

```

# Segunda parte del recuperatorio
""" Leer 8 numeros enteros y cargar un arreglo
con los primeros 6 numeros pares.
Si no hay numeros pares, cargarlos con ceros.
Y si no llegan a ser 6, completar hasta el sexto
con el ultimo valor par ingresado.
"""
def cargar_datos(arreglo):
    for i in range(0, 8):
        arreglo.append(int(input(f"Ingrese el número {i+1}/{8}: ")))

def completar_pares(arreglo, arreglo_resultado):
    # 1. Filtrar los primeros 6 pares de la entrada
    ultimo_par = 0
    for num in arreglo:
        if len(arreglo_resultado) < 6:
            if num % 2 == 0:
                arreglo_resultado.append(num)
                ultimo_par = num # Almacenar el último par encontrado

    while len(arreglo_resultado) < 6:

        if len(arreglo_resultado) == 0:
            # Si no hay pares (len==0), completar con ceros
            valor_completar = 0
            arreglo_resultado.append(valor_completar)
        else:
            # Si hay menos de 6, completar con el último par ingresado
            valor_completar = ultimo_par
            arreglo_resultado.append(valor_completar)

def mostrar(arreglo):
    print(arreglo)

print("\n-----\n")
entrada = [] # Arreglo de entrada (8 números)
resultado = [] # Arreglo de resultado (6 pares o ceros)
cargar_datos(entrada)
print("\n-----\n")
print("Cargar los 8 números de entrada...\n") # Fijo la carga a 8 números
print(entrada)

if len(entrada) == 8: # Verificamos que se hayan ingresado 8 números
    print("\n-----\n")
    completar_pares(entrada, resultado)

    # Después de la ejecución, len(resultado) siempre será 6 si len(entrada) era 8.
    if len(resultado) == 6:
        print("El arreglo de 6 elementos es:\n")
        mostrar(resultado)
    else:
        # Esto sería un error interno de la función completar_pares si ocurre.
        print("Error: El arreglo de resultado no tiene el tamaño correcto.")
else:
    # Este bloque solo se alcanza si cargar_datos no leyó 8 elementos
    print("\nNo se ingresaron los 8 datos requeridos.")

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

```