

Guía de Trabajos Prácticos II – Programación Estructurada

NOTA: Todos los programas de la guía deben contar al menos con una función o procedimiento.

1. Dada una temperatura ingresada en la escala Fahrenheit (valor entero), convertirla a la escala Celsius, y mostrar ambas temperaturas.

El programa debe invocar a una función FtoC que realice la conversión, recibiendo el valor en Fahrenheit y devolviendo el valor en Celsius (redondeando a entero).

La conversión es: $^{\circ}\text{C} = (^{\circ}\text{F} - 32) / 1,8$.

```
PROGRAM grados;
  USES Crt;
VAR
  f, c : integer;

FUNCTION FtoC( fahrenheit : integer ) : integer;
BEGIN
  FtoC:= round( 5 / 9 * ( fahrenheit - 32.0 ) )
END;

PROCEDURE pedirValor ( VAR x : integer );
BEGIN
  CLRSCR;
  WRITE ( 'Ingresa la temperatura en Fahrenheit: ' );
  READLN ( x );
END;

PROCEDURE calcular ( VAR y : integer );
BEGIN
  CLRSCR;
  WRITELN( 'Conversion de temperaturas: ' );
  WRITELN( ' Fahrenheit: ', y : 5 );
  WRITELN( ' Celsius: ', FtoC ( y ) : 5 );
END;

BEGIN
  pedirValor ( f );
  calcular ( f );

  REPEAT Until Keypressed;
END.
```

2. Definir un arreglo de palabras a modo de diccionario. Permitir el ingreso de una frase e informar las palabras no encontradas en el diccionario.

3. Decir el numero de vocales que aparecen en una frase introducida por el usuario.
4. Escriban un programa que suministrado un día, mes y año calcule la fecha del siguiente día, retornando el día, mes y año de la misma. Tanto el formato de entrada como de salida será: dd mm aaaa.
 - Nota: Un año es bisiesto si es divisible por 4, pero no por 100. Una excepción son los años divisibles por 400, que son todos bisiestos.
5. El valor del número Pi se puede calcular sabiendo que la serie: $1 - 1/3 + 1/5 - 1/7 + 1/9 - 1/11 + 1/13 - 1/15 \dots$ tiende a $\pi/4$. Escriban un programa que calcule una aproximación al número Pi usando un número de términos suministrado por el usuario. Se debe validar un mínimo (20) y máximo (1000) de términos permitidos, de tal forma que cuando se ingrese un valor fuera del rango se muestre un mensaje de error.
6. Escriban un programa que dado un número nos diga si este es primo. Los números primos son enteros mayores que 1 sin divisores enteros positivos, exceptuando el 1 y ellos mismos.

Observaciones a aplicar obligatoriamente para mejorar la performance del programa:

- Todos los números primos son impares excepto el 2.
 - Sólo necesitamos comprobar la divisibilidad por números superiores a la raíz cuadrada del número.
7. Escriban un programa que tome como entrada un número natural de 10 cifras a lo sumo (Si no es un número natural de 10 cifras a lo sumo, debe mostrarse un mensaje de error). El programa debe mostrar como salida:
 - Un mensaje indicando si el número es capicúa o no.
 - Un mensaje indicando la mayor cifra y la cantidad de veces que aparece.
 - Un mensaje indicando la menor cifra y la cantidad de veces que aparece.
 - Un mensaje indicando la cifra que más veces aparece y la cantidad de apariciones.
 8. Escribir un programa que permita el ingreso de dos números binarios y retorne como resultado el número binario producto de la suma de los ingresados.
 9. Escribir un programa que, utilizando procedimientos con parámetros, lea desde el teclado las unidades y el precio que quiere comprar, y si son más de 5 las unidades introducidas le haga un descuento del 10%. El programa finaliza cuando se ingresa una cantidad igual a 0.
 10. Escribir un programa, que con funciones, verifique si un caracter introducido es un número o no.
 11. Escribir un programa en Pascal que reciba un numero del 1 al 12 desde el

teclado y muestre el número de días correspondiente al mes que corresponda con ese día (usar funciones).

12. Hacer un programa que obtenga la distancia entre dos puntos que se encuentran en el plano. El programa finalizará cuando la distancia sea 0.