

- Ejercicio 1: Sistema de autenticación multiusuario

Escribe un programa que solicite el nombre de usuario y la contraseña. Dependiendo de las credenciales introducidas, el sistema debe verificar tres niveles de acceso:

- Nivel 1: Administrador (usuario: "admin", contraseña: "1234")
- Nivel 2: Usuario normal (usuario: "user", contraseña: "abcd")
- Nivel 3: Invitado (usuario: "guest", contraseña: "xyz")

Si las credenciales no coinciden, mostrar un mensaje de "Acceso denegado".

- Ejercicio 2: Cálculo de descuentos en una tienda

Un programa que solicite el precio de un producto y el tipo de cliente. Dependiendo del tipo de cliente, aplicar diferentes descuentos:

- Cliente regular: 5%
- Cliente premium: 10%
- Cliente VIP: 20%

Si el precio es mayor a 500 y el cliente es VIP, aplicar un descuento adicional del 5%.

- Ejercicio 3: Cálculo de notas y promedio

Desarrolla un programa que pida al usuario las notas de 5 materias. Luego, calcula el promedio y utiliza `if-else` para asignar una calificación:

- A (90-100)
- B (80-89)
- C (70-79)
- D (60-69)
- F (menos de 60)

Si alguna nota es negativa o mayor a 100, mostrar un error.

- Ejercicio 4: Validación de una fecha

Escribe un programa que solicite al usuario una fecha (día, mes, año) y valide si la fecha es correcta. Considera los años bisiestos y los meses que tienen

diferentes números de días. Usa `switch` para verificar los meses, y `if-else` para los años bisiestos.

- Ejercicio 5: Sistema de clasificación de triángulos

Solicita al usuario las longitudes de los tres lados de un triángulo. Usa una serie de `if-else` y operadores lógicos para determinar el tipo de triángulo:

- Equilátero: Los tres lados son iguales.
- Isósceles: Dos lados son iguales.
- Escaleno: Todos los lados son diferentes.
- Invalido: No forman un triángulo.

- Ejercicio 6: Calculadora de impuestos según ingresos

Escribe un programa que calcule los impuestos de una persona en función de sus ingresos anuales. Usa `if-else` para determinar los tramos:

- Menos de 20,000: 5%
- Entre 20,000 y 40,000: 10%
- Entre 40,000 y 80,000: 20%
- Más de 80,000: 30%

Además, si el usuario tiene más de 3 dependientes, reducir el impuesto en un 5%.

- Ejercicio 7: Juego de piedra, papel, tijera, lagarto, Spock

Crea un programa que permita a dos usuarios jugar a "Piedra, papel, tijera, lagarto, Spock". Usa `switch` para determinar el ganador según las siguientes reglas:

- Tijeras cortan papel.
- Papel cubre piedra.
- Piedra aplasta lagarto.
- Lagarto envenena Spock.
- Spock rompe tijeras.
- Tijeras decapitan lagarto.
- Lagarto come papel.

- Papel desautoriza a Spock.
- Spock vaporiza piedra.
- Piedra aplasta tijeras.

- Ejercicio 8: Algoritmo de números perfectos

Escribe un programa que determine si un número dado por el usuario es un número perfecto. Un número perfecto es aquel que es igual a la suma de sus divisores (excluyendo el propio número). Usa `if-else` y bucles para realizar la verificación.

- Ejercicio 9: Calculadora avanzada con operaciones complejas

Desarrolla una calculadora que permita realizar operaciones básicas (+, -, \*, /) y también potencias, raíces y factoriales. Usa `switch` para seleccionar la operación. Asegúrate de manejar los errores como divisiones entre cero, potencias con exponente negativo y raíces cuadradas de números negativos.

- Ejercicio 10: Sistema de gestión de calificaciones con promedio ponderado

Crea un programa que permita al usuario introducir las calificaciones de 6 materias, junto con el peso porcentual de cada una. Calcula el promedio ponderado usando `if-else` para verificar que la suma de los pesos sea 100%. Si no es así, muestra un mensaje de error. Además, si el promedio es menor a 60%, mostrar un mensaje indicando que el usuario ha reprobado.