

Instrucciones para el Laboratorio

Para cada ejercicio del laboratorio, se requiere seguir estos pasos:

1. **Desarrollo del Pseudocódigo:** Antes de comenzar a programar, elabora un pseudocódigo para cada solución. El pseudocódigo debe detallar la lógica y los pasos necesarios para resolver el problema de manera clara y estructurada. Esto te ayudará a comprender mejor la solución antes de codificarla.
2. **Implementación del Programa:** Una vez que tengas el pseudocódigo listo y revisado, procede a programar la solución en Python. Asegúrate de que tu código siga la lógica que has definido en el pseudocódigo y cumpla con los requisitos especificados en cada ejercicio.

Recuerda que una buena planificación a través del pseudocódigo facilitará el proceso de programación y te permitirá identificar posibles errores antes de implementar el código.

Ejercicio: Transformación de Palabras

Dada la lista de palabras:

```
words = ['play', 'filling', 'bar', 'theatre', 'easygoing', 'date', 'lead', 'that', 'story', 'island']
```

1. **Imprimir Letras Mayúsculas:** Escribe un programa que imprima todas las letras de cada palabra en la lista en mayúsculas, separando cada letra por una coma.

```
# tu pseudocodigo aquí

# 1. Crear una lista de palabras.
# 2. Para cada palabra en la lista:
#     a. Crear una cadena vacía para guardar las letras en mayúsculas.
#     b. Para cada letra de la palabra:
#         i. Cambiar la letra a mayúscula.
#         ii. Agregar la letra mayúscula a la cadena.
#         iii. Si no es la última letra, agregar una coma después.
#     c. Imprimir la cadena con todas las letras mayúsculas separadas por comas.

# tu codigo aquí

for word in words:
    uppercase_letters = ""
    for index in range(len(word)):
        uppercase_letters += word[index].upper()
        if index < len(word) - 1:
            uppercase_letters += ", "
    print(uppercase_letters)
```

```
P, L, A, Y
F, I, L, L, I, N, G
B, A, R
T, H, E, A, T, R, E
E, A, S, Y, G, O, I, N, G
D, A, T, E
L, E, A, D
T, H, A, T
S, T, O, R, Y
I, S, L, A, N, D
```

1. **Crear una Nueva Lista:** Genera una nueva lista que contenga solo las palabras que tengan 5 o más letras. Luego, imprime esta nueva lista.

```
# tu pseudocodigo aquí
```

```
# 1. Crear una lista de palabras.
# 2. Crear una lista vacía para guardar las palabras que cumplan la
condición.
# 3. Para cada palabra en la lista original:
#     a. Si la palabra tiene 5 o más letras:
#         i. Agregar la palabra a la nueva lista.
# 4. Mostrar la nueva lista.
```

```
# tu codigo aquí
```

```
new_list = []

for word in words:
    if len(word) >= 5:
        new_list.append(word)

print(new_list)

['filling', 'theatre', 'easygoing', 'story', 'island']
```

1. **Encontrar Palabras que Comienzan con 't':** Imprime la primera palabra en la lista que comience con la letra 't'. Si no hay ninguna, imprime un mensaje indicando que no se encontró ninguna palabra que empiece con 't'.

```
# tu pseudocodigo aquí
```

```
# 1. Crear una lista de palabras.
# 2. Crear una variable para saber si se encontró una palabra que
cumpla la condición (encontrado o no).
# 3. Para cada palabra en la lista:
#     a. Si la palabra empieza con 't':
#         i. Imprimir la palabra y cambiar la variable a encontrado.
#         ii. Salir del bucle.
# 4. Si no se encontró ninguna palabra que empiece con 't', imprimir
un mensaje que diga que no se encontró ninguna.
```

```
# tu código aquí

found = False

for word in words:
    if word.startswith('t'):
        print(word)
        found = True
        break

if not found:
    print("No se encontró ninguna palabra que empiece con 't'.")

theatre
```

1. **Contar las Vocales:** Escribe un programa que cuente e imprima el número total de vocales en todas las palabras de la lista.

```
# tu pseudocódigo aquí

# 1. Definir la lista de palabras.
# 2. Inicializar una variable para contar las vocales (contador).
# 3. Para cada palabra en la lista:
#     a. Para cada letra en la palabra:
#         i. Si la letra es una vocal (a, e, i, o, u):
#             - Incrementar el contador.
# 4. Imprimir el total de vocales contadas.

# tu código aquí

vowel_count = 0
vowels = 'aeiou'

for word in words:
    for letter in word.lower():
        if letter in vowels:
            vowel_count += 1

print("Número total de vocales:", vowel_count)

Número total de vocales: 19
```

1. **Palabras que Contienen la Letra 'a':** Crea una nueva lista que contenga solo las palabras que tengan la letra 'a'. Imprime esta nueva lista.

```
# tu pseudocódigo aquí

# 1. Crear una lista de palabras.
# 2. Crear una lista vacía para guardar las palabras que tienen la letra 'a'.
# 3. Para cada palabra en la lista:
```

```

# a. Si la palabra contiene la letra 'a':
# i. Agregar la palabra a la nueva lista.
# 4. Imprimir la nueva lista con las palabras que contienen la letra
'a'.

# tu codigo aquí

words_with_a = []

for word in words:
    if 'a' in word:
        words_with_a.append(word)

print(words_with_a)

['play', 'bar', 'theatre', 'easygoing', 'date', 'lead', 'that',
'island']

```

1. **Palabras en Orden Inverso:** Imprime la lista de palabras en orden inverso, de la última a la primera.

```

# tu pseudocodigo aquí

# 1. Crear una lista de palabras.
# 2. Para cada palabra en la lista, comenzando desde la última hasta
la primera:
# a. Imprimir la palabra.

# tu codigo aquí

for i in range(len(words) - 1, -1, -1):
    print(words[i])

island
story
that
lead
date
easygoing
theatre
bar
filling
play

```

Numeros Primos

1. **Lista de Números Primos:** Crea una lista que contenga los números primos entre 1 y 20. Imprime la lista.

```

# tu pseudocodigo aquí

```

```

# 1. Crear una lista vacía para guardar los números primos.
# 2. Para cada número desde 2 hasta 20:
#     a. Inicializar una variable para saber si el número es primo
#        (es_primo = True).
#     b. Para cada número desde 2 hasta la raíz cuadrada del número
#        actual:
#         i. Si el número actual se puede dividir sin dejar residuo:
#            - Cambiar es_primo a False y salir del bucle.
#     c. Si es_primo sigue siendo True, agregar el número a la lista de
#        primos.
# 3. Imprimir la lista de números primos encontrados.

# tu codigo aquí

primes = []

for num in range(2, 21):
    is_prime = True
    for i in range(2, int(num**0.5) + 1):
        if num % i == 0:
            is_prime = False
            break
    if is_prime:
        primes.append(num)

print(primes)

[2, 3, 5, 7, 11, 13, 17, 19]

```

1. **Contar Números Primos:** Escribe un programa que cuente cuántos números primos hay entre 1 y 30. Imprime el total.

```

# tu pseudocodigo aquí

# 1. Inicializar un contador en 0 para los números primos.
# 2. Para cada número desde 2 hasta 30:
#     a. Inicializar una variable para saber si el número es primo
#        (es_primo = True).
#     b. Para cada número desde 2 hasta la raíz cuadrada del número
#        actual:
#         i. Si el número actual se puede dividir sin dejar residuo:
#            - Cambiar es_primo a False y salir del bucle.
#     c. Si es_primo sigue siendo True, aumentar el contador en 1.
# 3. Imprimir el contador de números primos encontrados.

# tu codigo aquí

prime_count = 0

for num in range(2, 31):
    is_prime = True

```

```

for i in range(2, int(num**0.5) + 1):
    if num % i == 0:
        is_prime = False
        break
if is_prime:
    prime_count += 1

print("Total de números primos entre 1 y 30:", prime_count)

```

Total de números primos entre 1 y 30: 10

1. **Suma de Números Primos:** Crea un programa que calcule la suma de los números primos menores de 50 y la imprima.

```

# tu pseudocodigo aquí

# 1. Inicializar una variable para la suma en 0.
# 2. Para cada número desde 2 hasta 49:
#     a. Inicializar una variable para saber si el número es primo
#        (es_primo = True).
#     b. Para cada número desde 2 hasta la raíz cuadrada del número
#        actual:
#         i. Si el número actual se puede dividir sin dejar residuo:
#            - Cambiar es_primo a False y salir del bucle.
#     c. Si es_primo sigue siendo True, añadir el número a la suma
#        total.
# 3. Imprimir la suma total de los números primos encontrados.

# tu codigo aquí

total_sum = 0

for num in range(2, 50):
    is_prime = True
    for i in range(2, int(num**0.5) + 1):
        if num % i == 0:
            is_prime = False
            break
    if is_prime:
        total_sum += num

print("Suma de los números primos menores de 50:", total_sum)

```

Suma de los números primos menores de 50: 328

1. **Verificar si un Número es Primo:** Escribe un programa que pida al usuario un número y que determine si es primo. Imprime un mensaje indicando si el número es primo o no.

```

# tu pseudocodigo aquí

# 1. Pedir al usuario que ingrese un número.

```

```

# 2. Inicializar una variable para saber si el número es primo
(es_primo = True).
# 3. Si el número es menor que 2:
#     a. Cambiar es_primo a False.
# 4. Para cada número desde 2 hasta la raíz cuadrada del número
ingresado:
#     a. Si el número ingresado se puede dividir sin dejar residuo:
#         - Cambiar es_primo a False y salir del bucle.
# 5. Si es_primo sigue siendo True, imprimir que el número es primo;
de lo contrario, imprimir que no es primo.

# tu código aquí

num = int(input("Ingresa un número: "))

is_prime = True

if num < 2:
    is_prime = False

for i in range(2, int(num**0.5) + 1):
    if num % i == 0:
        is_prime = False
        break

if is_prime:
    print(f"{num} es un número primo.")
else:
    print(f"{num} no es un número primo.")

Ingresa un número: 277

277 es un número primo.

```

1. **Primos Menores a un Número:** Crea un programa que pida al usuario un número y que imprima todos los números primos menores que ese número.

```

# tu pseudocódigo aquí

# 1. Pedir al usuario que ingrese un número.
# 2. Crear una lista vacía para guardar los números primos.
# 3. Para cada número desde 2 hasta el número ingresado:
#     a. Inicializar una variable para saber si el número es primo
(es_primo = True).
#     b. Para cada número desde 2 hasta la raíz cuadrada del número
actual:
#         i. Si el número actual se puede dividir sin dejar residuo:
#             - Cambiar es_primo a False y salir del bucle.
#     c. Si es_primo sigue siendo True, agregar el número a la lista de
primos.
# 4. Imprimir la lista de números primos encontrados.

```

```
# tu codigo aquí

num = int(input("Ingresa un número: "))

primes = []

for n in range(2, num):
    is_prime = True
    for i in range(2, int(n**0.5) + 1):
        if n % i == 0:
            is_prime = False
            break
    if is_prime:
        primes.append(n)

print(f"Números primos menores que {num}: {primes}")
```

Ingresa un número: 20

Números primos menores que 20: [2, 3, 5, 7, 11, 13, 17, 19]

1. **Lista de Primos en un Rango:** Crea un programa que genere e imprima los números primos entre 10 y 50.

```
# tu pseudocodigo aquí

# 1. Crear una lista vacía para guardar los números primos.
# 2. Para cada número desde 10 hasta 50:
#     a. Inicializar una variable para saber si el número es primo
#        (es_primo = True).
#     b. Para cada número desde 2 hasta la raíz cuadrada del número
#        actual:
#         i. Si el número actual se puede dividir sin dejar residuo:
#            - Cambiar es_primo a False y salir del bucle.
#     c. Si es_primo sigue siendo True, agregar el número a la lista de
#        primos.
# 3. Imprimir la lista de números primos encontrados.
```

```
# tu codigo aquí

primes = []

for num in range(10, 51):
    is_prime = True
    for i in range(2, int(num**0.5) + 1):
        if num % i == 0:
            is_prime = False
            break
    if is_prime:
        primes.append(num)
```



```
print("Números primos entre 10 y 50:", primes)
```

```
Números primos entre 10 y 50: [11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47]
```

People

Dada la siguiente lista de personas:

```
people = [  
    {  
        "name": "Juan",  
        "age": 34,  
        "n_kids": 2,  
        "city": "Madrid"  
    },  
    {  
        "name": "Pepe",  
        "age": 27,  
        "n_kids": 0,  
        "city": "Barcelona"  
    },  
    {  
        "name": "Sonia",  
        "age": 41,  
        "n_kids": 1,  
        "city": "Valencia"  
    },  
    {  
        "name": "Lucía",  
        "age": 22,  
        "n_kids": 2,  
        "city": "Sevilla"  
    },  
    {  
        "name": "Leo",  
        "age": 55,  
        "n_kids": 5,  
        "city": "Bilbao"  
    },  
    {  
        "name": "Marta",  
        "age": 30,  
        "n_kids": 3,  
        "city": "Zaragoza"  
    },  
    {  
        "name": "Carlos",
```

```

        "age": 45,
        "n_kids": 0,
        "city": "Granada"
    }
]

```

1. **Contar Personas:** Escribe un programa que cuente cuántas personas hay en la lista `people`. Imprime el total.

```

# tu pseudocodigo aquí

# 1. Crear una lista de personas.
# 2. Contar cuántas personas hay en la lista.
# 3. Imprimir el total de personas.

# tu codigo aquí

total_people = len(people)

print("Total de personas en la lista:", total_people)

Total de personas en la lista: 7

```

1. **Contar Personas con Hijos:** Crea un programa que cuente cuántas personas tienen al menos un hijo. Imprime el número de estas personas.

```

# tu pseudocodigo aquí

# 1. Crear una lista de personas.
# 2. Inicializar un contador en 0 para las personas con hijos.
# 3. Para cada persona en la lista:
#     a. Si la persona tiene más de 0 hijos:
#         i. Aumentar el contador en 1.
# 4. Imprimir el número de personas que tienen al menos un hijo.

# tu codigo aquí

count_with_kids = 0

for person in people:
    if person["n_kids"] > 0:
        count_with_kids += 1

print("Número de personas con al menos un hijo:", count_with_kids)

Número de personas con al menos un hijo: 5

```

1. **Contar Hijos Totales:** Escribe un programa que calcule cuántos hijos hay en total entre todas las personas de la lista. Imprime el total.

```

# tu pseudocodigo aquí

# 1. Crear una lista de personas.
# 2. Inicializar un contador en 0 para el total de hijos.
# 3. Para cada persona en la lista:
#     a. Sumar el número de hijos (n_kids) al contador total.
# 4. Imprimir el total de hijos contados.

# tu codigo aquí

total_kids = 0

for person in people:
    total_kids += person["n_kids"]

print("Total de hijos entre todas las personas:", total_kids)

Total de hijos entre todas las personas: 13

```

1. **Promedio de Edad:** Calcula y muestra la edad promedio de las personas en la lista.

```

# tu pseudocodigo aquí

# 1. Crear una lista de personas.
# 2. Inicializar un contador en 0 para la suma de edades y otro para
    contar el número de personas.
# 3. Para cada persona en la lista:
#     a. Sumar la edad de la persona al contador de suma de edades.
#     b. Aumentar el contador de personas en 1.
# 4. Calcular la edad promedio dividiendo la suma de edades por el
    contador de personas.
# 5. Imprimir la edad promedio.

# tu codigo aquí

total_age = 0
count_people = len(people)

for person in people:
    total_age += person["age"]

average_age = total_age / count_people

print("La edad promedio de las personas en la lista es:", average_age)

La edad promedio de las personas en la lista es: 36.285714285714285

```

1. **Lista de Personas que Terminan en 'a':** Genera una lista con los nombres de las personas cuyo nombre termina en "a". Imprime esta lista.

```

# tu pseudocodigo aquí

```

```

# 1. Crear una lista de personas.
# 2. Inicializar una lista vacía para guardar los nombres que terminan
    en 'a'.
# 3. Para cada persona en la lista:
#     a. Si el nombre de la persona termina en 'a':
#         i. Agregar el nombre a la lista de nombres.
# 4. Imprimir la lista de nombres que terminan en 'a'.

# tu código aquí

names_ending_with_a = []

for person in people:
    if person["name"].endswith('a'):
        names_ending_with_a.append(person["name"])

print("Nombres de personas que terminan en 'a':", names_ending_with_a)

Nombres de personas que terminan en 'a': ['Sonia', 'Lucía', 'Marta']

```

1. **Actualizar Información por Año:** En un año, las personas cuyos nombres terminan en "a" tendrán un niño extra. Crea una nueva lista de diccionarios con la información actualizada de las personas (edad incrementada en 1 y `n_kids` aumentado en 1 para las que terminan en "a"). Imprime esta nueva lista.

```

# tu pseudocódigo aquí

# 1. Definir la lista de personas.
# 2. Inicializar una nueva lista vacía para almacenar la información
    actualizada.
# 3. Para cada persona en la lista:
#     a. Incrementar la edad en 1.
#     b. Si el nombre de la persona termina en 'a':
#         i. Aumentar n_kids en 1.
#     c. Añadir la información actualizada a la nueva lista.
# 4. Imprimir la nueva lista con la información actualizada.

# tu código aquí

updated_people = []

for person in people:
    updated_person = person.copy() # Copiar la información original
    updated_person["age"] += 1 # Incrementar la edad
    if updated_person["name"].endswith('a'):
        updated_person["n_kids"] += 1 # Aumentar n_kids si termina en
'a'
    updated_people.append(updated_person) # Añadir a la nueva lista

```

```
print("Información actualizada de las personas:")  
print(updated_people)
```

```
Información actualizada de las personas:  
[{'name': 'Juan', 'age': 35, 'n_kids': 2, 'city': 'Madrid'}, {'name':  
'Pepe', 'age': 28, 'n_kids': 0, 'city': 'Barcelona'}, {'name':  
'Sonia', 'age': 42, 'n_kids': 2, 'city': 'Valencia'}, {'name':  
'Lucía', 'age': 23, 'n_kids': 3, 'city': 'Sevilla'}, {'name': 'Leo',  
'age': 56, 'n_kids': 5, 'city': 'Bilbao'}, {'name': 'Marta', 'age':  
31, 'n_kids': 4, 'city': 'Zaragoza'}, {'name': 'Carlos', 'age': 46,  
'n_kids': 0, 'city': 'Granada'}]
```