**Taller complejidad**

Nombre: Daniela Lozano Villalobos

Yosefh Steven Peña Rodriguez

Asignatura:

Estructura de datos

Profesor:

# Luis Carlos Moreno



Pontificia Universidad Javeriana Bogotá D. C. 2025-30

# **Estructuras Lineales**

Plan de Pruebas:

**1. Prueba de Creación de Impresoras**

* **Objetivo**: Verificar que las impresoras se creen correctamente y se mantengan en estado inactivo por defecto.
* **Acción**: Crear varias impresoras utilizando el constructor de la clase Impresora.
* **Esperado**: Las impresoras deben tener el estado "inactivo" al momento de la creación.

**Caso de prueba**:

* Entrada: Crear Impresora("Impresora A")
* Resultado esperado: Estado de "Impresora A" debe ser "inactiva".

**2. Prueba de Agregar Trabajo a la Impresora**

* **Objetivo**: Comprobar que los trabajos se agregan correctamente a la cola de trabajos de una impresora.
* **Acción**: Crear trabajos de impresión e intentar agregarlos a una impresora.
* **Esperado**: Los trabajos deben ser agregados correctamente a la cola de trabajos de la impresora y cambiar el estado de la impresora a "activa".

**Caso de prueba**:

* Entrada: Crear TrabajoImpresion("Documento A", "Usuario 1", 5) y agregarlo a la Impresora("Impresora A").
* Resultado esperado: El trabajo debe aparecer en la cola de trabajos de la impresora, y el estado de la impresora debe cambiar a "activa".

**3. Prueba de Despachar Trabajo**

* **Objetivo**: Verificar que los trabajos se despachan correctamente, es decir, que se imprimen uno a uno y el estado de la impresora cambia a "inactiva" cuando la cola de trabajos está vacía.
* **Acción**: Despachar trabajos desde la impresora y observar cómo se procesan los trabajos.
* **Esperado**: El primer trabajo debe imprimirse completamente, y el estado de la impresora debe volver a "inactivo" cuando la cola esté vacía.

**Caso de prueba**:

* Entrada: Despachar un trabajo de la Impresora("Impresora A") que tiene trabajos en cola.
* Resultado esperado: El trabajo se imprime y el estado de la impresora cambia a "inactiva" si la cola está vacía.

**4. Prueba de Eliminar Impresora**

* **Objetivo**: Verificar que las impresoras solo se puedan eliminar cuando su cola de trabajos está vacía.
* **Acción**: Intentar eliminar una impresora mientras tiene trabajos pendientes.
* **Esperado**: La impresora no debe eliminarse si la cola de trabajos no está vacía, y debe eliminarse solo cuando la cola esté vacía.

**Caso de prueba**:

* Entrada: Intentar eliminar Impresora("Impresora A") después de agregar un trabajo.
* Resultado esperado: No se eliminará la impresora si tiene trabajos en cola.

**5. Prueba de Visualización del Sistema**

* **Objetivo**: Verificar que el sistema de impresión pueda mostrar correctamente el estado de todas las impresoras.
* **Acción**: Llamar al método visualizarSistema() para ver el estado y la cola de trabajos de cada impresora.
* **Esperado**: Se debe imprimir la lista de impresoras con su nombre, estado, y el tamaño de la cola de trabajos.

**Caso de prueba**:

* Entrada: Llamar a visualizarSistema() después de agregar varias impresoras y trabajos.
* Resultado esperado: El sistema debe mostrar la información correctamente para cada impresora.

**6. Prueba de Asignación de Trabajo a Impresora Menos Cargada**

* **Objetivo**: Comprobar que el sistema asigna trabajos a la impresora con la menor cantidad de trabajos en cola.
* **Acción**: Agregar varios trabajos y verificar que los trabajos se asignen correctamente a la impresora menos cargada.
* **Esperado**: El sistema debe seleccionar la impresora con la cola de trabajos más corta para agregar un nuevo trabajo.

**Caso de prueba**:

* Entrada: Agregar tres trabajos y verificar cuál impresora recibe cada trabajo.
* Resultado esperado: El sistema asigna los trabajos a la impresora con la menor cola de trabajos.

**Conclusión:**

Este plan de pruebas cubre aspectos esenciales del sistema de impresión, desde la creación de impresoras y la asignación de trabajos hasta la gestión de la cola de trabajos y la eliminación de impresoras. Se debe realizar cada prueba para asegurar el funcionamiento correcto de cada componente y operación del sistema. Además, el uso de pruebas unitarias o integración permitirá identificar errores en el sistema y asegurarse de que todo funcione como se espera en escenarios reales.



