

## Ejercitación Corte de Control

### 1. Plataformas de Streaming

Contexto: Una plataforma de streaming quiere un reporte de cuántas horas totales han pasado los usuarios viendo contenido.

- **Estructura:** struct Visualizacion { int idUsuario; string contenido; int minutosVistos; };
- **Precondición:** El vector debe estar ordenado por idUsuario.
- **Tarea:** Realizar un Corte de Control para mostrar por pantalla: "Usuario [ID]: totalizó [Suma de minutos] minutos de visualización". Al final del programa, mostrar el total general de minutos de todos los usuarios.

### 2. Torneo de eSports

Contexto: Se registran los puntos obtenidos por jugadores de distintos equipos en una jornada competitiva.

- **Estructura:** struct Partida { string nombreEquipo; string nicknameJugador; int puntosObtenidos; };
- **Precondición:** El vector debe estar ordenado por nombreEquipo.
- **Tarea:** Implementar un algoritmo que informe por cada equipo el puntaje total acumulado por todos sus integrantes. El reporte debe decir: "Equipo: [Nombre], Puntaje Total: [Suma de puntos]".

### 3. Festival de Rock Clásico

Contexto: Se recolectan los datos de las ventas de entradas para un festival donde tocan bandas legendarias en diferentes sectores (Platea, Campo, VIP).

- **Estructura:** struct VentaEntrada { string nombreBanda; string sector; float precioCobrado; };
- **Precondición:** El vector debe estar ordenado por nombreBanda.
- **Tarea:** Generar un listado que detalle la recaudación total por cada banda. Por cada cambio de banda, el programa debe imprimir: "Banda: [Nombre] - Recaudación total: \$[Suma de precios]".

### 4. Gastos de Viaje

Contexto: Un grupo de viajeros registra sus gastos diarios realizados en un viaje por Europa, categorizados por el tipo de gasto (Comida, Transporte, Alojamiento).

- **Estructura:** struct GastoViaje { string categoria; string detalle; float monto; };
- **Precondición:** El vector debe estar ordenado por categoria.
- **Tarea:** Utilizar Corte de Control para mostrar cuánto dinero se gastó en total por cada categoría. Además, informar cuál fue la categoría que representó el mayor gasto total para el grupo.

### 5. Inversiones en Criptoactivos / Bolsa

Contexto: Un inversor registra sus operaciones de compra de diferentes activos financieros durante el último semestre.

- **Estructura:** struct Operacion { string activoCripto; int cantidadComprada; float precioUnitario; };
- **Precondición:** El vector debe estar ordenado por simboloActivo.
- **Tarea:** Realizar un Corte de Control para determinar cuántas unidades totales posee el inversor de cada activo (por ejemplo, cuántos BTC, cuántos ETH, etc.). El programa debe mostrar: "Activo: [Símbolo] - Cantidad total acumulada: [Suma de cantidades]".