6.2 Local and Global Indentifiers

Identifiers are names of variables, constants and procedures. A *global* identifier is an identifier not delcared inside a procedure (or function - in 6.4). A global identifier can be used from where it is declared until the end of the program. A *local* identifier is an identifier declared inside a procedure (or function). They can only be used inside the procedure and they only exist while the procedure is running.

In general, we should try to avoid using global variables inside procedures.

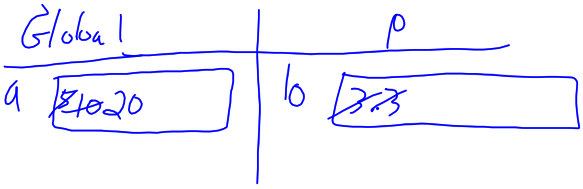
Here is an example program we will trace through. When keeping track of the variables you want to distinguish between the global ones and the ones in each procedure.

```
var a : int := 5
procedure p
    var b : int := 3
    put "a = ", a
    put "b = ", b
    a := a + a
end p
put "a: ", a
put "b: ", b
p
put "a: ", a
p
put "a: ", a
```

global

local

can't use a local variable outside procedure



Output

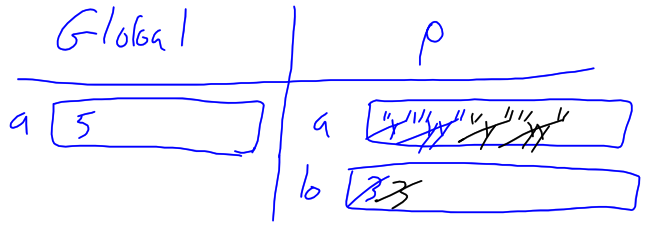
a: 5
a = 5
b = 3
a: 10
a = 10
b = 3
a: 20

It is possible to have local and global variables with the same name. If that is the case, when in the procedure you use the local variable. The global variable would not be visible in the procedure. Anywhere else in the program you would be able to use the global variable.

Let's add 1 line to the previous example and trace it again. We will add a local variable to the procedure `p` that has the same name as the global variable. The added line is in blue.

```
var a : int := 5
procedure p
    var a : string := "y"
    var b : int := 3
    put "a = ", a
    put "b = ", b
    a := a + a
end p
put "a: ", a
put "b: ", b
p
put "a: ", a
p
put "a: ", a
```

still illegal



Output

a: 5
a = y
b = 3
a: 5
a = y
b = 3
a: 5

Read section 6.2 in the textbook. Do exercise 6.2 #1-3 (I'll post solutions on Friday).

Remember the assignment was due last Monday if you haven't handed it in yet. We will do 6.3 this coming Tuesday.