



## **ALGORITMOS Y ESTRUCTURAS DE DATOS (TSDS)**

ASIGNATURA:

ALGORITMOS Y ESTRUCTURAS DE DATOS

PROFESOR:

Ing. Lorena Chulde MSc.

PERÍODO ACADÉMICO:

2023-B

### **TAREA 6 Grupal**

#### **TÍTULO:**

**DISEÑO DE ALGORITMOS ESTRUCTURAS ITERATIVAS**

**Nombre de los estudiantes:**

**Guerra Lovato Josué Eduard**

**Pérez Orosco Carlos David**



**2023-B**

## PROPÓSITO DE LA TAREA

Aplicar sentencias de algoritmos mediante las estructuras de cíclicas WHILE, FOR para la resolución de ejercicios sencillos.

## Taller en clase:

Promedio aritmético

$$\bar{X} = \frac{X_1 + X_2 + X_3 + \dots + X_n}{n}$$

1. Calcule la media de varias notas ingresadas por teclado. El usuario ingresará tantas notas hasta que ingrese el "0".

```
#Ejercicio 1-Media.py > ...
1  #Ejercicio 1
2  # Pedir al usuario un número
3  # Realizar la suma de dichos números /Sacar el promedio
4  # Ejercicio echo por Carlos Pérez
5
6  print(["Ejercicio 1:\npromedio de un número"])
7  numeros = []
8  while True:
9      try:
10         numero = float(input("Ingrese una nota: "))
11         print("Ingrese 0 para salir\n")
12         if numero == 0:
13             break
14         numeros.append(numero)
15     except ValueError:
16         print("Por favor, ingrese un número válido.")
17 if not numeros:
18     print("No se ingresaron notas.")
19 else:
20     suma_numero = sum(numeros)
21     cantidad_numeros = len(numeros)
22     promedio = suma_numero / cantidad_numeros
23     print("Notas ingresadas: ", numeros )
24     print("Cantidad de notas: ", cantidad_numeros)
25     print("Suma de notas: ", suma_numero)
26     print("Promedio de notas: ",round(promedio, 2))
27 print("Creado por Carlos perez")
```

Ejecución del programa: Al iniciar pedirá al usuario cualquier nota el cual se acumulará dentro de un bucle, que al final dará el resultado de toda la operación junto con los números ingresados haciendo los respectivos cálculos tanto como suma como promedio.

```
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber > & C:/Users/Josué/
oritos y estructuras de datos/Deberes/Deber 6/#Ejercicio 1-Media.py
Ejercicio 1:
promedio de un número
Ingrese una nota: 16
Ingrese 0 para salir

Ingrese una nota: 20
Ingrese 0 para salir

Ingrese una nota: 14
Ingrese 0 para salir

Ingrese una nota: 15
Ingrese 0 para salir

Ingrese una nota: 12
Ingrese 0 para salir

Ingrese una nota: 20
Ingrese 0 para salir

Ingrese una nota: 18
Ingrese 0 para salir

Ingrese una nota: 0
Ingrese 0 para salir

Notas ingresadas: [16.0, 20.0, 14.0, 15.0, 12.0, 20.0, 18.0]
Cantidad de notas: 7
Suma de notas: 115.0
Promedio de notas: 16.43
Creado por Carlos perez
```

2. Dado un número entero positivo, mostrar su factorial. El factorial de un número se obtiene multiplicando todos los números enteros positivos que hay entre el 1 y ese número.

$$\text{Factorial of a Number}$$

$$n! = n * (n-1) * (n-2) * \dots * 1$$

```
#Ejercicio 2-Factorial.py > ...
1  # Ejercicio 2: cálculo de factorial
2  # Pedir al usuario un número
3  # Realizar el cálculo factorial
4  # Ejercicio realizado por Carlos Pérez
5
6  num = int(input("Ejercicio 4: Cálculo de factorial\nIngrese un número para calcular su factorial: "))
7
8  if num < 0:
9      print("El factorial no puede ser negativo.")
10 else:
11     factorial = 1
12     print(f"Fórmula del factorial: {num}! = ", end="")
13     for i in range(1, num + 1):
14         print(i, end="")
15         if i < num:
16             print(" * ", end="")
17     print()
18     for i in range(1, num + 1):
19         factorial *= i
20     print("El factorial de", num, " es: ", factorial)
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber 6> & C:/Users/Josué/AppData/Local/Microsoft/WindowsApps/oritmos y estructuras de datos/Deberes/Deber 6/#Ejercicio 2-Factorial.py
Ejercicio 4: Cálculo de factorial
Ingrese un número para calcular su factorial: 8
Fórmula del factorial: 8! = 1 * 2 * 3 * 4 * 5 * 6 * 7 * 8
El factorial de 8 es: 40320
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber 6> |
```

El programa almacena en una cadena de if-else el procedimiento a mostrar por lo que el usuario al ingresar un número al azar, inmediatamente se mostrara en pantalla la fórmula del factorial y su resultado.

3. Crear un bucle que cuente los números pares e impares del 1 al 100, los números deberán ser ingresados por teclado, hasta que el usuario digite "0". Al finalizar, informar la cantidad de números pares y de impares leídos en total.



EL RESULTADO DEL EJERCICIO SE MUESTRA EN LA SIGUIENTE PÁGINA

```
#Ejercicio 3-Pares_e_Impares.py > ...
1  # Ejercicio 3
2  # Pedir al usuario un número
3  # Ejercicio realizado por Josué Guerra
4
5  print("Ejercicio 1: Cantidad de pares o impares")
6
7  while True:
8      print("El programa se está iniciando...")
9      contador_pares = 0
10     contador_impares = 0
11     while True:
12         try:
13             numero = int(input("Ingrese un número (1-100): "))
14             if 0 <= numero <= 100:
15                 if numero == 0:
16                     break
17                 contador_pares += numero % 2 == 0
18                 contador_impares += numero % 2 != 0
19             else:
20                 print("Número fuera del rango permitido (1-100). Inténtelo de nuevo.")
21         except ValueError:
22             print("Por favor, ingrese un número válido.")
23     print("\nResumen:")
24     print("Cantidad de números pares:", contador_pares)
25     print("Cantidad de números impares:", contador_impares)
26     if numero == 0:
27         print("Finalizando programa")
28         break
```

### Ejecución del programa:

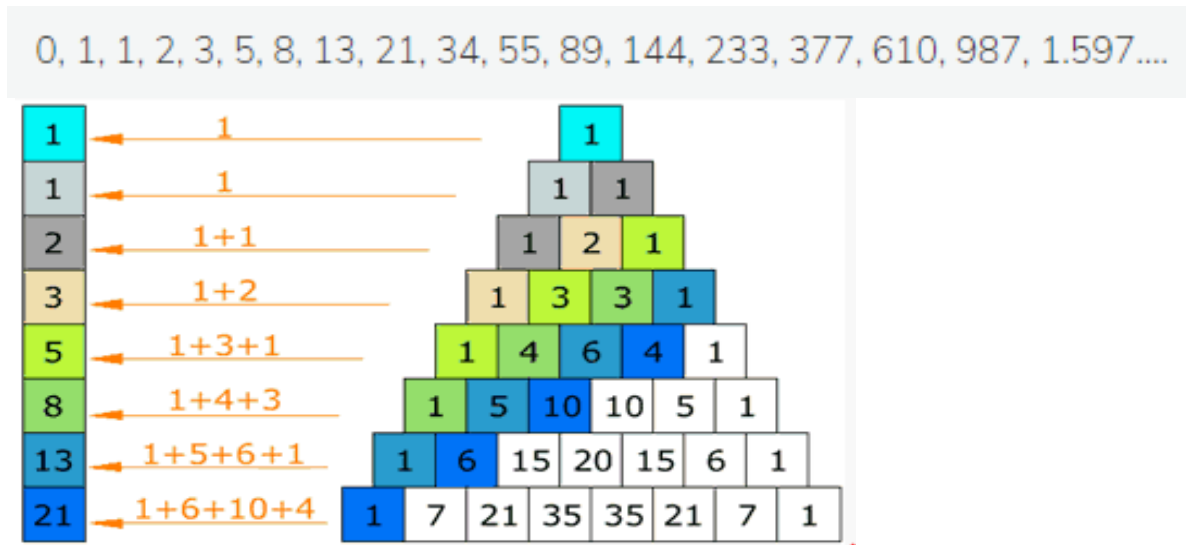
```
27     print("Finalizando programa")
28     break
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

```
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber > & C:/Users/Josué/
oritos y estructuras de datos/Deberes/Deber 6/#Ejercicio 3-Fibonacci.py"
Ejercicio 1: Cantidad de pares o impares
El programa se está iniciando...
Ingrese un número (1-100): 5
Ingrese un número (1-100): 110
Número fuera del rango permitido (1-100). Inténtelo de nuevo.
Ingrese un número (1-100): 6
Ingrese un número (1-100): 8
Ingrese un número (1-100): 0
```

El programa pedirá al usuario que inserte una serie de números al azar el cual se va almacenando, hasta que se presione 0 al final muestra la cantidad de pares e impares

4. Crear un programa que imprima la Sucesión de Fibonacci, desde el número 0 hasta el 1597, horizontalmente. (7 líneas de código)



```
#Ejercicio 4-Fibonacci.py > ...
1 # Ejercicio 7: Serie de Fibonacci /Josué Guerra
2 # Pedir al usuario n números
3 # Se mostrará hasta ese número o el anterior si es el caso
4 print("Ejercicio 7: Serie de Fibonacci\n")
5 n = 1597; a, b = 0, 1
6 serie = [a]
7 while a <= n:
8     a, b = b, a + b
9     serie.append(a)
10 print("Serie de Fibonacci hasta", n, ":", *serie, sep=", ")
11
12
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber 6> & C:/Users/Josué/AppData/Local/Microsoft/Windows/Maps/Algoritmos y estructuras de datos/Deberes/Deber 6/#Ejercicio 4-Fibonacci.py

Ejercicio 7: Serie de Fibonacci

Serie de Fibonacci hasta, 1597, :, 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber 6> |

El programa está resumido en 7 Líneas de código las 3 primeras son comentarios, los cuales se indicaron en clase que deberían estar tanto el ejercicio, la orden como el nombre del creador del código.

# Tarea

## Realizar los siguientes ejercicios



### 1. While

Escriba un programa que simule un banco. El programa solicitará primero una cantidad, que será la cantidad de dinero que queremos ahorrar. A continuación, el programa solicitará una y otra vez las cantidades que se irán ahorrando, hasta que el total ahorrado iguale o supere al objetivo (\$ 1000). El programa deberá comprobar que las cantidades sean positivas, no se permitirán ingresar cantidades negativas.

**Use WHILE o FOR, según crea conveniente**



```
#Ejercicio 1-Tarea-Ahorro.py X #Ejercicio 1-Media.py
Ejercicios de deber > #Ejercicio 1-Tarea-Ahorro.py > ...
1 # Ejercicio 1
2 # Pedir al usuario una cantidad para ahorrar
3 # Ejercicio echo por Carlos Perez
4
5 print ("---Bienvenido---")
6 ahorro = float(input("Ingrese la cantidad de dinero que desea ahorrar: "))
7 suma_ahorro = 0
8
9 while ahorro < 0:
10     if ahorro < 0:
11         print("No puede ahorrar cantidades negativas")
12         ahorro = float(input("Ingrese la cantidad de dinero que desea ahorrar: "))
13 while suma_ahorro < ahorro:
14     dinero = float(input("Ingrese el valor de su deposito: "))
15     if dinero < 0:
16         print("No puede ingresar cantidades negativas")
17     else:
18         suma_ahorro = suma_ahorro + dinero
19         print(suma_ahorro, " + ", dinero, " = ", suma_ahorro)
20 print("Felicitades usted cumple su meta :D")

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber > & C:/Users/Josué/AppData/L
oritos y estructuras de datos/Deberes/Deber 6/Ejercicios de deber/#Ejercicio 1-Tarea-Ahorro.py"
---Bienvenido---
Ingrese la cantidad de dinero que desea ahorrar: 150
Ingrese el valor de su deposito: 25
25.0 + 25.0 = 25.0
Ingrese el valor de su deposito: 60
85.0 + 60.0 = 85.0
Ingrese el valor de su deposito: 75
160.0 + 75.0 = 160.0
Felicitades usted cumple su meta :D
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber 6> |
```

2. Dado un número, cuente el número total de dígitos de un número. Por ejemplo, el número es 75869, por lo que la salida debería ser 5.

$$\begin{array}{r}
 123 : 10 = 12 \\
 12 : 10 = 1 \\
 1 : 10 = 0
 \end{array}
 \quad
 \begin{array}{l}
 R=3 \\
 R=2 \\
 R=1
 \end{array}
 \quad
 \begin{array}{l}
 1 \\
 1 \\
 1
 \end{array}
 \left. \vphantom{\begin{array}{l} 1 \\ 1 \\ 1 \end{array}} \right\} + 3$$

Ejercicios de deber > #Ejercicio 2-Tarea-Contador.py > ...

```

1  # Ejercicio 2
2  # Pedir al usuario una número al azar y contar cuantos caracteres tiene
3  # Ejercicio echo por Carlos Perez
4
5  print("Ejercicio 2: Contador")
6  def contar_digitos(numero):
7      contador = 0
8      while numero != 0:
9          numero = numero // 10
10         contador = contador + 1
11     return contador
12
13     numero_a_contar = float(input("Ingrese el numero que quiera: "))
14     print("El numero de digitos es de: ",contar_digitos(numero_a_contar))

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber > & C:/Users/Josué/AppData/oritmos y estructuras de datos/Deberes/Deber 6/Ejercicios de deber/#Ejercicio 2-Tarea-Contador.py
Ejercicio 2: Contador
Ingrese el numero que quiera: 17658
El numero de digitos es de: 5
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber 6>

```

3. Mostrar un menú con tres opciones:

- 1- comenzar programa,
- 2- imprimir listado,
- 3-finalizar programa.

A continuación, el usuario debe poder seleccionar una opción (1, 2 ó 3). Si elige una opción incorrecta, informarle del error. El menú se debe volver a mostrar luego de ejecutada cada opción, permitiendo volver a elegir. Si elige las opciones 1 ó 2 se imprimirá un texto. Si elige la opción 3, se interrumpirá la impresión del menú y el programa finalizará.

LA EJECUCIÓN Y EL CÓDIGO DEL PROBLEMA SE ENCUENTRA EN LA SIGUIENTE HOJA.

```

Ejercicios de deber > #Ejercicio 3-Tarea-Menu.py > ...
1  # Ejercicio 3
2  aux = 0
3  while aux != 3:
4      print ("---Bienvenido---")
5      print ("1. Comenzar programa")
6      print ("2. Imprimir Listado")
7      print ("3. Finalizar programa")
8      aux = int (input ("Ingresa tu seleccion: "))
9      if aux != 1 and aux != 2 and aux != 3:
10         print ("Seleccion invalida")
11     elif aux == 1:
12         print ("El programa se esta iniciando...")
13     elif aux == 2:
14         print ("1, 2, 3 , 4 , 5, 6, 7, 8, 9, 10")

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber 6\Ejercicios de deber\Ejercicio 3-Tarea-Menu.py
---Bienvenido---
1. Comenzar programa
2. Imprimir Listado
3. Finalizar programa
Ingresa tu seleccion: 1
El programa se esta iniciando...
---Bienvenido---
1. Comenzar programa
2. Imprimir Listado
3. Finalizar programa
Ingresa tu seleccion: 2
1, 2, 3 , 4 , 5, 6, 7, 8, 9, 10
---Bienvenido---
1. Comenzar programa
2. Imprimir Listado
3. Finalizar programa
Ingresa tu seleccion: 3
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber 6>

```

4. Crear un programa que permita al usuario ingresar los montos de las compras de un cliente (se desconoce la cantidad de datos que cargará, la cual puede cambiar en cada ejecución), cortando el ingreso de datos cuando el usuario ingrese el monto 0 .



Si ingresa un número negativo, no se debe procesar y se debe pedir que ingrese un nuevo monto. Al finalizar, informar el total a pagar teniendo que cuenta que, si las ventas superan el total de \$1000, se le debe aplicar un 10% de descuento.

LA EJECUCIÓN Y EL CÓDIGO DEL PROBLEMA SE ENCUENTRA EN LA SIGUIENTE HOJA.



```
Ejercicios de deber > #Ejercicio 4-Tarea-Compra.py > ...
1  # Ejercicio 4
2  # Pedir al usuario la cantidad de compra y sumar al final
3  # Ejercicio echo por Carlos Perez
4
5  print("Ejercicio 4 : Compra")
6  suma = 0
7  numero = 1
8  contador = 0
9
10 while numero != 0:
11     numero = float(input("Ingrese el valor de su compra: "))
12     print("Si desea finalizar ingrese 0.")
13     suma = suma + numero
14     contador += 1
15     print(suma)
16
17 if suma >= 1000:
18     respuesta = suma - (suma * 0.1)
19     print("El total a pagar es de: ", respuesta)
20 else:
21     print("El total a pagar es de: ", suma)
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

```
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber > & C:/Users/Josué/AppData/Local/Programs/Python/Python39-6/Scripts/python.exe D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber 6\Ejercicios de deber/#Ejercicio 4-Tarea-Compra.py
Ejercicio 4 : Compra
Ingrese el valor de su compra: 120
Si desea finalizar ingrese 0.
120.0
Ingrese el valor de su compra: 16
Si desea finalizar ingrese 0.
136.0
Ingrese el valor de su compra: 45
Si desea finalizar ingrese 0.
181.0
Ingrese el valor de su compra: 0
Si desea finalizar ingrese 0.
181.0
El total a pagar es de: 181.0
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber 6> |
```

Se ingresan valores al azar lo cual se van almacenando dentro de un bucle hasta que el usuario inserte el número 0, e inmediatamente de presionar el valor de 0 saldrá el total a pagar.

5. El día de la madre usted desea comprarle un regalo a su madre, usted visita almacenes Coral y decide comprarle varios objetos que le podrían gustar a su madre.



El almacén tiene clasificados los artículos, según las siguientes categorías:  
 Perfumería, Joyería, Maquillaje, Ropa  
 Cada categoría tiene los siguientes artículos:

Perfumería	Costo \$	Joyería	Costo \$
Tentación	30	Aretes	7
Primavera	28	Collar	5
Otoño	15	Cadena	20
Seducción	35	Pulsera	15
Maquillaje		Ropa	
Sombras	8	Blusa	25
Maquillaje	5	Chaqueta	60
Labiales	4	Pantalón	18
Rimel	6	Abrigo	90

Para adquirir los artículos, deberá seleccionar del menú **"Tipo de categoría"**, la categoría y luego podrá comprar el artículo.

Una vez que compre el artículo, el sistema le preguntará el subtotal de lo comprado y además, le preguntará si desea comprar otro artículo de esa categoría, si ya no desea comprar artículos de esa categoría, deberá salir al menú **"Tipo de categoría"**, y seleccionar otra categoría y seguir comprando artículos.

Al finalizar la compra, el sistema le presentará el subtotal de la compra con la cantidad de artículos comprados por categoría y por la compra total.

Por favor realice todos los controles que crea conveniente.

EL CÓDIGO Y SU EJECUCIÓN SE ENCUENTRAN EN LA SIGUIENTE PÁGINA

```

Ejercicios de deber > #Ejercicio 5-Compra Madre.py > ...
1  # Ejercicio 5
2  # Compra por el día de la madre
3  # Ejercicio hecho por Josué Guerra
4
5  print("Ejercicio 5: Compra día de la madre")
6
7  print("---/--/--/ Centro comercial el CORAL /-/--/--")
8  print("Bienvenido")
9  print("Este es el sistema del centro comercial el CORAL")
10 print(" 1. Entrar categorías de compra")
11 print(" 2. Mostrar factura")
12 print(" 3. Salir ")
13 categorias = {
14     1: {"Perfumería": {"Tentación": 30, "Primavera": 28, "Otoño": 15, "Seducción": 35}},
15     2: {"Joyería": {"Aretes": 7, "Collar": 5, "Cadena": 20, "Pulsera": 15}},
16     3: {"Maquillaje": {"Sombras": 8, "Maquillaje": 5, "Labiales": 4, "Rimel": 6}},
17     4: {"Ropa": {"Blusa": 25, "Chaqueta": 60, "Pantalón": 18, "Abrigo": 90}}
18 }
19 subtotal_total = 0
20 articulos_comprados = {categoria: 0 for categoria in categorias}
21
22 while True:
23     opcion = int(input("Seleccione una opción: "))
24     if opcion == 1:
25         print("\nCategorías disponibles:")
26         for num_categoria, categoria in categorias.items():
27             print(num_categoria, "-", list(categoria.keys())[0])
28         num_categoria_elegida = int(input("Ingrese el número de la categoría de compra: "))
29         if num_categoria_elegida in categorias:
30             categoria_elegida = list(categorias[num_categoria_elegida].keys())[0]
31             while True:
32                 print("\nArtículos disponibles en", categoria_elegida + ":")
33                 for num_articulo, (articulo, precio) in enumerate(categorias[num_categoria_elegida][categoria_elegida].items(), 1):
34                     print(num_articulo, "-", articulo, "$", precio)
35                 num_articulo_elegido = int(input("Ingrese el número del artículo que desea comprar: "))
36                 if 1 <= num_articulo_elegido <= len(categorias[num_categoria_elegida][categoria_elegida]):
37                     articulo_elegido = list(categorias[num_categoria_elegida][categoria_elegida].keys())[num_articulo_elegido - 1]
38                     cantidad = int(input("Ingrese la cantidad de unidades: "))
39                     subtotal_total += categorias[num_categoria_elegida][categoria_elegida][articulo_elegido] * cantidad
40
41                 if categoria_elegida in articulos_comprados:
42                     articulos_comprados[categoria_elegida] += cantidad
43                 else:
44                     articulos_comprados[categoria_elegida] = cantidad
45                 comprar_otro = input("¿Desea comprar otro artículo de esta categoría? (S/N): ")
46                 if comprar_otro.upper() != "S":
47                     break
48             else:
49                 print("Artículo no válido. Inténtelo de nuevo.")
50         else:
51             print("Categoría no válida. Inténtelo de nuevo.")
52
53     elif opcion == 2:
54         print("\nMostrar factura:")
55         for categoria, cantidad in articulos_comprados.items():
56             if cantidad > 0:
57                 subtotal_categoria = sum([categorias[num_categoria][categoria_articulo][articulo] * cantidad for categoria_articulo in categorias[num_categoria] for articulo, _ in categorias[num_categoria][categoria_articulo].items()])
58                 print(categoria + ":", cantidad, "unidades - Subtotal: $", subtotal_categoria)
59         print("\nTotal de la compra:", subtotal_total)
60     elif opcion == 3:
61         print("Saliendo del sistema. ¡Gracias por su compra!")
62         break
63     else:
64         print("Opción no válida. Inténtelo de nuevo.")
65

```

### COMPLEMENTO DE CAPTURA DE LA LÍNEA 57:

```

56
57 :categoria_articulo][articulo] * cantidad for categoria_articulo in categorias[num_categoria] for articulo, _ in categorias[num_categoria][categoria_articulo].items())
58 :", subtotal_categoria)

```

### EJECUCIÓN DEL CÓDIGO EN LA SIGUIENTE PÁGINA:

```

Ejercicios de deber > #Ejercicio 5-Compra Madre.py > ...
35 num_articulo_elegido = int(input("Ingrese el número del artículo que desea comprar: "))

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber > & C:/Users/Josué/AppData/Local/Microsoft/WindowsApps/python3.11.exe
os/Deberes/Deber 6/Ejercicios de deber/#Ejercicio 5-Compra Madre.py"
Ejercicio 5: Compra día de la madre
--/--/--/ Centro comercial el CORAL /--/--/--
Bienvenido
Este es el sistema del centro comercial el CORAL
1. Entrar categorías de compra
2. Mostrar factura
3. Salir
Seleccione una opción: 1

Categorías disponibles:
1 - Perfumería
2 - Joyería
3 - Maquillaje
4 - Ropa
Ingrese el número de la categoría de compra: 3

Artículos disponibles en Maquillaje:
1 - Sombras $ 8
2 - Maquillaje $ 5
3 - Labiales $ 4
4 - Rimel $ 6
Ingrese el número del artículo que desea comprar: 2
Ingrese la cantidad de unidades: 8
¿Desea comprar otro artículo de esta categoría? (S/N): N
Seleccione una opción: 2

Mostrar factura:
Maquillaje: 8 unidades - Subtotal: $ 1544

Total de la compra: 40
Seleccione una opción: 3
Saliendo del sistema. ¡Gracias por su compra!
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber 6>

```

Como tal se presenta un menú para que el usuario pueda ingresar a las diferentes acciones, también se muestra las categorías y al presionar que desea ver la factura mostrara con el subtotal y el total a pagar, para finalizar o salir del sistema el usuario deberá presionar el número 3 y con eso podrá finalizar la compra.

6. Crear un programa que permita al usuario ingresar títulos de libros por teclado, finalizando el ingreso al leerse el string "\*" (asterisco). Cada vez que el usuario ingrese un string de longitud 1 que contenga sólo una barra ("/") se considera que termina una línea. Por cada línea completa, informar cuántos dígitos numéricos (del 0 al 9) aparecieron en total (en todos los títulos de libros que componen en esa línea).

Finalmente, informar cuántas líneas completas se ingresaron.

Ejemplo de ejecución:

Libro: Los 3 mosqueteros  
 Libro: Historia de 2 ciudades  
 Libro: /  
 Línea completa. Aparecen 2 dígitos numéricos.  
 Libro: 20000 leguas de viaje submarino  
 Libro: El señor de los anillos  
 Libro: /  
 Línea completa. Aparecen 5 dígitos numéricos.  
 Libro: 20 años después  
 Libro: \*  
 Fin. Se leyeron 2 líneas completas.



Ejercicios de deber > #Ejercicio 6-Tarea-Libros > ...

```

1  # Ejercicio 6
2  # Pedir al usuario los títulos de libros
3  # Ejercicio echo por Josué Guerra
4
5  print("Ejercicio 6: Conteo de dígitos numéricos en títulos de libros")
6
7  lineas_completas = 0
8  digitos_totales = 0
9  print(" Use * para acabar y muestre las lineas completas")
10 print(" Use / para ver los dígitos numéricos de un título")
11 while True:
12     titulo = input("Libro: ")
13
14     if titulo == "*":
15         break
16     elif titulo == "/":
17         print("Línea completa. Aparecen", digitos_totales, "dígitos numéricos.")
18         lineas_completas += 1
19         digitos_totales = 0
20     else:
21         digitos_totales += sum(1 for caracter in titulo if caracter.isdigit())
22 print("Fin. Se leyeron", lineas_completas, "líneas completas.")

```

## Ejecución del código

```
22 print("Fin. Se leyeron", lineas_completas, "líneas completas.")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber > & C:/Users/Josué/AppData/L
oritos y estructuras de datos/Deberes/Deber 6/Ejercicios de deber/#Ejercicio 6-Tarea-Libros"
Ejercicio 6: Conteo de dígitos numéricos en títulos de libros
 Use * para acabar y muestre las lineas completas
 Use / para ver los dígitos numéricos de un título
Libro: Los 3 mosqueteros
Libro: Blancanieves y los 7 enanitos
Libro: Nunca Jamas
Libro: /
Línea completa. Aparecen 2 dígitos numéricos.
Libro: Los 100 y nunca quien
Libro: Atravez de mi ventana
Libro: /
Línea completa. Aparecen 3 dígitos numéricos.
Libro: *
Fin. Se leyeron 2 líneas completas.
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Deberes\Deber 6>

```

**A continuación se presenta el enlace de GITHUB donde se encuentran los códigos de los ejercicios realizados en el deber.**

**ENLACE GITHUB:**

<https://github.com/JosueGuerra2023B/Estructuras-Datos2023B/tree/master/Deberes/Deber%206>

**ENTREGABLES:**

- Una vez culminada tu tarea, captura las pantallas de la ejecución del problema con tus datos y súbela en el apartado del aula virtual "S6-Tarea-6:
- Sube los ejercicios al git o al drive y entrega la url de los archivos .py
- Recuerda el nombre del archivo deberá ser: **Tarea6\_Algoritmos\_2023B\_NApellido**.

**RECURSOS NECESARIOS**

- Acceso a Internet.
- Imaginación.
- VSC