

Razonamiento para la Construcción del Código

Situación 2: Validación de Fechas

Sebastian Andres Manotas Blanco

Análisis del problema

El sistema aceptaba fechas inexistentes (p. ej., 30/02/1995, 32/01/2020, 29/02/2021). Problemas detectados:

- Falta de verificación de **rango del año** (se sugiere 1900–2025).
- Falta de verificación de **rango del mes** (1–12).
- Cómputo incorrecto de **días por mes** y del caso especial de **año bisiesto**.
- Mensajes de error poco informativos (el usuario no sabe cuántos días tiene el mes).

Diseño lógico (algoritmo)

1. Leer día, mes y año.
2. Validar año en [1900, 2025]. Si no, error y fin.
3. Validar mes en [1, 12]. Si no, error y fin.
4. Con `switch(mes)` determinar `diasDelMes`:
 - Meses 31 días: 1, 3, 5, 7, 8, 10, 12.
 - Meses 30 días: 4, 6, 9, 11.
 - Febrero (2):
 - Si bisiesto ($(anio \bmod 4 = 0 \wedge anio \bmod 100 \neq 0) \vee (anio \bmod 400 = 0)$) \rightarrow 29 días.
 - Si no \rightarrow 28 días.
5. Validar día en [1, `diasDelMes`]. Si no, error indicando cuántos días tiene el mes.
6. Si todo es válido, mostrar “Fecha válida” y, opcionalmente, si el mes es el actual, aplicar descuento por cumpleaños.

Diagrama de flujo

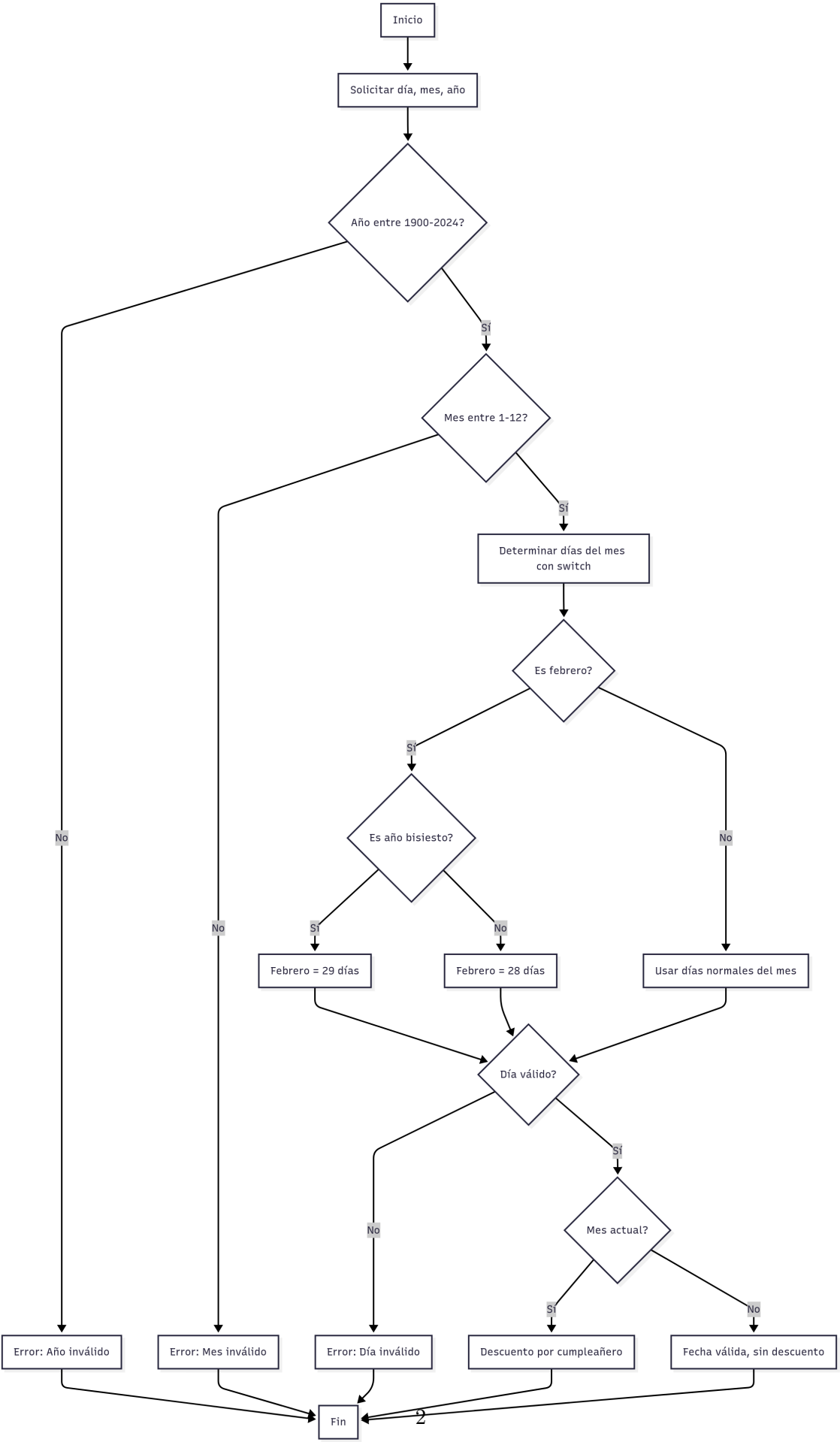


Figura 1: Diagrama de flujo. Situación 1