

Una posible implementación en Python de una interpretación de una fórmula f de la lógica proposicional, $I:P_f \rightarrow \{1, 0\}$, es la siguiente:

$I = \{ 'P':0, 'Q':1, 'R':1, 'S':0 \}$ # VALORES VERDAD DE LETRAS PROPOSICIONALES

De esta manera, por ejemplo, el valor de verdad de p se obtiene mediante $I['P']$ cuyo valor es 0.

EJERCICIO 1: Implemente en Python la función V_I y compare con los resultados del ejercicio 1 del taller “Interpretaciones”.

Transcriba (ojo: No haga copy-paste) el siguiente código en Python3 y explique qué es lo que hace:

```
letrasProposicionales = ['p', 'q', 'r'] # lista con las letras proposicionales
interps = [] # lista todas las posibles interpretaciones (diccionarios)
aux = {} # primera interpretacion

for a in letrasProposicionales:
    aux[a] = 1 # inicializamos la primera interpretacion con todo verdadero

interps.append(aux) # ... y la incluimos en interps

for a in letrasProposicionales:
    interps_aux = [i for i in interps] # lista auxiliar de nuevas interpretaciones

    for i in interps_aux:
        aux1 = {} # diccionario auxiliar para crear nueva interpretacion

        for b in letrasProposicionales:
            if a[b] == 1:
                aux1[b] = 0 # Cambia el valor de verdad para b
            else:
                aux1[b] = i[b] # ... y mantiene el valor de verdad para las otras letras

        interps.append(aux1) # Incluye la nueva interpretacion en la lista

print('Interpretaciones: ')
for i in interps:
    print(i)
```

EJERCICIO 2: Implemente en Python una función que determine si una fórmula es válida, contingente o insatisfacible. Verifique su implementación con las siguientes fórmulas:

- a. $p \rightarrow (\neg p \rightarrow q) \Rightarrow$ Válida (y satisfacible)
- b. $\neg p \wedge q \Rightarrow$ Contingente (Satisfacible y falseable)
- c. $(\neg p \wedge (\neg p \rightarrow \neg q)) \wedge q \Rightarrow$ Insatisfacible (y falseable)