

EJERCICIOS ADICIONALES

1. Dado el siguiente programa:

```
program Ejercicio1_ad;
  procedure intercambio(var num1,num2 : integer);
  var
    aux : integer;
  begin
    aux := num1;
    num1 := num2;
    num2 := aux;
  end;
  procedure sumar(num1 : integer; var num2 : integer);
  begin
    num2 := num1 + num2;
  end;
var
  i, num1, num2 : integer;
begin
  read(num1);
  read(num2);
  for i := 1 to 3 do begin
    intercambio(num1,num2);
    sumar(i,num1);
  end;
  writeln(num1);
end.
```

- a. ¿Qué imprime si se leen los valores num1=10 y num2=5 ?
- b. ¿Qué imprime si se leen los valores num1=5 y num2=10 ?

2. Realice un programa modularizado que lea 10 pares de números (X,Y) e informe, para cada par de números, la suma y el producto de todos los números entre X e Y.

Por ejemplo, dado el par (3,6), debe informar:

“La suma es 18” (obtenido de calcular 3+4+5+6)

“El producto es 360” (obtenido de calcular 3*4*5*6)

3. Realizar un programa modularizado que lea información de 200 productos de un supermercado. De cada producto se lee código y precio (cada código es un número entre 1 y 200). Informar en pantalla:
- Los códigos de los dos productos más baratos.
 - La cantidad de productos de más de 16 pesos con código par.

4.
 - a. Realizar un módulo que reciba como parámetro el radio de un círculo y retorne su diámetro y su perímetro.
 - b. Utilizando el módulo anterior, realizar un programa que analice información de planetas obtenidas del Telescopio Espacial Kepler. De cada planeta se lee su nombre, su radio (medido en kilómetros) y la distancia (medida en años luz) a la Tierra. La lectura finaliza al leer un planeta con radio 0, que no debe procesarse. Informar:
 - Nombre y distancia de los planetas que poseen un diámetro menor o igual que el de la Tierra (12.700 km) y mayor o igual que el de Marte (6.780 km).
 - Cantidad de planetas con un perímetro superior al del planeta Júpiter (439.264 km).
5. 5. En la “Práctica 1 - Ejercicios Adicionales” se resolvieron 3 problemas complejos sin utilizar módulos. Al carecer de herramientas para modularizar, esos programas resultaban difíciles de leer, de extender y de depurar.
 - a) Analice sus soluciones a dichos problemas, e identifique:
 - qué porciones de su código podrían modularizarse? en qué casos propondría una estructura de módulos anidada?
 - qué tipo de módulo (función o procedimiento) conviene utilizar en cada caso? existe algún caso en los que sólo un tipo de módulo es posible?
 - qué mecanismos de comunicación conviene utilizar entre los módulos propuestos?
 - b) Implemente nuevamente los 3 programas, teniendo en cuenta los módulos propuestos en el inciso anterior