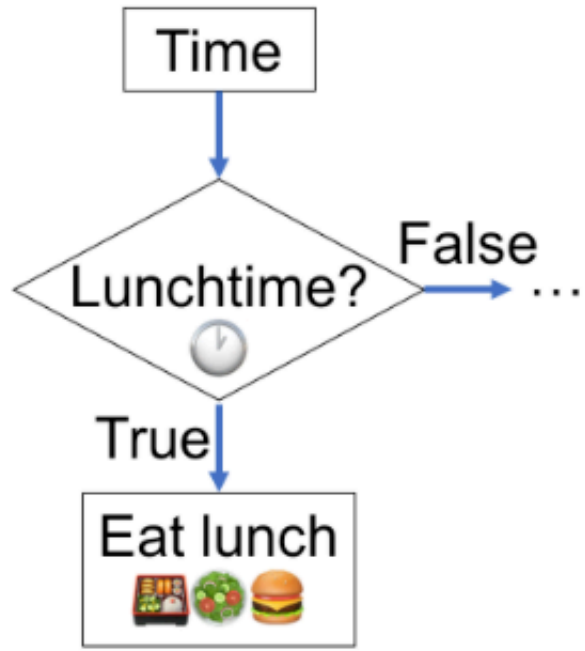
If... elif.. else...

Runs the block of code under “if” only if the condition is true.

Otherwise, runs the block of code under “elif” only if the condition is true.

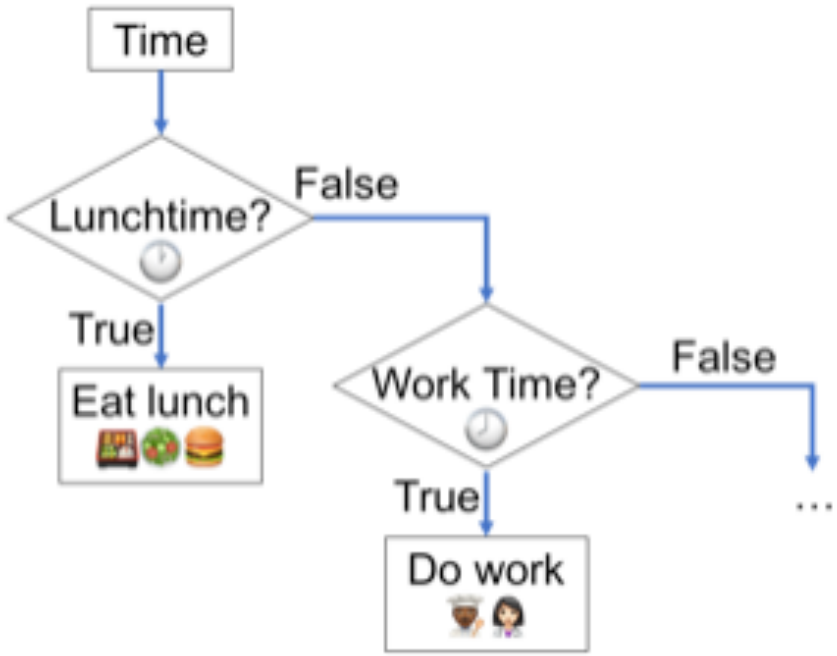
Otherwise, runs the “else” code otherwise.

```python
if lunch_time is True:
    print("eat lunch")
```



We can check if an alternative to the `if` statement is true using an `else if` statement.

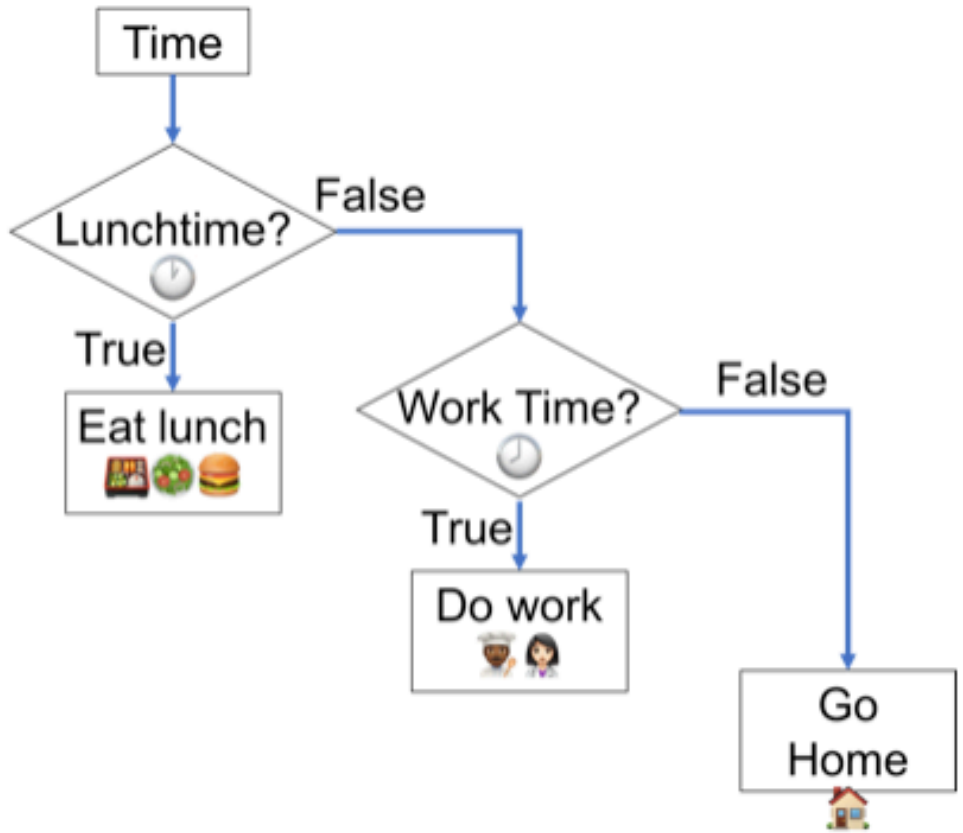
```
if lunch_time is True:
    print("eat lunch")
elif work_time is True:
    print("do work")
```



Often it is useful to include an `else` statement.

If none of the `if` and `else if` statements are satisfied, the code following the `else` statement will be executed.

```
if lunch_time is True:
    print("eat lunch")
elif work_time is True:
    print("do work")
else:
    print("go home")
```



Q11.2.A

A UK currency conversion company calculates the amount received by the customer as:

$$r = p * m * f$$

r = amount received by customer
p = amount paid by customer (GBP £)
m = market conversion rate
f = scale factor (depends on r)

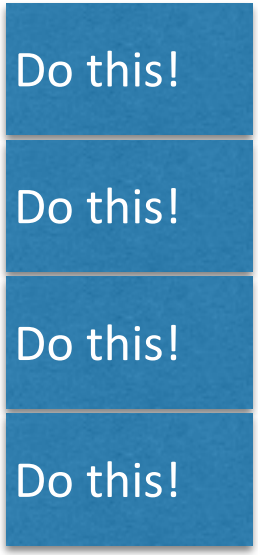
m depends on the currency being bought.
For US dollars $, m = 1.36

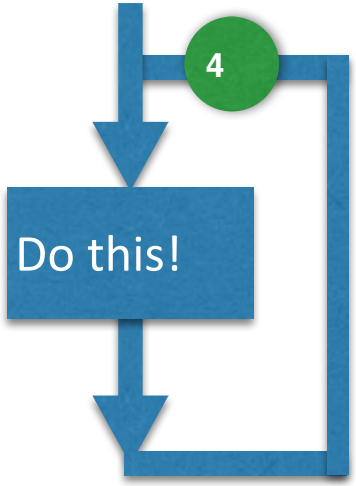
f depends on the amount paid, **p**:

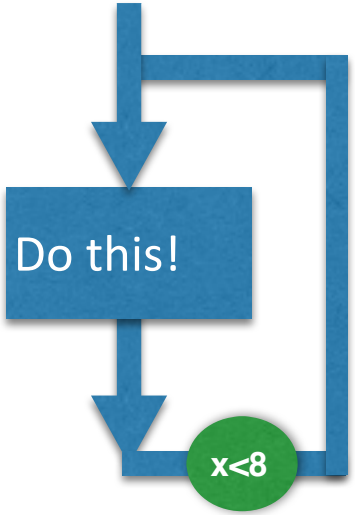
	f
$p \leq$ £100	0.9
£100 $< p \leq$ £1,000	0.925
£1000 $< p \leq$ £10,000	0.95
$p >$ £10,000	0.97

Write a program using conditional statements to calculate **r**, for an input value, **p**.

Loops







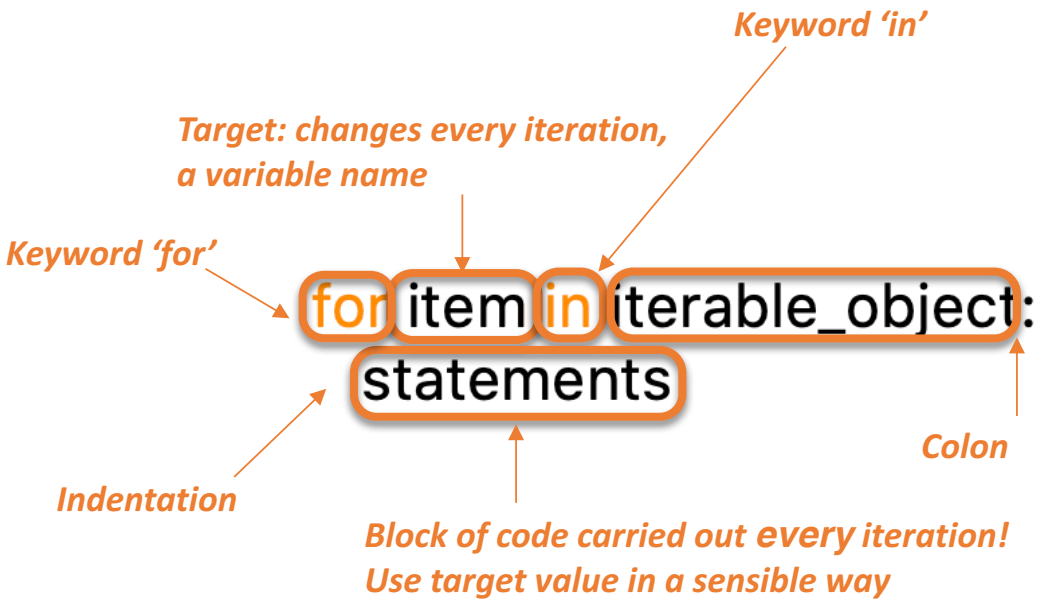
n iterations

condition

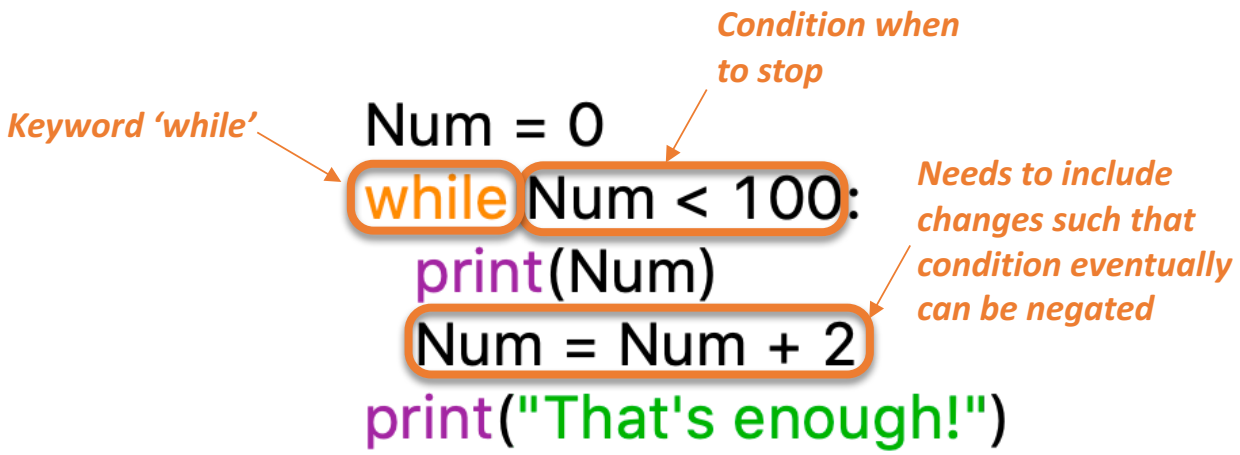
For Loop

While Loop

For Loop


Keyword 'in'
Target: changes every iteration,
a variable name
Keyword 'for'
for item in iterable_object:
statements
Colon
Indentation
Block of code carried out every iteration!
Use target value in a sensible way

While Loop


Condition when to stop
Keyword 'while'
Num = 0
while Num < 100:
print(Num)
Num = Num + 2
Needs to include changes such that condition eventually can be negated
print("That's enough!")

Q.11.2.B
Write a program that prints all the integers from 0 to 6 inclusive, using a loop.

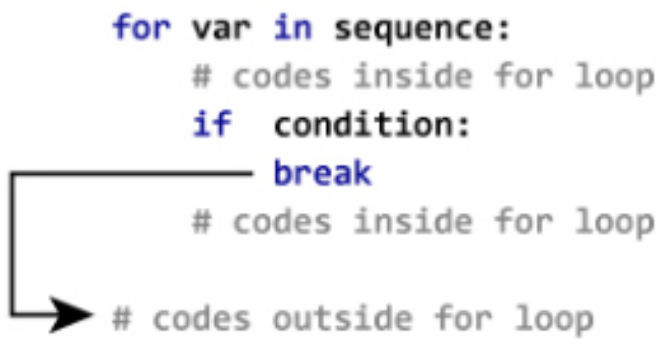
Q.11.2.C
Write a program that prints every second element of a list, using a loop.

Q11.2.D
Write a program that requests a number from the user until the number input by the user is greater than 10.

Break & Continue

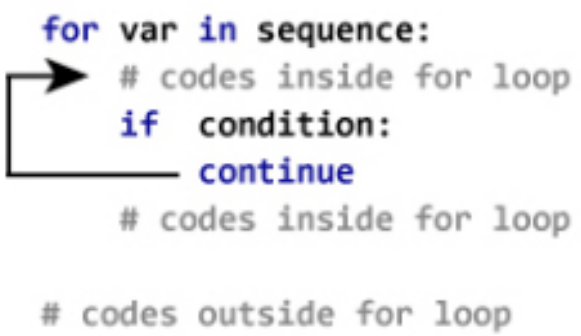
Break

Sometimes we want to exit a for or while loop prematurely i.e skip all remaining values



Continue

Sometimes, instead of *skipping all remaining values*, we want to skip *just one value* in a loop.



Q.11.2.E
Write a program that prints all the integers from 0 to 6 inclusive, **except 3 and 4**, using a loop.
(Modify your answer to Q.11.2.B)

Q.11.2.F
Write a program that prints every element of a list of numbers, using a loop, **exiting the loop before printing the number if a number greater than 10 is reached**.