ADA PROGRAMMING LANGUAGE TRAINING

Program 03
Check Positive

Pedro Fernando Flores Palmeros

1 INTRODUCTION

The main syntaxis of the `if` control structure is the next

```
if <condition> then
   statement 01;
   ...
   statement n;
else
   statement 01;
   ...
   statement m;
end if;
```

in this case `<condition>` is the condition to be tested it does not require the <> symbols.

2 CODE

```
with Ada.Text_IO; use Ada.Text_IO;
with Ada.Integer_Text_IO; use Ada.Integer_Text_IO;

procedure check_positive is
    N : Integer;
begin
    -- Put a String
    Put ("Enter an integer value: ");
```

```
9
10      -- Read in an integer value
11      Get(N);
12
13      -- Print an integer into the screen
14      Put(N);
15
16      if N > 0 then
17         -- Print a is a positive number
18         Put_Line(" is a positive number");
19      else
20         Put_Line(" is a non-positive number");
21      end if;
22  end check_positive;
```

3 Main parts

Observe that in this case in line 1 and 2 the packages that are going to be used are declared, and that is why in the lines 8, 15, 16, 17 the commands are without Ada.Text_IO. For further information in the first program is a detailed explanation of it.

3.1 Get(N)

This function is part of the package Ada.Text_IO, and it is useful for reading Integers from the keyboard, observe that a variable of Integer type has to be sent in the function argument.

3.2 Variable declaration

The variables and subsfunctions that are going to be used and implemented in the main procedure has to be defined in the lines between the procedure declaration and the begin of the implementation of the procedure. In this particular case between line 4 and 5.

3.3 if - end if directives

The main part of this document is the `if - then - end if` directives. In line 13 the `if` statement is used, compared with other programming languages the condition is not sorrounded by parenthesis but it could be if there are associative conditions. After the condition the reserved word `then` is used and after that the different statements of the body related to the conditional are added.

As in other programming languages when two paths are needed when a condition is tested, the case in which the condition is filled and the second scenario in which the condition is not filled the `else` directive is needed, and beteween the `else` and the `end if;` the statements to be executed must be placed.