Loops

# What are they?

A loop is a programming structure that enables repeating a collection of instructions multiple times.

# The 3 different types of loops:

1. The **while** loop: the test for exiting the loop is applied at the beginning

**In pseudocode**

WHILE *<condition>* DO

*<body of loop>*

ENDWHILE

**In the language C**

while (*<condition>*)

{

*<body of loop>*

}

**In English**

**a)** if *<condition>* is false, then exit loop and continue after it; else:

**b)** execute the *<body of loop>*

**c)** go back to **a)**

<*condition*> is true?

no

yes

<*body of loop*>

1. the **do ... while** loop: the test for exiting the loop is applied at the end

**In pseudocode**

REPEAT

*<body of loop>*

WHILE *<condition>*

**In the language C**

do

{

*<body of loop>*

}

while (*<condition>*);

**In English**

**a)** execute the <*body of loop*>

**b)** if <*condition*> is true, then go back to **a)**

<*body of loop*>

<*condition*> is true?

no

yes

1. the **for** loop:  
    the looping parameters are on one line, and are not modified from within the body of the loop

**In pseudocode**

FOR *<var>* = *<min>* TO *<max*> JUMP *<increment>*

<*body of loop*>

ENDFOR

**In the language C**

for(<*initialization*>;*<condition>*;*<increment>*)

{

*<body of loop>*

}

**In English**

**a)** *<initialization>*

**b)** if *<condition>* is false, then exit loop and continue after it; else:

**c)** execute the *<body of loop>*

**d)** execute the *<increment>* part

**e)** go back to **b)**

<*initialization*>

no

<*condition*> is true?

yes

<*increment*>

<*body of loop*>

# Which type to choose?

1. If you know the number of iterations the loop will need, then use a **for loop**.
2. Else: if you need to apply the test for exiting the loop at the beginning, then use a **while loop**.
3. Else: use a **do … while loop**.