



Universidad Americana

Documentación de trabajo realizado en clase

Génesis Nicole Ramos Zambrana

Allysson Gabriela Palma Alvarado

Andres Humberto Porras

Silvia Gigdalia Ticay Lopez

26 de Mayo del 2025

Documentación de trabajo realizado en grupo en clase sobre Manejo de Pila y Cola.

Objetivo del programa

Que la cola almacene los elementos posicionados en pares estén en la cola y los posicionados en impares en la pila.

Implementamos las estructuras FIFO(colas) y LIFO(pila), vistas en clase.

Realizamos 2 archivos, uno llamado moduloalypila donde están todos los módulos y clases principales del código y el otro llamado Main donde interactuamos con el usuario.

Archivo modulocoalypila

Clase Nodo ya que fue implementado con Nodos

Constructor del Nodo para representar el elementos en las estructuras

clase pila

Donde se guardan los elementos de la pila

Con 3 Funciones principales su constructor, Apilar y recorrer los elementos de la pila

```
✓ class Pila:
    def __init__(self):
        self.tope = None # Inicio vacío, sin nodos

    def apilar(self, dato):
        # Crear nuevo nodo y ponerlo en el tope
        nuevo_nodo = Nodo(dato)
        nuevo_nodo.siguiente = self.tope
        self.tope = nuevo_nodo
        print(f"{dato} agregado a la pila.")

    def recorrer(self):
        # Imprime los elementos de la pila desde el tope hacia abajo
        actual = self.tope
        print("Pila (impares): ", end="")
        while actual is not None:
            print(actual.dato, end=" ")
            actual = actual.siguiente
        print()
```

Clase Cola

Constructor de la clase Cola, funciones principales como encolar y recorrer

- Se inicializa con:
 - Una lista vacía para los elementos de la cola.
 - Una pila (Pila impares) para guardar los elementos en posiciones impares.

Métodos:

encolar(dato)

Agrega un dato a la cola y realiza la separación de pares/impares

desencolar()

Elimina y retorna el primer elemento (si hay).

vacía()

Retorna True si la cola está vacía.

recorrer()

Imprime todos los elementos de la cola.

```
class Cola:
    def __init__(self):
        self.frente = None
        self.final = None
        self.pila_impares = Pila() # Pila para guardar impares

    def encolar(self, dato):
        if dato % 2 == 0:
            # Si es par, lo agregamos a la cola
            nuevo_nodo = Nodo(dato)
            if self.final is None:
                # Cola vacía: frente y final apuntan al nuevo nodo
                self.frente = nuevo_nodo
                self.final = nuevo_nodo
            else:
                # Encolamos al final
                self.final.siguiente = nuevo_nodo
                self.final = nuevo_nodo
            print(f"{dato} es par agregado a la cola.")
        else:
            # Si es impar, lo agregamos a la pila de impares
            self.pila_impares.apilar(dato)
            print(f"{dato} es impar enviado a la pila.")

    def recorrer(self):
        # Imprime los elementos de la cola desde el frente hacia el final
        actual = self.frente
        print("Cola (pares): ", end="")
        while actual is not None:
            print(actual.dato, end=" ")
            actual = actual.siguiente
        print()
```

Archivo Main

Archivos donde interactuamos con el usuario, orientando donde tiene que poner poner los elementos y mostrándole los elementos

Importamos nuestras clases principales donde ya tenemos todos nuestros métodos y librerías para el color más que todo para el diseño

Se crea una cola vacía:

Entrada de datos del usuario:

- Se piden números por consola.
- Si se escribe "Fin", el ciclo termina(Se le avisa al usuario).
- Si el valor no es numérico, se muestra un mensaje de error.
- Los números válidos se agregan a la cola.

Se muestra el contenido final:

- La cola, que ahora solo contiene elementos en posiciones pares.
- La pila, que contiene los elementos de posiciones impares (en orden inverso, por se

```
def main():
    cola = Cola() # Creamos una instancia de Cola

    print(Fore.BLUE + Style.BRIGHT + "Ingrese números Para terminar, escriba 'fin'.")
    while True:
        entrada = input(Fore.LIGHTBLUE_EX + Style.BRIGHT + "Número:" + Style.RESET_ALL)
        if entrada.lower() == "fin":
            break
        if not entrada.isdigit():
            print(Fore.RED + Style.BRIGHT + "Por favor, ingrese un número válido o 'fin' para salir.")
            continue
        numero = int(entrada)
        cola.encolar(numero)

    print(Fore.GREEN + Style.BRIGHT + "\nContenido de la cola (pares):")
    cola.recorrer()

    # Mostrar contenido de la pila de impares
    print(Fore.WHITE + Style.BRIGHT + "\nContenido de la pila (impares):")
    cola.pila_impares.recorrer()
```

r LIFO).

