

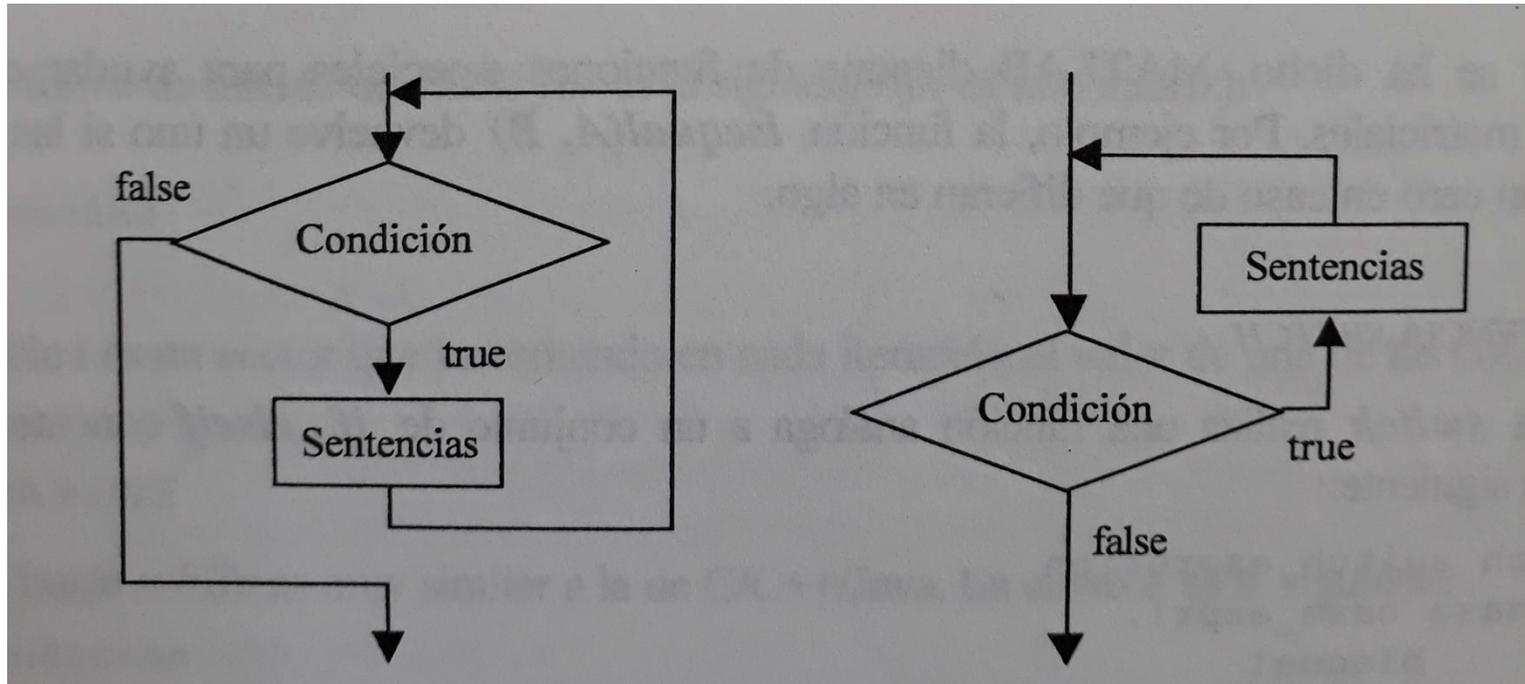
# SciLab

Óptica Computacional 2018

clase4

# Programación en Scilab

## Bifurcaciones y Bucles



Los **Bucles** permiten repetir una instrucción (u otra análoga) sobre diferentes datos siempre que se cumpla una dada condición (**control**)

Loop control based on a boolean expression **while**

```
while    (boolean expression)
```

```
        instructions
```

```
end
```

```
x = 16;  
while ( x > 1 )  
    x = x/2;  
end
```

```
for   variable = start: step: end

      instructions

end
```

## do

language keyword for loops

## Description

May be used inside `for` or `while` instructions to separate the loop variable definition and the set of instructions.

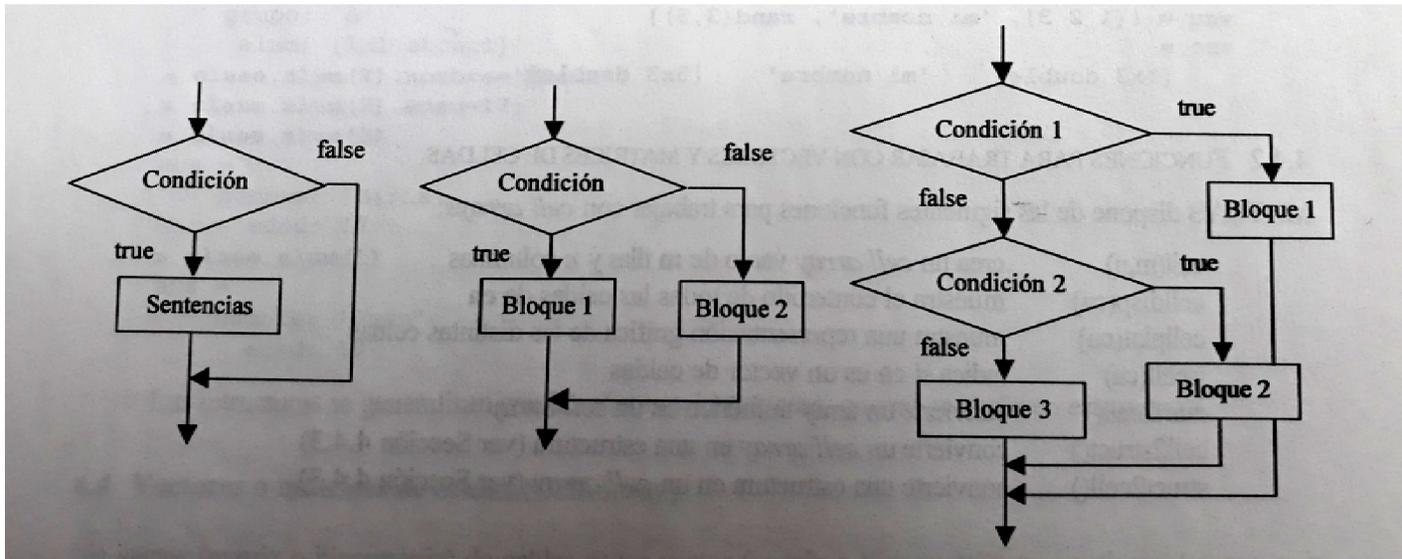
## Examples

```
i = 0
while i<5 do
    disp("i");
    i = i + 1;
end
```

```
n=5;
for i = 1:n do
    for j = 1:n do
        a(i,j) = 1/(i+j-1);
    end;
end
end
```

```
n = 10;
for k = 1:n
    y(k) = exp(k);
end
```

## Bifurcaciones y Bucles



Las **bifurcaciones** permiten una u otra operación según se **Cumpla** o **No se cumpla** una dada **condición**.

## Conditional statements `if`

```
if    boolean expression    then
    instructions 1
else
    instructions 2
end
```

```
if (x>=0) then
    disp("x is positive");
else
    disp("x is negative");
end
```

```
if rand(1,1) > 0.5 then
    disp("True");
else
    disp("False");
end
```

rather than

```
if rand(1,1) > 0.5 then disp("True"); else disp("False"); end
```

 The number of characters used to define the body of any conditional instruction (if while for or select/case) must be limited to 16k.

## Examples

```
i=2
for j = 1:3,
    if i == j then
        a(i,j) = 2;
    elseif abs(i-j) == 1 then
        a(i,j) = -1;
    else a(i,j) = 0;
    end,
end
```