



ALGORITMOS Y ESTRUCTURAS DE DATOS (TSDS)

ASIGNATURA:

ALGORITMOS Y ESTRUCTURAS DE DATOS

PROFESOR:

Ing. Lorena Chulde MSc.

PERÍODO ACADÉMICO:

2023-B

LABORATORIO 1

Grupal

TÍTULO:

FUNCIONES

Nombres de los estudiantes:

Guerra Lovato Josué
Pérez Orosco Carlos
Soria Ansa Richard

```
int sumar(int numero1, int numero2)
{
    int suma = numero1 + numero2;
    return suma;
}
```

Diagram illustrating the components of a function definition:

- Parámetros de entrada** (Input Parameters): Points to the parameters `int numero1` and `int numero2` in the function signature.
- Instrucciones** (Instructions): Points to the body of the function, specifically the line `int suma = numero1 + numero2;`.
- Nombre de la función** (Function Name): Points to the function name `sumar` in the signature.
- Valor de retorno** (Return Value): Points to the `return` statement and the variable `suma`.

2023-B

PROPÓSITO DE LA TAREA

Reutilizar el código mediante funciones para una programación óptima.

OBJETIVO GENERAL

El propósito principal de las funciones y la recursividad en Python es ofrecer soluciones efectivas y sofisticadas para resolver problemas que requieren repeticiones o cálculos complejos. La recursividad permite que una función se llame a sí misma, lo que puede ser muy útil para abordar problemas mediante la subdivisión en subproblemas similares al problema original. Este enfoque busca reemplazar estructuras repetitivas con una solución más elegante y eficiente. Sin embargo, es crucial aplicar la recursividad con cautela, definiendo casos base, gestionando la asignación de memoria y evitando bucles infinitos.

Parte I

TALLER:

1. Pedir que el usuario ingrese valores hasta que ingrese el cero. Por cada valor, mostrar la suma de sus dígitos. Mediante una función realice la suma.

```
ejercicio_1.py > ...
1  # Pedir que el usuario ingrese valores hasta que ingrese el cero. Por cada valor,
2  # mostrar la suma de sus dígitos. Mediante una función realice la suma.
3  #Funcion
4  def Calcularsuma(numero):
5      suma = 0
6      while(numero!=0):
7          digito = numero%10
8          suma += digito
9          numero = numero//10
10     return suma
11
12     #Programa Principal
13
14     numero = int(input("Ingrese el número a analizar: "))
15
16     while (numero!=0):
17         print("La suma del número ingresado es:",Calcularsuma(numero))
18         numero = int(input("Ingrese el número a analizar: "))
19
20     # Richard Soira, Josue Guerra, Carlos Pérez
```

Ejecución:

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> & C:/Users/Jo
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejercicio_1.py"
Ingrese el número a analizar: 12
La suma del número ingresado es: 3
Ingrese el número a analizar: 5
La suma del número ingresado es: 5
Ingrese el número a analizar: 0
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> █
```

- Determinar la suma de las coordenadas de salida, dado que un usuario ingresa las mismas por teclado.

$$X = x + 12$$

$$Y = y + 15$$

```

ejercicio_2.py > ...
1  # Determinar la suma de las coordenadas de salida, dado que un usuario ingresa las mismas por teclado
2  def calcularCoordenadas(x,y):
3      x += 12
4      y += 15
5      suma = x+y
6      return suma
7
8  x = int(input("Ingrese el valor de la coordenada x: "))
9  y = int(input("Ingrese el valor de la coordenada y: "))
10
11 print("La suma de las coordenadas", x, y, "es:", calcularCoordenadas(x,y))
12
13 # Richard Soria, Josué Guerra, Carlos Pérez

```

Ejecución:

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> & C:/Users/Jos
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejercicio_2.py"
Ingrese el valor de la coordenada x: 360
Ingrese el valor de la coordenada y: 4580
La suma de las coordenadas 360 4580 es: 4967
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase>

```

- Solicitar al usuario que ingrese valores hasta que ingrese el cero. Por cada valor ingresado sumar sus dígitos y mostrarlos en pantalla. Al finalizar, calcular la suma total de todos los valores ingresados y la suma de sus dígitos.

```

ejercicio_3.py > ...
1  #Solicitar al usuario que ingrese valores hasta que ingrese el cero. Por cada
2  #valor ingresado sumar sus dígitos y mostrarlos en pantalla. Al finalizar,
3  #calcular la suma total de todos los valores ingresados y la suma de sus dígitos.
4
5  #Funcion
6  def Calcularsuma(numero):
7      suma = 0
8      while(numero!=0):
9          digito = numero%10
10         suma += digito
11         numero = numero//10
12     return suma
13
14 #Programa Principal
15
16 numero = int(input("Ingrese el número a analizar: "))
17
18 suma_numero = 0
19 total_digitos = 0
20 while (numero!=0):
21     suma_numero+=numero
22     digitos = Calcularsuma(numero)
23     total_digitos += digitos
24     print("La suma del número ingresado es:",Calcularsuma(numero))
25     print("La suma de los números ingresado es:", suma_numero)
26     print("La suma de todos los digitos es:", total_digitos)
27     numero = int(input("Ingrese el número a analizar: "))
28
29 # Richard Soria, Josué Guerra, Carlos Pérez

```

Ejecución:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> & C:\Users\J...
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejercic
Ingrese el número a analizar: 16
La suma del número ingresado es: 7
La suma de los números ingresado es: 16
La suma de todos los dígitos es: 7
Ingrese el número a analizar: 12
La suma del número ingresado es: 3
La suma de los números ingresado es: 28
La suma de todos los dígitos es: 10
Ingrese el número a analizar: 0
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> |
```

4. Solicitar al usuario que ingrese un número entero e imprimir en pantalla si es primo o no. Utilice una función booleana que lo decida.

```
ejercicio_4.py > ...
1  #Solicitar al usuario que ingrese un número entero e imprimir en
2  #pantalla si es primo o no. Utilice una función booleana que lo decida.
3
4  def numeroPrimo(numero):
5      contador = 0
6      for i in range(1,numero+1):
7          if ((numero%i)==0):
8              contador+=1
9      return contador == 2
10 numero = int(input("Ingrese el número a validad: "))
11 if numeroPrimo(numero):
12     print("El número es primo")
13 else:
14     print("El número no es primo")
15
16 print(numeroPrimo(numero))
17
18 # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> & C:/Users/J...
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejercicio_4.py"
Ingrese el número a validad: 18
El número no es primo
False
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> & C:/Users/J...
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejercicio_4.py"
Ingrese el número a validad: 5
El número es primo
True
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> |
```

5. Imprimir en pantalla la cantidad de ocurrencias de un dígito que se encuentra en un número entero ingresado por el usuario.

Nota: El usuario digitará tanto el número entero como el dígito.

```
ejercicio_5.py > ...
1  # Imprimir en pantalla la cantidad de ocurrencias de un dígito que se encuentra
2  # en un número entero ingresado por el usuario. Nota: El usuario digitará tanto
3  # el número entero como el dígito. Utilice una función que calcule la frecuencia.
4
5  def ocurrencias(numero, digito):
6      cantidad_digitos = 0
7      for i in str(numero):
8          if i == str(digito):
9              cantidad_digitos += 1
10     return cantidad_digitos
11
12     numero = int(input("Ingrese el número: "))
13     digito = int(input("Ingrese el dígito que desea conocer su ocurrencia: "))
14     print("El dígito se repite:", ocurrencias(numero, digito))
15
16     # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> & C:/
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejercici
Ingrese el número: 1257813216879415987461
Ingrese el dígito que desea conocer su ocurrencia: 5
El dígito se repite: 2
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> & C:/
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejercici
Ingrese el número: 159687423571852204
Ingrese el dígito que desea conocer su ocurrencia: 3
El dígito se repite: 1
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> |
```

6. Calcular el factorial de un número ingresado por el usuario.

Nota: Utilice una o más funciones, según sea necesario.

Utilice una función que calcule la frecuencia.

```
ejercicio_6.py > ...
1  # Calcular el factorial de un número ingresado por el usuario. Nota: Utilice una o
2  # más funciones, según sea necesario. Utilice una función que calcule la frecuencia.
3
4  def factorial(numero):
5      factorial = 1
6      for i in range(1, numero+1):
7          factorial *= i
8      return factorial
9
10     numero = int(input("Ingrese el número para conocer su factorial: "))
11     print("El factorial de su número es:", factorial(numero))
12
13     # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> &
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejerc
Ingrese el número para conocer su factorial: 12
El factorial de su número es: 479001600
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> |
```

7. Escribir una función que, dado un número de DNI, retorne True si el número es válido y False si no lo es. Para que un número de DNI sea válido debe tener 10 dígitos.

```
ejercicio_7.py > ...
1  #Escribir una función que, dado un número de DNI, retorne True si el número es
2  #válido y False si no lo es. Para que un número de DNI sea válido debe tener 10 dígitos.
3
4  def validarDNI(dni):
5      cantidad = 0
6      while(dni != 0):
7          dni //= 10
8          cantidad += 1
9      return cantidad==10
10
11  dni = int(input("Ingrese su dni: "))
12  print(validarDNI(dni))
13
14  # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> & C:
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejercic
Ingrese su dni: 175842
False
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> & C:
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejercic
Ingrese su dni: 1752370344
True
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> |
```

El siguiente ejercicio se encuentra en la siguiente página.

8. Contar de forma descendente hasta cero el número que ingrese por teclado 1 usuario. (Función recursiva).

```
ejercicio_8.py > ...
1 # Contar de forma descendente hasta cero el número que ingrese por teclado 1 usuario. (Función recursiva)
2
3 def paresDecendente(numero):
4     numero-=1
5     if(numero>0):
6         print(numero)
7         paresDecendente(numero)
8 numero = int(input("Ingrese el número para para conocer sus anteriores pares: "))
9 numero+=1
10 paresDecendente(numero)
11
12 # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> & C:\
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejercic
Ingrese el número para para conocer sus anteriores pares: 5
5
4
3
2
1
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> |
```

9. Contar de forma descendente hasta cero los números pares a partir de un número ingresado por el usuario. (Función recursiva)

```
ejercicio_9.py > ...
1 # Contar de forma descendente hasta cero los números pares a partir de un número ingresado por el usuario. (Función recursiva)
2
3 def paresDecendente(numero):
4     numero-=2
5     if(numero>0):
6         print(numero)
7         paresDecendente(numero)
8 numero = int(input("Ingrese el número para para conocer sus anteriores pares: "))
9 numero+=2
10 paresDecendente(numero)
11
12 # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> & C:\
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejercic
Ingrese el número para para conocer sus anteriores pares: 5
5
3
1
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> |
```

10. Calcular el factorial de un número ingresado por el usuario. (Función recursiva)

```

ejercicio_10.py > ...
1  # Calcular el factorial de un número ingresado por el usuario. (Función recursiva)
2
3  def factorial(numero):
4
5      if(numero==0 or numero==1):
6          return 1
7      elif(numero>0):
8          numero_factorial = factorial(numero-1)*numero
9      return numero_factorial
10
11 numero = int(input("Ingrese el número para conocer su factorial: "))
12 print("El factorial de ese número es:", factorial(numero))
13
14 # Richard Soria, Josué Guerra, Carlos Pérez

```

Ejecución:

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase> & C:
"d:/Documentos/2do Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios en clase/ejercic
Ingrese el número para conocer su factorial: 5
El factorial de ese número es: 120
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios en clase>

```

Prueba de escritorio:

Nombre: Guerra Josué, Pérez Carlos, Soria Richard					
Materia: Algoritmos					
Prueba de escritorio					
<pre> def calcularfactorial(numero): if numero == 0 or numero == 1: return 1 elif numero > 0: numero = numero * calcularfactorial(numero - 1) print("El factorial es: ", numero) return numero #Programa print(calcularfactorial(4)) </pre>	Paso	Estado Actual	Condición Verificada	Acción Realizada	Valor de numero
	1	Inicial	-	-	4
	2	-	False	-	-
	3	-	True	Cálculo Recursivo	4 * calcularfactorial(3)
	4	Recursivo	-	-	-
	5	-	False	-	-
	6	-	True	Cálculo Recursivo	3 * calcularfactorial(2)
	7	Recursivo	-	-	-
	8	-	False	-	-
	9	-	True	Cálculo Recursivo	2 * calcularfactorial(1)
	10	Recursivo	-	-	-
	11	Base	True	-	-
	12	Retorno	-	Retorna 1	1
	13	Recursivo	-	-	-
	14	Retorno	-	Retorna 2	2
	15	Recursivo	-	-	-
	16	Retorno	-	Retorna 6	6
	17	Final	-	Imprime y Retorna 6	6

Parte II

TAREA:

1. Solicitar al usuario que ingrese su dirección email.
Imprimir un mensaje indicando si la dirección es válida o no, valiéndose de una función para decidirlo.
Nota: El correo se considerará válido si tiene el símbolo "@".

```
Ejercicios de deber > Ejercicio_1_Correo.py > ...
1  # Solicitar al usuario que ingrese su dirección email.
2  # Imprimir un mensaje indicando si la dirección es válida o no,
3  # valiéndose de una función para decidirlo.
4  # Nota: El correo se considerará válido si tiene el símbolo "@".
5  def validarCorreo(correo):
6      if "@" in correo:
7          print("Dirección válida")
8          return exit(0)
9      else:
10         print("Ingrese una dirección de correo válida")
11
12  while(True):
13      correo = str(input("Ingrese su correo electrónico: "))
14      validarCorreo(correo)
15
16  # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios de deber/Ejercicio_1_Correo.py"
Ingrese su correo electrónico: jguerralovatogmail.com
Ingrese una dirección de correo válida
Ingrese su correo electrónico: jgerralovato@gmail.com
Dirección válida
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> |
```

Dado el código se verifica que el usuario inserte el carácter especial (@), sin eso el programa seguirá pidiendo que inserte un correo valido al momento de ingresar con todo y el carácter especial el programa terminara enseguida y en pantalla mostrara dirección valida

2. Escribir un programa que pida números al usuario, mostrar el factorial de cada uno y, al finalizar, la cantidad total de números leídos en total.

Nota: Utilice una o más funciones, si es necesario.

```
Ejercicios de deber > Ejercicio_2_Factorial.py > ...
1  # Escribir un programa que pida números al usuario, mostrar el factorial de cada uno y,
2  # al finalizar, la cantidad total de números leídos en total.
3  # Nota: Utilice una o más funciones, si es necesario.
4
5  def factorial(numero):
6      factorial = 1
7      for i in range(1, numero+1):
8          factorial *= i
9      return factorial
10 def continuar(opcion):
11     while(True):
12         if(opcion=="salir"):
13             return True
14         elif(opcion=="si"):
15             return False
16
17     contador = 0
18     while(True):
19         numero = int(input("Ingrese el número que desea conocer su factorial: "))
20
21         if (numero>0):
22             print("El factorial de", numero, "es igual a: ",factorial(numero))
23             contador+=1
24             opcion = str(input("¿Desea continuar (Escriba salir para finalizar)?\n"))
25             if (continuar(opcion)==True):
26                 break
27         elif (numero==0):
28             print("El factorial de 0 es igual a: 1")
29             contador+=1
30             opcion = str(input("¿Desea continuar (Escriba salir para finalizar)?\n"))
31             if (continuar(opcion)==True):
32                 break
33         else:
34             print("Ingrese una cantidad mayor y diferente de 0.")
35
36     print("Ha ingresado un total de", contador,"números")
37     # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> & C:/
Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios de deber/Ejerci
Ingrese el número que desea conocer su factorial: 6
El factorial de 6 es igual a: 720
¿Desea continuar (Escriba salir para finalizar)?
si
Ingrese el número que desea conocer su factorial: 5
El factorial de 5 es igual a: 120
¿Desea continuar (Escriba salir para finalizar)?
salir
Ha ingresado un total de 2 números
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> |
```

3. Solicitar al usuario el ingreso de números primos. La lectura finalizará cuando ingrese un número que no sea primo. Por cada número, mostrar la suma de sus dígitos. También solicitar al usuario un dígito e informar la cantidad de veces que aparece en el número (frecuencia). Al finalizar el programa, mostrar el factorial del mayor número ingresado.

```
Ejercicios de deber > Ejercicio_3_NPrimos.py > ...
1  # Solicitar al usuario el ingreso de números primos. La lectura finalizará cuando ingrese
2  # un número que no sea primo. Por cada número, mostrar la suma de sus dígitos. También
3  # solicitar al usuario un dígito e informar la cantidad de veces que aparece en el número
4  # (frecuencia). Al finalizar el programa, mostrar el factorial del mayor número ingresado.
5
6  def numeroPrimo(numero):
7      contador = 0
8      for i in range(1, numero+1):
9          if ((numero%i)==0):
10             contador+=1
11     return contador == 2
12
13 def calcularSuma(numero):
14     suma = 0
15     while(numero!=0):
16         digito = numero%10
17         suma += digito
18         numero = numero//10
19     return suma
20
21 def ocurrencias(numero, digito):
22     cantidad_digitos = 0
23     for i in str(numero):
24         if i == str(digito):
25             cantidad_digitos += 1
26     return cantidad_digitos
27
28 def factorial(numero):
29     factorial = 1
30     for i in range(1, numero+1):
31         factorial *= i
32     return factorial
33
34 numeros_almacenados = []
35
36 while(True):
37     numero = int(input("Ingrese el número a validar: "))
38     numeros_almacenados.append(numero)
39     if numeroPrimo(numero):
40         print("El número es primo")
41         print("La suma de los dígitos del número", numero,"es:", calcularSuma(numero))
42         digito = int(input("Ingrese el dígito que desea conocer su ocurrencia: "))
43         print("El digito se repite:",ocurrencias(numero,digito))
44     else:
45         print("El número no es primo")
46         break
47
48 numero_mayor = max(numeros_almacenados)
49 print("El factorial del mayor número es:", factorial(numero_mayor))
50
51 # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> & C:/Users/
Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios de deber/Ejercicio_3
Ingrese el número a validar: 5
El número es primo
La suma de los dígitos del número 5 es: 5
Ingrese el dígito que desea conocer su ocurrencia: 3
El digito se repite: 0
Ingrese el número a validar: 7
El número es primo
La suma de los dígitos del número 7 es: 7
Ingrese el dígito que desea conocer su ocurrencia: 2
El digito se repite: 0
Ingrese el número a validar: 12
El número no es primo
El factorial del mayor número es: 479001600
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024>
```

4. Dada una temperatura f en grados Fahrenheit, devuelva la temperatura en grados centígrados c , es decir, $c = 5(f - 32) / 9$. Implementa una función que,

```
Ejercicios de deber > Ejercicio_4_Temperatura.py > ...
1  # Dada una temperatura f en grados Fahrenheit, devuelva la temperatura en grados
2  # centígrados c, es decir, c = 5(f - 32)/9. Implementa una función.
3
4  def gradosC(grados_F):
5      grados_celcius = round((5/9)*(grados_F-32),2)
6      return grados_celcius
7
8  def continuar(opcion):
9      while(True):
10         if(opcion=="si"):
11             return True
12         elif(opcion=="no"):
13             return exit(0)
14
15  print("Conversión de grados Fahrenheit a grados Centígrados ")
16  while(True):
17      grados_F = int(input("Ingrese la cantidad de grados Farhenheit a convertir a Centígrados: "))
18      print(grados_F, "°F equivalen a", gradosC(grados_F),"°C")
19      opcion = str(input("¿Desea continuar (Escriba salir para finalizar)\n?"))
20      continuar(opcion)
21
22  # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> & C:/
Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/EJercicios de deber/Ejerci
Conversión de grados Fahrenheit a grados Centígrados
Ingrese la cantidad de grados Farhenheit a convertir a Centígrados: 125
125 °F equivalen a 51.67 °C
¿Desea continuar (Escriba salir para finalizar)?
si
Ingrese la cantidad de grados Farhenheit a convertir a Centígrados: 185
185 °F equivalen a 85.0 °C
¿Desea continuar (Escriba salir para finalizar)?
no
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> 
```

El código como tal realiza la conversión de grados Fahrenheit a grados centígrados, lo que el usuario inserta los números a convertir y el programa arroja en pantalla el resultado, también pide si el usuario desea continuar o no dependiendo de eso el programa o continuara o finalizara.

La resolución del siguiente ejercicio se encuentra en la siguiente página.

5. Solicitar al usuario el ingreso de números primos. La lectura finalizará cuando ingrese un número que no sea primo. Por cada número, mostrar la suma de sus dígitos. También solicitar al usuario un dígito e informar la cantidad de veces que aparece en el número (frecuencia). Al finalizar el programa, mostrar el factorial del mayor número ingresado.

```
Ejercicios de deber > Ejercicio_5_SumaNPrimos.py > ...
1  # Solicitar al usuario el ingreso de números primos.
2  # La lectura finalizará cuando ingrese un número que no sea primo.
3  # Por cada número, mostrar la suma de sus dígitos.
4  # También solicitar al usuario un dígito e informar la cantidad de veces que aparece en el número (frecuencia).
5  # Al finalizar el programa, mostrar el factorial del mayor número ingresado.?
6  # import math
7
8  def is_prime(n):
9      if n == 2 or n == 3: return True
10     if n < 2 or n%2 == 0: return False
11     if n < 9: return True
12     if n%3 == 0: return False
13     r = int(n**0.5)
14     f = 5
15     while f <= r:
16         if n%f == 0: return False
17         if n%(f+2) == 0: return False
18         f +=6
19     return True
20
21 def digit_frequency(num, digit):
22     count = 0
23     for i in str(num):
24         if i == digit:
25             count += 1
26     return count
27
28 def digit_sum(num):
29     sum = 0
30     for i in str(num):
31         sum += int(i)
32     return sum
33
34 def factorial(n):
35     if n == 0:
36         return 1
37     else:
38         return n * factorial(n-1)
39
40 # Lectura de números primos
41 nums = []
42 while True:
43     num = int(input("Ingrese un número primo (ingrese un número no primo para finalizar): "))
44     if is_prime(num):
45         nums.append(num)
46     else:
47         break
48
49 # Lectura de dígito y frecuencia
50 digit = input("Ingrese un dígito para ver su frecuencia en los números ingresados: ")
51 max_num = max(nums)
52
53 # Mostrar resultados
54 for num in nums:
55     print("Número:", num, "suma de dígitos:", digit_sum(num))
56     print("Frecuencia del dígito", digit, "en el número", num, ":", digit_frequency(num, digit))
57
58 print("Factorial del mayor número ingresado (", max_num, "):", factorial(max_num))
59 # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> & C:/
Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/EJercicios de deber/Ejerci
Ingrese un número primo (ingrese un número no primo para finalizar): 2
Ingrese un número primo (ingrese un número no primo para finalizar): 3
Ingrese un número primo (ingrese un número no primo para finalizar): 5
Ingrese un número primo (ingrese un número no primo para finalizar): 7
Ingrese un número primo (ingrese un número no primo para finalizar): 6
Ingrese un dígito para ver su frecuencia en los números ingresados: 4
Número: 2 suma de dígitos: 2
Frecuencia del dígito 4 en el número 2 : 0
Número: 3 suma de dígitos: 3
Frecuencia del dígito 4 en el número 3 : 0
Número: 5 suma de dígitos: 5
Frecuencia del dígito 4 en el número 5 : 0
Número: 7 suma de dígitos: 7
Frecuencia del dígito 4 en el número 7 : 0
Factorial del mayor número ingresado ( 7 ): 5040
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024>

```

6. Implementa una función llamada `laboral` que pregunte al usuario cuál es su ocupación e imprima “Ok, tu trabajo es”.

```

Ejercicios de deber > Ejercicio_6_Ocupación.py > ...
1  # Implementa una función llamada laboral que pregunte al usuario cuál es su ocupación e imprima "Ok, tu trabajo es ".
2
3  def laboral ():
4      ocupacion = input("¿Cuál es tu ocupación? \n")
5      print ("Ok, tu trabajo es",ocupacion,)
6
7  print ("---Bienvenido---")
8  laboral()
9  # Richard Soria, Josué Guerra, Carlos Pérez

```

Ejecución:

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> & C:/
Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/EJercicios de deber/Ejerci
---Bienvenido---
¿Cuál es tu ocupación?
Desarrollador de softwre
Ok, tu trabajo es Desarrollador de softwre
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024>

```

El código solo usa una función en la cual pide al usuario insertar su ocupación y este retorna un texto más junto con la ocupación insertada.

EL SIGUIENTE EJERCICIO SE ENCUENTRA EN LA SIGUIENTE PÁGINA

7. Implementa una función llamada `area_rectangulo` (`base`, `altura`) que devuelva el área del rectángulo a partir de una base y una altura. Calcula el área de un rectángulo de 15 de base 10 de altura.

```
Ejercicios de deber > Ejercicio_7AreaRect.py > ...
1  # Implementa una función llamada area_rectangulo(base, altura) que devuelva el área del rectángulo a partir de una base y una altura.
2  # Calcula el área de un rectángulo de 15 de base 10 de altura.
3
4  def area_rectangulo (base, altura):
5      area = base * altura
6      return area
7
8  print ("---Calculadora area de un rectangulo---")
9  base = float (input ("Ingrese la base: "))
10 altura = float (input ("Ingrese la altura: "))
11
12 print ("El area del rectangulo es",area_rectangulo (base,altura))
13 # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> & C:/Users/Josue/AppData/Local/Microsoft/WindowsApps/python3.11/python.exe D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios de deber\Ejercicio_7AreaRect.py
---Calculadora area de un rectangulo---
Ingrese la base: 16
Ingrese la altura: 8
El area del rectangulo es 128.0
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> |
```

8. Construya una función que devuelva el área y la longitud de una circunferencia de radio r que se introducirá como parámetro. Si no se especifica ningún parámetro se entenderá que el radio es la unidad.

```
Ejercicios de deber > Ejercicio_8_Circunferencia.py > ...
1  # Construya una función que devuelva el área y la longitud de una circunferencia de radio r que se introducirá como parámetro.
2  # Si no se especifica ningún parámetro se entenderá que el radio es la unidad.
3
4  import math
5  def area_y_longitud (radio):
6      area = math.pi * radio * 2
7      longitud = math.pi * (radio**2)
8      print(" El area es",round(area,2))
9      print(" La longitud es ",round(longitud,2))
10
11 radio = float (input ("Ingrese el radio del circulo: "))
12
13 if radio == 0:
14     radio = 1
15     area_y_longitud (radio)
16 else:
17     area_y_longitud (radio)
18 # Richard Soria, Josué Guerra, Carlos Pérez

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> & C:/Users/Josue/AppData/Local/Microsoft/WindowsApps/python3.11/python.exe D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024\Ejercicios de deber\Ejercicio_8_Circunferencia.py
Ingrese el radio del circulo: 12
El area es 75.4
La longitud es 452.39
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> |
```

9. Desarrollar un programa que permita al usuario obtener un identificador para cada uno de los socios de un club la “La Unión”. Para eso ingresará el nombre completo y número de DNI de cada socio, indicando que finalizará el procesamiento mediante el ingreso de un nombre vacío.

- Precondición: el formato del nombre de los socios será: nombre apellido. Podría ingresarse más de un nombre, en cuyo caso será: nombre1 nombre2 apellido. Si un socio tuviera más de un apellido, el usuario sólo ingresará uno.
- Se debe validar que el número de DNI tenga 10 dígitos. En caso contrario, el programa debe dejar al usuario en un bucle hasta que ingrese un DNI correcto.
- Por cada socio se debe imprimir su identificador único, el cual estará formado por: el primer nombre, la cantidad de letras del apellido y los primeros 3 dígitos de su DNI. Ejemplo:

Nombre: Alba María Linares

DNI: 25834910

Alba7258

```
Ejercicios de deber > Ejercicio_9_usuario.py > ...
1 # Desarrollar un programa que permita al usuario obtener un identificador
2 #de la "La Unión". Para eso ingresará el nombre completo y número de DNI de cada
3 #socio, indicando que finalizará el procesamiento mediante el ingreso de un nombre vacío.
4
5 def get_socio_id():
6     nombre1 = input("Ingrese el nombre del socio (deje en blanco para finalizar): ")
7     while nombre1 != "":
8         if len(nombre1.split()) < 2:
9             print("Formato de nombre inválido. Por favor, ingrese el nombre en el formato: nombre1 nombre2 apellido.")
10            nombre1 = input("Ingrese el nombre del socio (deje en blanco para finalizar): ")
11            continue
12            dni = input("Ingrese el DNI del socio (10 dígitos): ")
13            while len(dni) != 10:
14                print("Formato de DNI inválido. Por favor, ingrese un DNI válido (10 dígitos).")
15                dni = input("Ingrese el DNI del socio (10 dígitos): ")
16            nombre2 = nombre1.split()[0]
17            apellido = nombre1.split()[-1]
18            apellido_len = len(apellido)
19            socio_id = nombre2 + str(apellido_len) + dni[:3]
20            print("ID de socio:", socio_id)
21            nombre1 = input("Ingrese el nombre del socio (deje en blanco para finalizar): ")
22
23 get_socio_id()
24
25 # Richard Soria, Josué Guerra, Carlos Pérez
```

Ejecución:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Semestre/Algoritmos y estructuras de datos/Laboratorios/Lab3 19-01-2024/Ejercicios de deber/Ejercicio_9_usuario.py"
Ingrese el nombre del socio (deje en blanco para finalizar): Josue Eduard Lovato
Ingrese el DNI del socio (10 dígitos): 1752370344
ID de socio: Josue6175
Ingrese el nombre del socio (deje en blanco para finalizar): Richard Mauricio Soria
Ingrese el DNI del socio (10 dígitos): 1752864982
ID de socio: Richard5175
Ingrese el nombre del socio (deje en blanco para finalizar): Perez David Carlos
Ingrese el DNI del socio (10 dígitos): 1758694273
ID de socio: Perez6175
Ingrese el nombre del socio (deje en blanco para finalizar):
PS D:\Documentos\2do Semestre\Algoritmos y estructuras de datos\Laboratorios\Lab3 19-01-2024> █
```


10. Calcular la serie de Fibonacci (Función recursiva)

```
Ejercicios de deber > Ejercicio_10_Fibonacci.py > ...
1  # Calcular la serie de Fibonacci (Función recursiva
2
3  def fibonacci(numero):
4      if numero <= 0:
5          return 0
6      elif numero == 1:
7          return 1
8      else:
9          return fibonacci(numero - 1) + fibonacci(numero - 2)
10
11 numero = int (input ("Ingrese el numero maximo de repeticiones: "))
12 for i in range(numero):
13     resultado = fibonacci(i)
14     print(resultado, end=" ")
15     # Richard Soria, Josué Guerra, Carlos Pérez
```

ENLACES

Enlace GitHub:

<https://github.com/JosueGuerra2023B/Estructuras-Datos2023B/tree/master/Laboratorios/Lab3%2019-01-2024>

CONCLUSIONES

En resumen, las funciones y la recursividad en Python tienen como objetivo proporcionar soluciones sofisticadas y efectivas para abordar problemas que implican repeticiones o cálculos complejos. La recursividad, al permitir que una función se llame a sí misma, busca resolver problemas mediante la subdivisión en subproblemas similares al original, con el propósito de reemplazar estructuras repetitivas por una solución más elegante y eficiente. No obstante, es de vital importancia aplicar la recursividad con cuidado, definiendo casos base, gestionando la asignación de memoria y evitando bucles infinitos.

RECOMENDACIONES

Es crucial abordar el uso de la recursividad en Python con un enfoque analítico y preciso, estableciendo claramente los casos base para cada problema. Además, se debe manejar con cuidado la asignación de memoria y realizar pruebas exhaustivas para prevenir la ocurrencia de bucles infinitos. La recursividad, a pesar de ser una herramienta poderosa, debe utilizarse con prudencia, considerando alternativas no recursivas cuando sea factible.