

Repeating an action : loops

Interesting points:

- > What is a While- loop?
- > Why and when do we need it?
- > Examples

While- Loops... What are these?

A WHILE instruction defines a loop which is a control flow statement that allows code to be executed repeatedly as long as a given test is true. In other words, it is a facility offered to the programmer to flow control and to instruct the computer to repeat a task upon a given condition.

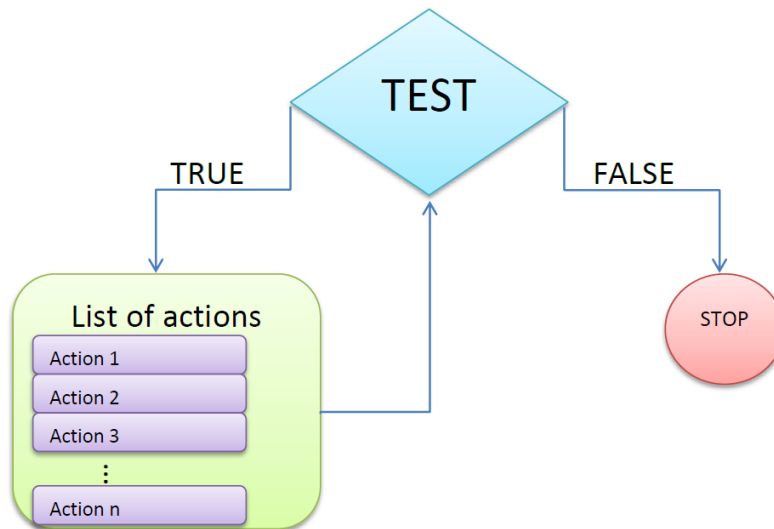
A While- Loop can be represented by this pseudo code:

```

WHILE
  (test)
  DO
    (list of actions)

```

Or by this schema:



Explaining the scheme: The while construct allows the repetitive execution of a list of actions, as long as the test controlling the while loop evaluates to true. If the test evaluates to false, the while loop will be terminated and usually another block of commands outside the body of the while- loop will be executed.

There is no limit to the amount of instructions put into a while. Interestingly, one or several IF/THEN/ELSE instructions can also be inserted instead of an action if there is a need to do some test (s) when inside the loop.

Repeating an action : loops

Why do we need While- loops?

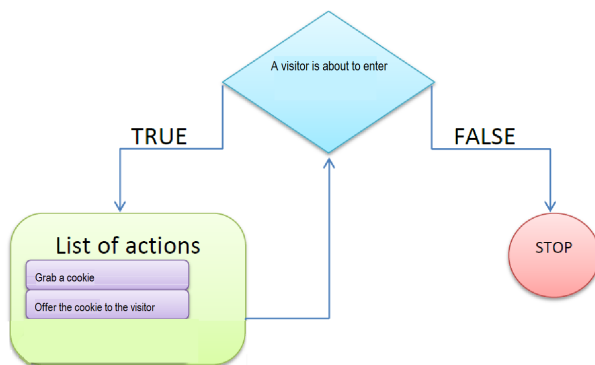
A while loop allows us to build elegant and efficient scripts, reducing significantly the number of commands of our script.

For example, consider the following scenario: 'a venue is on and hundreds of visitors arrived. You would like to offer handmade cookies to them while entering the area. You have plenty of cookies and a queue full of visitors In order to implement this script you will need a while loop.

The script in pseudo- code:

```
While (a visitor is about to enter) {
    Grab a cookie
    Offer the cookie to the visitor
}
```

The script in a scheme:



The same scenario without the While- loop would be problematic and would look like this:

```
IF (a visitor is about to enter) {
    Grab a cookie
    Offer the cookie to the visitor
}
IF (a visitor is about to enter) {
    Grab a cookie
    Offer the cookie to the visitor
}
IF (a visitor is about to enter) {
    Grab a cookie
    Offer the cookie to the visitor
}
IF (a visitor is about to enter) {
    Grab a cookie
    Offer the cookie to the visitor
}
[...]
```

We will need as many IFs as the number of the visitors that are in the queue.

We can easily realize that the While loop is a powerful structure that allows the programmer to flow control and to achieve repeatedly execution of actions.

Look carefully the following script. What is this script for?

```
bottles= 50
while (bottles ≠ 0) {
    grab a bottle
    place it on the shelf
    bottles= bottles -1
}
```

This script is used for placing 50 bottles on a shelf. While bottles are not zero (meaning: bottles exist), a bottle is grabbed and placed on the shelf and the total number of bottles is reduced by one.

So, the second time the number of the bottles will be 49, the condition will be true and a bottle will be grabbed and placed on the self. The bottles will be 48. Again the condition will evaluate to true; a bottle will be grabbed and placed on the shelf and the total number of bottles will be 47...

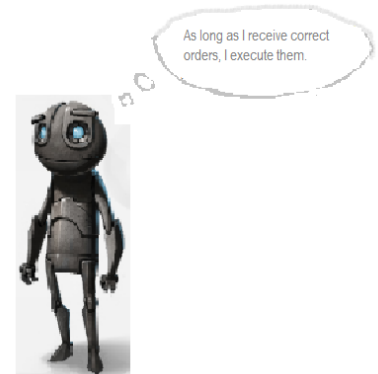
When the bottles will be zero nothing happens and the script terminates. Note: bottles ≠ 0 means that the number of bottles is not zero.

While loops in cMinds...

In the cMinds demonstrator, the WHILE code automatically adapts to its environment. When the WHILE is drag and dropped, automatically a field to put the test appears, guiding the user. Once the test is dragged in place, a DO image appears immediately with a field to put the

consequent action. Once the consequent action is dragged, another consequent action box appears (you can have up to 8 boxes) along with a cancel button if no further action is required.

In the example on the right we have inserted 2 boxes for actions. We could have inserted more or less in accordance with our needs.



Repeating an action : loops

Can you now give us an example from the tutorial area?

Tutorial area

Let's consider the case where the user wants to order the robot to pick an apple from the tree and put it in the basket below as long as the branch on top of him has some ripe ones. To give this instruction to the robot, the user will have to do the following

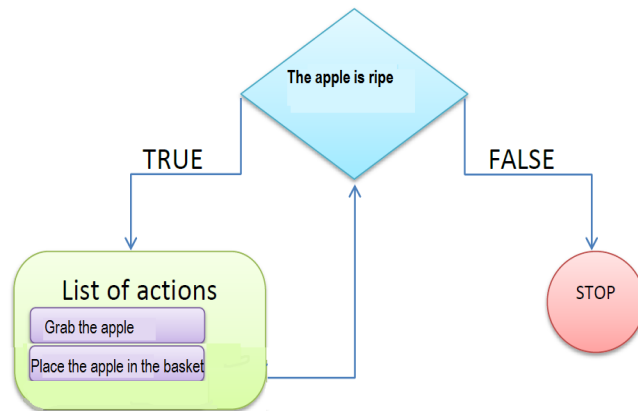
The script ▶▶



Which is the while condition that evaluates to true or false? ▼

The apple is ripe

Graphical representation ▼



Can you now give us an example from the Math activity?

Math Activity

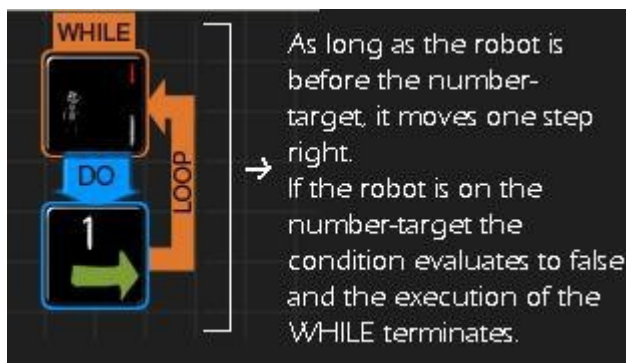
Consider the following scenario: The robot is on 5 and wants to reach 46. Help the robot move on 46 (number- target).

We can solve the problems in various ways. We will here demonstrate a solution that exploits While- loops.

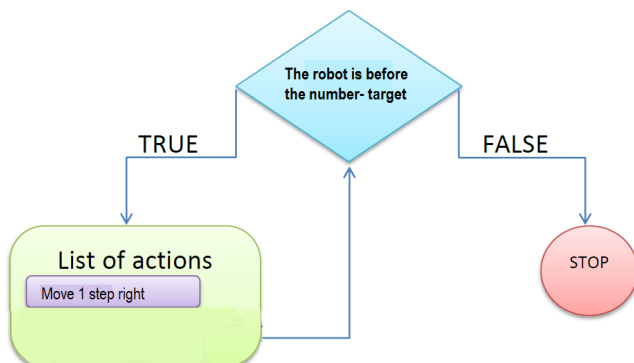
Which is the while condition that evaluates to true or false? ▼

The robot is before the number- target.

The script ▼



Graphical representation of the solution ▼



If we keep the same idea and decide not to use the While- loop, we will end up with a script that includes 41 'move one step right' actions. Here you can see the inefficient script...and you can realize once again the 'beauty' and 'dynamic functionality' of the While- loop.

