



Universidad de Nariño

Ingeniería de Sistemas

Estructuras de Datos

Trabajo de Implementación de Pilas y Colas

Módulo **bed.lineales.pila.py**:

```
class Pila:
    """Clase que implementa el funcionamiento del TAD Pila
    """

    def __init__(self):
        """Método constructor que realiza la creación e inicialización de
        una Pila
        """
        pass

    def es_vacia(self):
        """Método que verifica si la pila se encuentra vacía

        Returns
        -----
        bool
            Retorna True si la pila es vacia. False en caso contrario
        """
        pass

    def apilar(self, nuevo_dato):
        """Método que realiza la entrada de un nuevo dato a la pila.
        Realizar la validación de Homogeneidad para cada dato ingresado
        a la pila

        Parameters
        -----
        nuevo_dato : object
            El nuevo dato a ser adicionado a la pila

        Returns
        -----
        bool
            True si nuevo_dato fue apilado. False en caso contrario
        """
        pass

    def desapilar(self):
        """Método que saca/quita el último nodo (elimina el nodo) de la pila
        y retorna su dato

        Returns
        -----
        object|None
            El dato del nodo desapilado y None cuando la pila no contenga
            nodos/datos
        """
        pass

    def cima(self):
        """Método que retorna el dato del último nodo ingresado en la pila,
        sin quitarlo de la misma

        Returns
        -----
        object|None
            El dato del último nodo ingresado y None cuando la pila no
            contenga nodos/datos
        """
        pass

    def __len__(self):
        """Método que retorna el número de nodos que contiene la pila

        Returns
        -----
        int
            Tamaño de la pila
        """
        pass

    def __str__(self):
        """Método especial encargado de retornar una cadena con los datos
        actuales que se encuentran en la pila (sin desapilarlos)

        Returns
        -----
        str
            Una cadena que muestre todos los datos que actualmente almacena
            la pila, en el siguiente formato:
            "==(c)==
              (*[dato_n]*)
              ::
              (dato₃)
              ::
              (dato₂)
              ::
              (dato₁)"
            Cuando hay un sólo dato:
            "==(c)==
              (*[dato_n]*)"
            Cuando no hay datos:
            "==(c)=="
        """
        pass
```

Módulo **bed.lineales.cola.py**:

```
class Cola:
    """Clase que implementa el funcionamiento del TAD Cola
    """

    def __init__(self):
        """Método que realiza la creación e inicialización de la Cola
        """
        pass

    def es_vacia(self):
        """Método que verifica si la cola se encuentra vacía

        Returns
        -----
        bool
            Retorna True si la cola es vacia. False en caso contrario
        """
        pass

    def encolar(self, nuevo_dato):
        """Método que adiciona un nuevo dato al final de la cola. Realizar la
        validación de Homogeneidad para cada dato ingresado a la cola

        Parameters
        -----
        nuevo_dato : object
            El nuevo dato a ser adicionado a la cola

        Returns
        -----
        bool
            True si nuevo_dato fue encolado. False en caso contrario
        """
        pass

    def desencolar(self):
        """Método que saca/quita el primer nodo (elimina el nodo) de la cola
        y retorna su dato

        Returns
        -----
        object|None
            El dato del primer nodo de la cola y None cuando la cola no
            contenga nodos/datos
        """
        pass

    def frente(self):
        """Método que retorna el dato del primer nodo de la cola, sin quitarlo
        de la misma

        Returns
        -----
        object|None
            El dato del primer nodo en la cola y None cuando la cola no
            contenga nodos/datos
        """
        pass

    def __len__(self):
        """Método que retorna del número de nodos que contiene la cola

        Returns
        -----
        int
            Tamaño de la cola
        """
        pass

    def __str__(self):
        """Método especial encargado de retornar una cadena con los datos
        actuales que se encuentran en la cola

        Returns
        -----
        str
            Una cadena que muestre todos los datos que actualmente almacena
            la cola, en el siguiente formato:
            "@|<-{dato₁}<-[dato₂]<-[dato₃]<-[dato_n]"
            Cuando hay un sólo dato:
            "@|<-{dato₁}"
            Cuando no hay datos:
            "@|"
        """
        pass
```