

CADP 2023

Práctica 2 (parte 1) – Funciones y procedimientos

Aclaración importante: si bien en esta práctica se presentan soluciones utilizando variables globales, el objetivo de las mismas es comprender su funcionamiento y mostrar algunos de los problemas asociados con su uso. A la hora de resolver ejercicios en las siguientes prácticas, NO se deberán utilizar variables globales.

1. Dado el siguiente programa, indicar qué imprime.

```
program alcance1;
var a,b: integer;
procedure uno;
var b: integer;
begin
  b := 3;           3
  writeln(b);       2
end;               1
begin
  a:= 1;
  b:= 2;
  uno;
  writeln(b, a);
end.
```

2. Dado el siguiente programa, indicar qué imprime.

```
program alcance2;
var a,b: integer;
procedure uno;
begin
  b := 3;
  writeln(b);
end;
begin
  a:= 1;           3
  b:= 2;           3
  uno;             1
  writeln(b, a);
end.
```

3. Dado el siguiente programa, indicar cuál es el error y su causa.

```
program alcance3;
var a: integer;
procedure uno;
var b: integer;
```

```

begin
  b:= 2;
  writeln(b);
end;
begin
  a:= 1;
  uno;
  writeln(a, b);
end.

```

2
1
ERROR

4. Dado los siguientes programas, explicar la diferencia.

```

program alcance4a;
var a,b: integer;
procedure uno;
begin
  a := 1;
  writeln(a);
end;
begin
  a:= 1;
  b:= 2;
  uno;
  writeln(b, a);
end.

```

```

program alcance4b;
procedure uno;
begin
  a := 1;
  writeln(a);
end;
var a,b: integer;
begin
  a:= 1;
  b:= 2;
  uno;
  writeln(b, a);
end.

```

La diferencia ocurre en la variable global. Si la variable es global, aunque sea en el procedimiento, se guarda en todo el programa. En 'alcance4b' no ocurre eso.

5. Dado el siguiente programa, indicar cuál es el error.

```

program alcance4;
function cuatro: integer;
begin
  cuatro:= 4;
end;
var a: integer;
begin
  cuatro;
  writeln(a);
end.

```

variable 'a' no está inicializada en el programa, solamente en la función.

6. a. Realice un módulo que lea de teclado números enteros hasta que llegue un valor negativo. Al finalizar la lectura el módulo debe imprimir en pantalla cuál fue el número par más alto.
- b. Implemente un programa que invoque al módulo del inciso a.

7. Dado el siguiente programa:

```
program alcanceYFunciones;
var
  suma, cant : integer;
function calcularPromedio : real
var
  prom : real;
begin
  if (cant = 0) then
    prom := -1
  else
    prom := suma / cant;
  end;
begin { programa principal }
  readln(suma);
  readln(cant);
  if (calcularPromedio <> -1) then begin
    cant := 0;
    writeln('El promedio es: ' , calcularPromedio)
  end;
else
  writeln('Dividir por cero no parece ser una buena idea');
end.
```

- a) La función calcularPromedio calcula y retorna el promedio entre las variables globales suma y cant, pero parece incompleta. ¿qué debería agregarle para que funcione correctamente?
 - b) En el programa principal, la función calcularPromedio es invocada dos veces, pero esto podría mejorarse. ¿cómo debería modificarse el programa principal para invocar a dicha función una única vez?
 - c) Si se leen por teclado los valores 48 (variable suma) y 6 (variable cant), ¿qué resultado imprime el programa? Considere las tres posibilidades:
 - i) El programa original
 - ii) El programa luego de realizar la modificación del inciso a)
 - iii) El programa luego de realizar las modificaciones de los incisos a) y b)
- i) error porque function no retorna valor.
ii) devuelve 'El promedio es: 8'
ii) Lo mismo que el B, solo que el resultado de tipo entero se almacena en una variable

8. Dado el siguiente programa:

```

program anidamientos;
  procedure leer;
  var
    letra : char;
  function analizarLetra : boolean
  begin
    if (letra >= 'a') and (letra <= 'z') then
      analizarLetra := true;
    else
      if (letra >= 'A') and (letra <= 'Z') then
        analizarLetra := false;
      end; { fin de la funcion analizarLetra }
    begin
      readln(letra);
      if (analizarLetra) then
        writeln('Se trata de una minúscula');
      else
        writeln('Se trata de una mayúscula');
      end; { fin del procedure Leer}
    end;
  var
    ok : boolean;
  begin { programa principal }
    leer;
    ok := analizarLetra;
    if ok then
      writeln('Gracias, vuelva prontoss');
    end.
  end.

```

a) i) (ok := analizar Letra). No puede ejecutarse en el programa principal debido a que es un submódulo y podrá ejecutarse solo en el procedure "leer".
ii) Declarando 'analizarLetra' fuera del procedure.

- a) La función **analizarLetra** fue declarada como un submódulo dentro del procedimiento leer. Pero esto puede traer problemas en el código del programa principal.
- ¿qué clase de problema encuentra?
 - ¿cómo se puede resolver el problema para que el programa compile y funcione correctamente?
- b) La función **analizarLetra** parece incompleta, ya que no cubre algunos valores posibles de la variable letra.
- ¿De qué valores se trata?
 - ¿Qué sucede en nuestro programa si se ingresa uno de estos valores?
 - ¿Cómo se puede resolver este problema?

- b) i) ¿números y caracteres? no sé.
ii) Saltaría un error a mitad del programa.
iii) Corrigiendo el if de la function. writeln en los if