**Actividad de Aplicación: Operadores ,Condicionales y Diagramas de Flujo**

Diseñar el diagrama de flujo que represente la lógica completa y el código de Python teniendo en cuenta los temas vistos anteriormente (condicionales, operadores y diagramas de flujo)

**1. Determinar signo y paridad**

Dado un número entero proporcionado por el usuario, determina primero si es positivo, negativo o cero. Si el número no es cero, establece además si es par o impar.

**2. Una persona ingresa su edad. Clasifica esa edad en una de las siguientes categorías:**

* Menor de edad (<18).
* Adulto joven (18–30).
* Adulto maduro (31–65).
* Adulto mayor (>65).

**3. Tarifa de transporte según día y hora**

Un usuario indica si el día es laborable (“S”/“N”) y la hora del día (0–23). Calcula la tarifa aplicable:

**Día laborable**:

* PICO si la hora está entre 7–9 o 17–19 (inclusive).
* NORMAL en cualquier otra hora.

**Fin de semana** (día no laborable):

* FIN DE SEMANA.

**4. Descuento en tienda según monto y tipo de cliente**

* **Enunciado:**  
  Un cliente ingresa el monto de su compra y si es VIP (“S”/“N”). Aplica el siguiente descuento:
  + **VIP:**
    - 20 % si monto ≥ 500.
    - 10 % si monto < 500.
  + **No VIP:**
    - 5 % si monto ≥ 500.
    - 0 % si monto < 500.

**5. Validación de contraseña simple**

El usuario crea una contraseña que debe cumplir dos requisitos:

* 1. Tener al menos 8 caracteres.
  2. Contener al menos el carácter @.

Si la longitud es insuficiente, informar “Contraseña muy corta”. Si la longitud es correcta pero falta @, informar “La contraseña debe incluir al menos un '@'”. Sólo cuando ambos requisitos se cumplan, mostrar “Contraseña válida”.