

# TECNÓLOGO EN INFORMÁTICA

## PRINCIPIOS DE PROGRAMACIÓN

### Práctico 1 — Ejercicios ordenados por dificultad

#### Ejercicio 1

Escribir un programa que pregunte al usuario su nombre, y luego lo salude.

#### Ejercicio 2

Dados dos números, mostrar la suma, resta, división y multiplicación de ambos.

#### Ejercicio 3

Calcular el perímetro y área de un rectángulo dada su base y su altura.

#### Ejercicio 4

Dados los catetos de un triángulo rectángulo, calcular su hipotenusa.

#### Ejercicio 5

Escribir un programa que convierta un valor dado en grados Fahrenheit a grados Celsius. Recordar que la fórmula para la conversión es:  $C = (F - 32) * \frac{5}{9}$ .

#### Ejercicio 6

Calcular la media de tres números pedidos por teclado.

#### Ejercicio 7

Pide al usuario dos números y muestra la «distancia» entre ellos (el valor absoluto de su diferencia, de modo que el resultado sea siempre positivo).

#### Ejercicio 8

Dadas dos variables numéricas A y B, que el usuario debe teclear, se pide realizar un algoritmo que intercambie los valores de ambas variables y muestre cuánto valen al final las dos variables.

#### Ejercicio 9

Realizar un programa que reciba una cantidad de minutos y muestre por pantalla a cuántas horas y minutos corresponde. Por ejemplo: 1000 minutos son 16 horas y 40 minutos.

#### Ejercicio 10

Un vendedor recibe un sueldo base más un 10% extra por comisión de sus ventas. El vendedor desea saber cuánto dinero obtendrá por concepto de comisiones por las tres ventas que realiza en el mes y el total que recibirá en el mes tomando en cuenta su sueldo base y comisiones.

#### Ejercicio 11

Una tienda ofrece un descuento del 15% sobre el total de la compra y un cliente desea saber cuánto deberá pagar finalmente por su compra.

### **Ejercicio 12**

Un alumno desea saber cuál será su calificación final en la materia de Algoritmos. Dicha calificación se compone de los siguientes porcentajes: 55% del promedio de sus tres calificaciones parciales, 30% de la calificación del examen final, 15% de la calificación de un trabajo final.

### **Ejercicio 13**

Pide al usuario dos pares de números  $(x1, y1)$  y  $(x2, y2)$ , que representen dos puntos en el plano. Calcula y muestra la distancia entre ellos.

### **Ejercicio 14**

Realizar un algoritmo que lea un número y que muestre su raíz cuadrada y su raíz cúbica. PSeInt no tiene ninguna función predefinida que permita calcular la raíz cúbica. ¿Cómo se puede calcular?

### **Ejercicio 15**

Dado un número de dos cifras, diseñar un algoritmo que permita obtener el número invertido. Ejemplo: si se introduce 23 que muestre 32.

### **Ejercicio 16**

Dos vehículos viajan a diferentes velocidades ( $v1$  y  $v2$ ) y están distanciados por una distancia  $d$ . El que está detrás viaja a una velocidad mayor. Se pide hacer un algoritmo para ingresar la distancia entre los dos vehículos (km) y sus respectivas velocidades (km/h) y con esto determinar y mostrar en qué tiempo (minutos) alcanzará el vehículo más rápido al otro.

### **Ejercicio 17**

Un ciclista parte de una ciudad A a las HH horas, MM minutos y SS segundos. El tiempo de viaje hasta llegar a otra ciudad B es de T segundos. Escribir un algoritmo que determine la hora de llegada a la ciudad B.

### **Ejercicio 18**

Pedir el nombre y los dos apellidos de una persona y mostrar las iniciales.

### **Ejercicio 19**

Escribir un algoritmo para calcular la nota final de un estudiante, considerando que: por cada respuesta correcta 5 puntos, por una incorrecta -1 y por respuestas en blanco 0. Imprimir el resultado obtenido por el estudiante.

### **Ejercicio 20**

Diseñar un algoritmo que nos diga el dinero que tenemos (en euros y céntimos) después de pedirnos cuántas monedas tenemos (de 2€, 1€, 50 céntimos, 20 céntimos o 10 céntimos).

### **Ejercicio 21**

Crea un diagrama de flujo, pseudocódigo y código de un programa que calcule el factorial de un número dado. El programa debe tomar el número como entrada y mostrar el factorial como salida.

Ejemplos: 3 → Factorial de 3 es:  $3 \times 2 \times 1 = 6$ ; 5 → Factorial de 5 es:  $5 \times 4 \times 3 \times 2 \times 1 = 120$ ; 6 → Factorial de 6 es:  $6 \times 5 \times 4 \times 3 \times 2 \times 1 = 720$ .

### **Ejercicio 22**

Crea un diagrama de flujo, pseudocódigo y código de un programa que cuente y muestre la cantidad de dígitos de un número dado. El diagrama debe tomar el número como entrada y mostrar la cantidad de dígitos como salida.

### **Ejercicio 23**

Crea un diagrama de flujo, pseudocódigo y código de un programa que calcule la suma de los dígitos de un número entero ingresado por el usuario.

### **Ejercicio 24**

Crea un diagrama de flujo, pseudocódigo y código de un programa que verifique si un número dado es un número perfecto o no. Un número es perfecto si la suma de todos sus divisores (a excepción de él mismo) suman la misma cantidad que el número.

Ejemplos: 6 → Es un número perfecto (ya que  $1 + 2 + 3 = 6$ ); 28 → Es un número perfecto (ya que  $1 + 2 + 4 + 7 + 14 = 28$ ); 9 → No es un número perfecto (ya que  $1 + 2 + 3 = 6$ ).

### **Ejercicio 25**

Crea un diagrama de flujo, pseudocódigo y código de un programa que calcule el máximo común divisor (MCD) de dos números dados. El diagrama debe tomar los dos números como entrada y mostrar el MCD como salida.

Ejemplos: 6 y 15 → El MCD (6, 15) es 3; 9 y 28 → El MCD (9, 28) es 1.

### **Ejercicio 26**

Crea un diagrama de flujo, pseudocódigo y código de un programa que calcule el mínimo común múltiplo (mcm) de dos números dados. El diagrama debe tomar los dos números como entrada y mostrar el mcm como salida.

Ejemplos: 6 y 15 → El mcm (6, 15) es 30; 9 y 28 → El mcm (9, 28) es 252.

### **Ejercicio 27**

Brinde pseudocódigo y código de un programa que reciba una contraseña (secuencia de caracteres) y reconozca si es segura o no. Se entiende que una contraseña es segura cuando tiene como mínimo 8 caracteres, incluyendo al menos una letra mayúscula, una letra minúscula y un número.

*Notas:* Puede asumir que el usuario ingresará el carácter ENTER para finalizar. Es necesario leer carácter a carácter con la función `scanf`.

### **Ejercicio 28**

Brinde pseudocódigo y código de un programa que cuente el número de vocales y consonantes en una frase dada. El programa debe solicitar al usuario que ingrese una frase y luego contar y mostrar la cantidad de vocales y consonantes en la frase.

*Notas:* Puede asumir que el usuario ingresará el carácter ENTER para finalizar. Es necesario leer carácter a carácter con la función `scanf`.

### **Ejercicio 29**

Crea un programa que cuente el número de palabras en una frase dada. El programa debe solicitar al usuario que ingrese una frase y luego contar y mostrar la cantidad de palabras en la frase.

*Notas:* Puede asumir que el usuario ingresará el carácter ENTER para finalizar. Es necesario leer carácter a carácter con la función `scanf`.

### **Ejercicio 30**

Implementa un programa que verifique si una cadena dada es una dirección de correo electrónico válida o no. Una dirección de correo electrónico válida debe contener un único signo "@" y al menos un punto después de "@".

*Notas:* Puede asumir que el usuario ingresará el carácter ENTER para finalizar. Es necesario leer carácter a carácter con la función `scanf`.

### **Ejercicio 31**

Crea un programa que convierta un número decimal dado a su equivalente en binario. El diagrama debe tomar el número decimal como entrada y mostrar el número binario como salida.

### **Ejercicio 32**

Crea un programa que convierta un número binario dado a su equivalente en decimal. El diagrama debe tomar el número binario como entrada y mostrar el número decimal como salida.

### **Ejercicio 33**

Implementa un juego en el que el programa genera un número aleatorio y el usuario intenta adivinarlo. Después de cada intento, el programa debe indicar si el número ingresado es mayor o menor que el número generado.

### **Ejercicio 34**

Crea un programa que genere una secuencia de 3 números aleatorios únicos en un rango de 1 a 5, sin repeticiones.