

"""Pregunta 2 45 puntos

Ingresar 6 números enteros y cargar un arreglo con un número positivo y otro negativo en ese orden. Si no hubiese números negativos o positivos, poner ceros en esas posiciones. Ejemplo: Se ingresan 1 -3 2 3 -4 10 El arreglo queda 1 -3 2 -4 3 0"""

#FUNCIONES

```
def ingresar_numeros(arre_num):
```

```
    i = 0
```

```
    while i < 6:
```

```
        num = int(input(f"Ingrese el número {i + 1}/6: "))
```

```
        i += 1
```

```
        arre_num.append(num)
```

```
    return arre_num
```

```
def construir_arreglo(arre_num):
```

```
    # 1. Separar los números por tipo (Positivos > 0, Negativos < 0)
```

```
    for num in arre_num:
```

```
        if num > 0:
```

```
            positivos.append(num)
```

```
        elif num < 0:
```

```
            negativos.append(num)
```

```
    # 2. Construir el arreglo final (tamaño 6)
```

```
    arreglo_final = []
```

```
    i = 0
```

```
    # El ciclo se repite 6 veces (de 0 a 5) para llenar las 6 posiciones
```

```
    while i < 6:
```

```
        # Posiciones pares (0, 2, 4) -> Necesitan un Positivo
```

```
        if i % 2 == 0:
```

```
            if len(positivos) > 0:
```

```
                # Tomar el primer positivo disponible y agregarlo
```

```
                arreglo_final.append(positivos.pop(0))
```

```
            else:
```

```
                # Si no hay positivos, poner cero
```

```
                arreglo_final.append(0)
```

```
        # Posiciones impares (1, 3, 5) -> Necesitan un Negativo
```

```
        else: # i % 2 != 0
```

```
            if len(negativos) > 0:
```

```
                # Tomar el primer negativo disponible y agregarlo
```

```
                arreglo_final.append(negativos.pop(0))
```

```
            else:
```

```
                # Si no hay negativos, poner cero
```

```
                arreglo_final.append(0)
```

```
        i += 1 # Incrementar el contador del ciclo principal
```

```
    return arreglo_final
```

```
# PROGRAMA PRINCIPAL
```

```
positivos = []
```

```
negativos = []
```

```
arreglo_final = []
```

```
print("\n--- Arreglo con númeors positivos y negativos ---\n")
```

```
print("A continuación, ingrese 6 números...\n")
```

```
# 1. Ingresar los 6 números
```

```
numeros_entrada = ingresar_numeros(arreglo_final)
```

```

# 2. Procesar y obtener el resultado
arreglo_resultante = construir_arreglo(numeros_entrada)

print("\n----- RESULTADO -----")
print("Números Ingresados (Entrada):")
print(numeros_entrada)
# 3. Aplicar la lógica de la condición final
if len(arreglo_resultante) > 0:
    print("El Arreglo (P, N, P, N, P, N) es:")
    print(arreglo_resultante)
else:
    # Esta condición de 'else' se ejecuta si el arreglo está vacío, lo cual no sucede
    # en este caso (el arreglo siempre tiene 6 elementos), pero se incluye para
    # cumplir la restricción solicitada.
    print("No se ingresaron valores")

print("\nFin del programa...")

"""
Entradas: 1, 2, 3, 4, 5, 6
*NOTA: El ejemplo en el enunciado dice: [1, 2, 3, 0, 0, 0]. EL código produce: [1, 0, 2, 0, 3, 0]
El resultado [1, 2, 3, 0, 0, 0] implicaría tomar 3 positivos seguidos y luego 3 ceros,
lo cual no cumple el patrón "un número positivo y otro negativo en ese orden".
Asumiendo que el patrón alternado es la REGLA, la salida debe ser [1, 0, 2, 0, 3, 0].
Si se tomará el primer positivo disponible al fallar el negativo, el resultado sería [1, 2, 3, 0, 0, 0],
pero eso no cumple el patrón estricto de P/N. La interpretación más estricta del patrón
alternado. """

```