

Definición inicial de las variables.

```
term 1 = 0
term 2 = 0
term 3 = 0
temp-error-zona-1 = 0
temp-error-zona-2 = 0
registros = 0
```

```
def histograma (temp-error-zona-1, temp-error-zona-2, registros):
    asteriscos-1 = temp-error-zona-1
    guiones-1 = registros - asteriscos-1
    asteriscos-2 = temp-error-zona-2
    guiones-2 = registros - asteriscos-2
```

Utilizar conteo de datos erróneos y registro.

```
# print(asteriscos-1 * "*" + guiones-1 * "-")
print("Zona 1: ", int(asteriscos-1) * "*" + guiones-1 * "-")
print("Zona 2: ", int(asteriscos-2) * "*" + guiones-2 * "-")
```

Multiplicar n veces asteriscos y guiones.
Falta función para imprimir fila del histograma

-5pts

```
def promedio (temp 1, temp 2, temp 3 (term 0, term c, term e) term 1, term 2, term 3):
```

Ordenar para Facilidad de cálculo

```
if term 1 == "C":
    term-c = term 1
elif term 1 == "E":
    term-e = term 1
else:
```

Definir correctamente según letra ingresada

```
    term-o = term 1
if term 2 == "O":
    term-o = term 2
elif term 2 == "I":
    term-e = term 2
else:
    term-c = term 2
if term 3 == "O":
    term-o = term 3
elif term 3 == "C":
    term-c = term 3
else:
    term-e = term 3
```

```
# promedio-zona-1 = (term-o + term-c) / 2
```

```
# promedio-zona-2 = (term-c + term-e) / 2
```

```
# print ("
```

```
term-c = term-c - (term-c * 0.15)
```

```
term-e = term-e + (term-e * 0.025)
```

```
promedio-zona-1 = (term-o + term-c) / 2
```

```
promedio-zona-2 = (term-c + term-e) / 2
```

```
print ("Promedio de temperaturas tras ajuste:")
```

```
print ("Zona 1: %.2f" % (promedio-zona-1))
```

```
print ("Zona 2: %.2f" % (promedio-zona-2))
```

Corrección de desviaciones conocidas.

Promedios tras ajuste.

Print datos calculados.

```
def evaluar_datos(promedio_zona-1, promedio_zona-2): # función conteo
    if promedio_zona-1 < 16 or promedio_zona-1 > 25: # de datos erróneos
        temp_error_zona-1 += 1 # no viables.
    if promedio_zona-2 < 16 or promedio_zona-2 > 25:
        temp_error_zona-2 += 1
```

debería ser una función que retorne opción válida (-2pts)

```
def menu(registros):
    try: # En caso de fallar, pedir valores correctos.
        opcion = int(input("1=Registro 2=Histograma 3=Salir"))
        if opcion == 1: # Ingreso de valores
            term1 = str(input("Termómetro 1: "))
            temp1 = float(input("Temperatura 1: "))
            term2 = str(input("Termómetro 2: "))
            temp2 = float(input("Temperatura 2: "))
            term3 = str(input("Termómetro 3: "))
            temp3 = float(input("Temperatura 3: "))
            ordenar = term(term1, term2, term3)
            promedio(temp1, temp2, temp3, term1, term2, term3)
            registros += 1 # Sumar registro exitoso.
            evaluar_datos(promedio_zona-1, promedio_zona-2) # Func. para estadísticas histograma.
            menu(registros) # Volver al menú.
        elif opcion == 2: # llamar histograma.
            histograma(temp_error_zona-1, temp_error_zona-2, registros) # Volver al menú.
        elif opcion == 3: # Salida con mensaje.
            print("Gracias por usar el sistema!")
        else: # Opción de menú inválida.
            print("Ingrese una opción válida") # Volver al menú.
            menu(registros)
```

```
except:
    print("Datos incorrectos, vuelva a ingresar los valores correctamente")
    menu(registros)
    # Datos no cumplen con definición str() o float() puesta en la
    # definición de las variables, se regresa al menú con advertencia.
```

```
print("Bienvenido al sistema!")
menu(registros)
```

Mensaje inicialización
Invocar menú interactivo

Están en el bloque principal
aunque no parezca.

Ver última hoja
con anotaciones

Rubros	Puntu
1	40
2	13
3	10
4	30
Total	(93)