

# Universidad Americana



## **Algoritmos y estructuras de datos**

**Integrante:**

- **Gabriel Antonio Rojas Uriarte**

**Marzo de 2025**

## Ejercicio 1

```
class Nodo:
    def __init__(self, dato):
        self.dato = dato
        self.siguiente = None #None significa nulo

class Lista_enlazada:
    def __init__(self):
        self.inicio = None

    def agregar(self, dato):
        nuevo = Nodo(dato)
        if self.inicio is None:
            self.inicio = nuevo
        else:
            actual = self.inicio
            while actual.siguiente:
                actual = actual.siguiente
            actual.siguiente = nuevo

    def imprimir(self):
        actual = self.inicio
        if actual is None:
            print("La lista está vacía")
        else:
            print("Elementos de la lista enlazada")
            while actual:
                print(f"-> {actual.dato}")
                actual = actual.siguiente
```

```
def main():
    lista = Lista_enlazada()

    while True:
        print("Menu de opciones")
        print("1. Agregar")
        print("2. Eliminar")
        print("3. Buscar")
        print("4. Modificar")
        print("5. Imprimir")
        print("6. Salir")
        print("Elija su opción", end = " ")
        opcion = int(input())

        if opcion == 1:
            try:
                print("Ingrese un numero entero: ", end = ' ')
                dato = int(input())
                lista.agregar(dato)
                print("Dato agregado")
            except ValueError:
                print("Dato no agregado")
        elif opcion == 5:
            lista.imprimir()
        elif opcion == 6:
            print("Salida del programa")
            break
        else:
            print("Opción no valida, intentelo de nuevo.")

if __name__ == '__main__':
    main()
```

Terminal:

```
Menu de opciones
1. Agregar
2. Eliminar
3. Buscar
4. Modificar
5. Imprimir
6. Salir
Elija su opción 1
Ingrese un numero entero: 12
Dato agregado
Menu de opciones
1. Agregar
2. Eliminar
3. Buscar
4. Modificar
5. Imprimir
6. Salir
Elija su opción 1
Ingrese un numero entero: 11
Dato agregado
```

Ejercicio 2

```
class Nodo:

    def __init__(self, dato):
        self.dato = dato
        self.siguiente = None # Puntero al siguiente nodo

class ListaEnlazada:

    def __init__(self):
        self.cabeza = None # Primer nodo de la lista

    def agregar(self, caracter):
        nuevo_nodo = Nodo(caracter)
        if self.cabeza is None:
            self.cabeza = nuevo_nodo
        else:
            actual = self.cabeza
            while actual.siguiente:
                actual = actual.siguiente
            actual.siguiente = nuevo_nodo

    def cantVocales(self):
        vocales = {'a', 'e', 'i', 'o', 'u', 'A', 'E', 'I', 'O', 'U'}
        actual = self.cabeza
        contador = 0
        while actual:
            if actual.dato in vocales:
                contador += 1
            actual = actual.siguiente
        return contador

    def mostrar_lista(self):
        actual = self.cabeza
        if actual is None:
            print("La lista está vacía")
        else:
            print("Elementos de la lista enlazada:")
            while actual:
                print(f"-> {actual.dato}")
                actual = actual.siguiente
```

```

def main():
    lista = ListaEnlazada()

    while True:
        print("\nMenu de opciones")
        print("1. Agregar un carácter")
        print("2. Contar vocales")
        print("3. Mostrar la lista enlazada")
        print("4. Salir")
        print("Elija su opción:", end=" ")
        opcion = input()

        if opcion == "1":
            caracter = input("Ingrese un carácter: ")
            if len(caracter) == 1: # Solo permite agregar un carácter
                lista.agregar(caracter)
                print(f"Carácter '{caracter}' agregado")
            else:
                print("Solo se permite un carácter")
        elif opcion == "2":
            print(f"Cantidad de vocales en la lista: {lista.cantVocales()}")
        elif opcion == "3":
            lista.mostrar_lista()
        elif opcion == "4":
            print("Salida del programa")
            break
        else:
            print("Opción no válida, inténtelo de nuevo.")

if __name__ == '__main__':
    main()

```

Terminal:

```
Menu de opciones
1. Agregar un carácter
2. Contar vocales
3. Mostrar la lista enlazada
4. Salir
Elija su opción: 1
Ingrese un carácter: a
Carácter 'a' agregado

Menu de opciones
1. Agregar un carácter
2. Contar vocales
3. Mostrar la lista enlazada
4. Salir
Elija su opción: 1
Ingrese un carácter: e
Carácter 'e' agregado

Menu de opciones
1. Agregar un carácter
2. Contar vocales
3. Mostrar la lista enlazada
4. Salir
Elija su opción: 2
Cantidad de vocales en la lista: 2

Menu de opciones
1. Agregar un carácter
2. Contar vocales
3. Mostrar la lista enlazada
4. Salir
Elija su opción: 3
Elementos de la lista enlazada:
-> a
-> e

Menu de opciones
1. Agregar un carácter
2. Contar vocales
3. Mostrar la lista enlazada
4. Salir
Elija su opción: 4
Salida del programa
PS C:\Users\gabri\OneDrive\Documentos\Análisis y diseño de sistemas\Sesion5>Listas enlazadas> |
```

Ejercicio3:

```

class Nodo:
    def __init__(self, dato):
        self.dato = dato
        self.siguiente = None # Puntero al siguiente nodo

class ListaEnlazada:
    def __init__(self):
        self.cabeza = None # Primer nodo de la lista

    def agregar(self, dato):
        nuevo_nodo = Nodo(dato)
        if self.cabeza is None:
            self.cabeza = nuevo_nodo
        else:
            actual = self.cabeza
            while actual.siguiente:
                actual = actual.siguiente
            actual.siguiente = nuevo_nodo

    def ImprimeInverso(self, P):
        elementos = []
        actual = self.cabeza
        contador = 0

        # Recorrer la lista y almacenar elementos desde la posición P
        while actual:
            if contador >= P:
                elementos.append(actual.dato)
            actual = actual.siguiente
            contador += 1

        # Imprimir elementos en orden inverso
        if elementos:
            print("Lista inversa desde la posición", P, ":", elementos[::-1])
        else:
            print(f"La posición {P} está fuera de los límites de la lista.")

    def mostrar_lista(self):
        actual = self.cabeza
        if actual is None:
            print("La lista está vacía")
        else:
            print("Elementos de la lista enlazada:")
            while actual:
                print(f"-> {actual.dato}")
                actual = actual.siguiente

```



```

def main():
    lista = ListaEnlazada()

    while True:
        print("\nMenú de opciones:")
        print("1. Agregar un número a la lista")
        print("2. Imprimir la lista enlazada")
        print("3. Imprimir lista inversa desde una posición P")
        print("4. Salir del programa")

        opcion = input("Seleccione una opción: ").strip()

        if opcion == "1":
            try:
                dato = int(input("Ingrese un número entero: ").strip())
                lista.agregar(dato)
                print(f"Número {dato} agregado a la lista.")
            except ValueError:
                print("Error: Debe ingresar un número entero válido.")
        elif opcion == "2":
            lista.mostrar_lista()
        elif opcion == "3":
            try:
                P = int(input("Ingrese la posición desde donde imprimir inverso: ").strip())
                lista.ImprimeInverso(P)
            except ValueError:
                print("Error: Debe ingresar un número entero válido.")
        elif opcion == "4":
            print("Saliendo del programa...")
            break
        else:
            print("Opción no válida. Inténtelo de nuevo.")

if __name__ == '__main__':
    main()

```

Terminal:

```
Menú de opciones:  
1. Agregar un número a la lista  
2. Imprimir la lista enlazada  
3. Imprimir lista inversa desde una posición P  
4. Salir del programa  
Seleccione una opción: 1  
Ingrese un número entero: 12  
Número 12 agregado a la lista.
```

```
Menú de opciones:  
1. Agregar un número a la lista  
2. Imprimir la lista enlazada  
3. Imprimir lista inversa desde una posición P  
4. Salir del programa  
Seleccione una opción: 1  
Ingrese un número entero: 11  
Número 11 agregado a la lista.
```

```
Menú de opciones:  
1. Agregar un número a la lista  
2. Imprimir la lista enlazada  
3. Imprimir lista inversa desde una posición P  
4. Salir del programa  
Seleccione una opción: 2  
Elementos de la lista enlazada:  
-> 12  
-> 11
```

```
Menú de opciones:  
1. Agregar un número a la lista  
2. Imprimir la lista enlazada  
3. Imprimir lista inversa desde una posición P  
4. Salir del programa  
Seleccione una opción: 3  
Ingrese la posición desde donde imprimir inverso: 0  
Lista inversa desde la posición 0 : [11, 12]
```

```
Menú de opciones:  
1. Agregar un número a la lista  
2. Imprimir la lista enlazada  
3. Imprimir lista inversa desde una posición P  
4. Salir del programa  
Seleccione una opción: 4  
Saliendo del programa...  
PS C:\Users\gabri\OneDrive\Documentos\Análisis y diseño de sistemas\Sesion5>Listas enlazadas>
```