

UNIVERSIDAD ESTATAL A DISTANCIA
VICERRECTORÍA ACADÉMICA ESCUELA DE
CIENCIAS EXACTAS Y NATURALES CARRERA
INGENIERÍA INFORMÁTICA



CÁTEDRA INGENIERÍA DE SOFTWARE

ASIGNATURA

00825 ESTRUCTURA DE DATOS

PROYECTO 2

VALOR: 2.0

II CUATRIMESTRE 2025

Estructura de Datos- Proyecto 2

I Objetivo de aprendizaje

- Comprender la estructura y funcionamiento de los arreglos unidimensionales y multidimensionales, y su uso en la resolución de problemas.
- Identificar y aplicar las estructuras de datos lineales como pilas y colas, reconociendo sus principios (LIFO y FIFO) y aplicaciones.
- Implementar y manipular arreglos, colas y pilas en Java, tanto con estructuras propias como utilizando clases disponibles en la API de Java.

II Objetivos de la actividad académica

- Evaluar la capacidad del estudiante para declarar, inicializar y manipular arreglos en Java, resolviendo operaciones como búsqueda, inserción, recorrido y modificación.
- Comprobar la comprensión y correcta implementación de estructuras de pila y cola de forma estática (con arreglos).
- Medir la habilidad para aplicar estas estructuras en la solución de problemas prácticos, como validación de expresiones, simulación de procesos, etc.

III Temas de estudio

- Pilas y colas.

IV Lineamientos generales

- Como herramienta de desarrollo se deberá utilizar el Netbeans en su versión más reciente. Se debe descargar el último release de la página [Apache Netbeans](#).
- También se debe utilizar el Java JDK más reciente. Pueden descargarlo de la página [Oracle](#).
- El producto por entregar corresponde al proyecto de Netbeans creado, incluyendo todas las librerías necesarias para su correcta ejecución. Este proyecto se debe subir a la plataforma Moodle en un archivo comprimido (.zip) con el nombre del estudiante y nombre del proyecto, antes de la fecha y hora límite de entrega.
- El programa debe realizarse en modo gráfico (GUI), es decir, no se permite en modo consola.

- No se permite el uso de cuadros de diálogo tipo MessageBox para solicitar o para mostrar datos. Para ello se pueden utilizar cajas de texto, etiquetas o listas gráficas según sea. La única excepción es para mostrar excepciones de la aplicación o para dar un mensaje al usuario por operaciones incorrectas en el sistema resultado de validaciones. Si la operación se realiza exitosamente, no se deben mostrar mensajes utilizando estos cuadros de diálogo.
- Los datos deben persistir en memoria en todo momento hasta que se cierre la aplicación.

V Instrucciones metodológicas

- Tipo de actividad: Individual.
- Modalidad: virtual.
- Fecha límite de entrega:
- No se permite el uso de inteligencia artificial ni colaboración externa, salvo autorización expresa del docente.
- Desarrolla tus soluciones utilizando buenas prácticas de programación (claridad, indentación, comentarios donde sean necesarios).
- Asegúrate de que el código compile y funcione correctamente.

VI Estructura del trabajo

Se deberá implementar un programa para gestionar una tienda por Departamentos y sus artículos.

El programa contendrá las siguientes pantallas o funcionalidades:

Registro de Departamentos

Pantalla para incluir los departamentos de la tienda.

Cada departamento incluido se almacenará en un arreglo de objetos de la clase "Departamento", la cual funcionará como una pila (primero elemento en entrar, último en salir).

Un atributo de la clase "Departamento" será un arreglo de objetos de la clase "Articulo", el cual se describe en la siguiente funcionalidad (Registro de artículos).

El tamaño de estos arreglos se definirá con el valor de 20.

No se podrán usar colecciones o interfaces de Java como Queue, Stack, ArrayList o ninguna otra colección ya implementada, sino que se deben utilizar **arreglos de objetos**.

En esta pantalla se podrá incluir registros de departamentos con los siguientes campos:

Nombre del campo	Tipo de dato
ID	Número entero. Debe ser un consecutivo único generado por el sistema.
Nombre	Texto.

Se debe mostrar una lista con los departamentos registrados, la cual se actualiza con cada inserción.

La lista debe ser un control tipo tabla (JTable u otro similar).

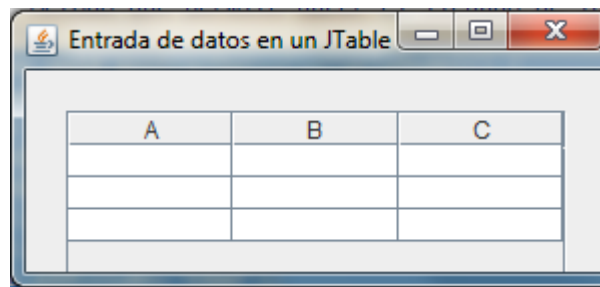


Figura 1

Registro de artículos:

En esta pantalla se debe mostrar una lista gráfica con los diferentes departamentos registrados y sus diferentes campos. Esta lista debe ser un control tipo tabla (ver Figura 1).

El usuario deberá seleccionar uno de los departamentos y a continuación ingresar los datos del artículo, el cual se va a relacionar con el departamento previamente seleccionado.

Los campos por ingresar para un artículo serán los siguientes:

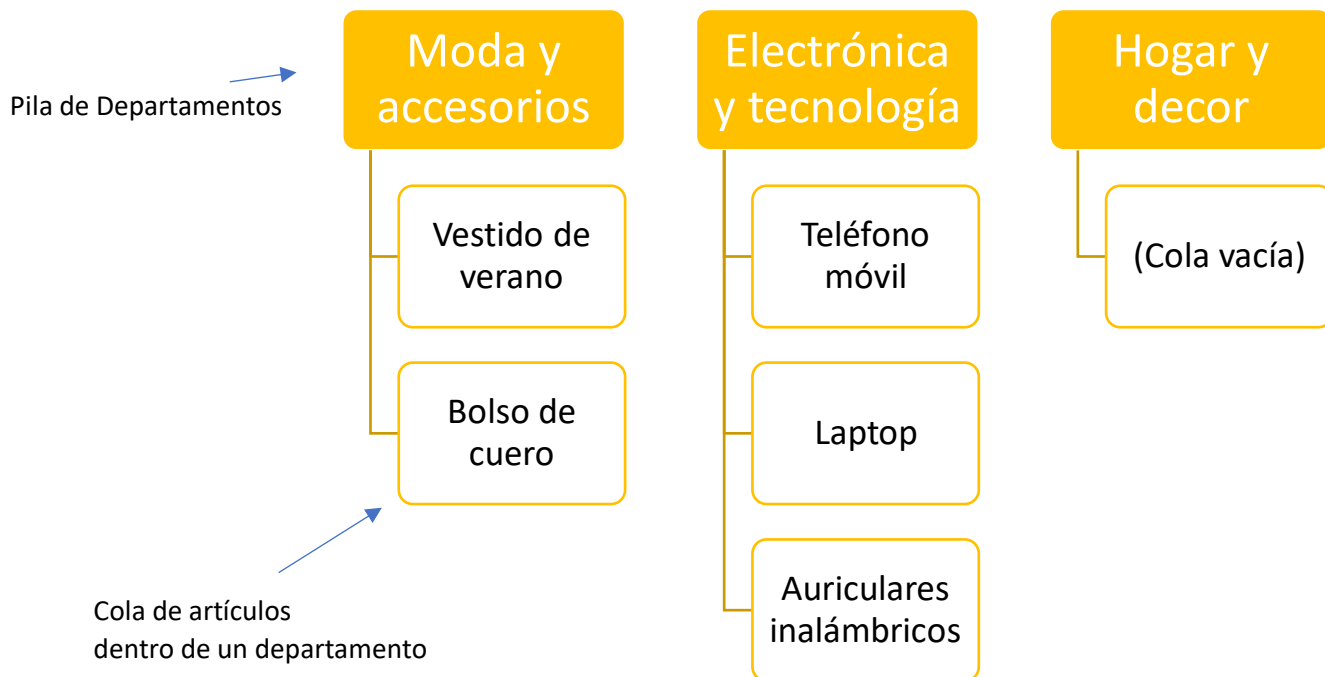
Nombre del campo	Tipo de dato
ID	Número entero. Debe ser un consecutivo generado por el sistema. Un solo consecutivo para todos los artículos de los diferentes departamentos.
Nombre	Texto.
Categoría	Se selecciona un valor de una lista (control tipo DropDownList) con las siguientes opciones: <ul style="list-style-type: none">• Ropa y accesorios• Electrónica• Hogar y muebles• Belleza y cuidado personal• Deportes y aire libre• Juguetes y juegos• Alimentos y bebidas

El almacenamiento de los artículos por departamento se realizará mediante un arreglo de objetos de la clase “Articulo”, el cual funcionará como una cola (primero en entrar, primero en salir).

No podrá utilizarse ninguna colección de Java como Queue, Stack, ArrayList ni ninguna otra colección. Debe trabajarse con **arreglos de objetos**.

Se deberá utilizar una cola **independiente** con los diferentes registros de artículos, dentro de cada atributo de la pila principal de departamentos.

Por ejemplo, la estructura de registros se definirá así:



Al inicio, la Pila de Departamentos estará vacía. Al incluir un primer departamento, éste contendrá una cola de artículos vacía, hasta que el usuario incluya los artículos de ese departamento.

Eliminación de artículos

En esta pantalla se mostrará la lista de Departamentos registrados usando un componente de tipo tabla (ver Figura 1).

El usuario podrá seleccionar uno de los Departamentos y a continuación se mostrarán los artículos registrados pertenecientes al departamento seleccionado (se deben mostrar todos los campos de los artículos) en un componente tipo tabla (Ver Figura 1).

Al hacer clic en un botón “Eliminar artículo”, se eliminará el artículo que se registró de primero en el departamento seleccionado (recordemos que se trata de una cola y por

este motivo no se debe seleccionar un elemento a eliminar). Posteriormente se refrescará la lista de artículos en la pantalla.

Traslado de artículos

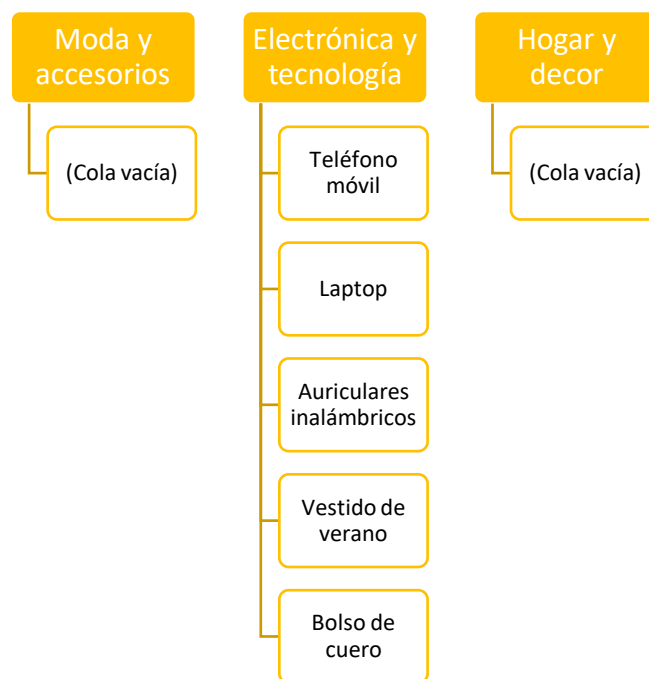
En esta pantalla el usuario debe poder trasladar una cola de artículos de un departamento a otro.

Para ello deberá seleccionar el departamento origen y el departamento destino.

Adicionalmente debe cumplir con las siguientes consideraciones:

1. La pila de departamentos debe tener al menos 2 elementos.
2. El departamento origen debe de tener al menos un artículo registrado.
3. El departamento origen y el destino no pueden ser el mismo.
4. Cada artículo por trasladar se saca de la cola origen y se incluye en la cola destino.

Por ejemplo, basándonos en la estructura de registros del ejemplo anterior, si el usuario traslada los artículos del departamento “Moda y accesorios” al departamento “Electrónica y tecnología”, el resultado final sería el siguiente:



Eliminación de departamentos

La eliminación de departamentos consiste en una acción que intentará eliminar de la pila principal el último elemento registrado (como en cualquier pila, por lo tanto, no se debe seleccionar el problema a eliminar).

La eliminación se hará efectiva únicamente si el último elemento de la pila no contiene ningún artículo asociado (es decir, su cola de artículos está vacía). En caso de existir al menos un artículo, se mostrará un mensaje indicando que no se puede eliminar el departamento debido a que tiene artículos relacionados.

Si el departamento es eliminado, se refrescará la lista de departamentos con los elementos existentes.

VII Rúbrica o escala de evaluación

NO.	INDICADORES POR EVALUAR	CUMPLIMIENTO		PUNTOS
		Cumple	No cumple	
REGISTRO DE DEPARTAMENTOS				
1.	<p>La pantalla del registro de problemas incluye los 2 campos indicados en el enunciado con sus correspondientes tipos de dato según lo especificado para cada campo.</p> <p>Cada registro incluido se almacena en un arreglo de objetos de la clase “Departamento”, el cual funciona como una pila (primer elemento en entrar, último en salir).</p> <p>No se utilizan colecciones o interfaces de Java como Queue, Stack, ArrayList o ninguna otra colección ya implementada.</p> <p>Un atributo de la clase “Departamento” es un arreglo de objetos de la clase “Artículo”.</p> <p>Nota: debe cumplir con todos los enunciados anteriores, sino no, la nota será 0.</p>			2
2.	<p>Se muestra una lista con los departamentos registrados, la cual se actualiza con cada inserción (debe incluir todos los campos del registro).</p> <p>La lista debe ser un control de tipo tabla (como en la Figura 1).</p>			2
REGISTRO DE ARTÍCULOS				
3.	<p>En esta pantalla se debe mostrar una lista gráfica con los diferentes departamentos registrados (incluyendo todos sus campos). La lista debe ser un control de tipo tabla (como en la Figura 1).</p> <p>El usuario puede seleccionar uno de los departamentos registrados.</p>			2
4.	<p>El usuario puede seleccionar uno de los departamentos y a continuación ingresar los datos del artículo, el cual se va a relacionar con el departamento previamente seleccionado.</p> <p>Los datos de ingreso incluyen los 3 campos indicados en el enunciado correspondiente a este apartado, considerando los tipos de dato señalados.</p> <p>El almacenamiento de los artículos por departamento (incluyendo los 3 campos especificados en el enunciado) se realizará mediante un arreglo de objetos de la clase “Articulo”, el cual funcionará como una cola (primero en entrar, primero en salir).</p> <p>No podrá utilizarse ninguna colección de Java como Queue, Stack, ArrayList ni ninguna otra colección.</p> <p>Tampoco se podrá implementar usando listas enlazadas con nodos.</p>			3

	Se deberá utilizar una cola independiente con los diferentes registros de artículos, dentro de cada elemento de la pila principal de departamentos. Nota: debe cumplir con todos los enunciados anteriores, sino no, la nota será 0.			
ELIMINACIÓN DE ARTÍCULOS				
5.	En esta pantalla se muestra una lista gráfica con los diferentes artículos registrados. El usuario puede seleccionar un departamento y se refresca la lista de artículos que tiene relacionados (se deben mostrar todos los campos de los artículos). Al hacer clic en un botón “Eliminar artículo”, se elimina el artículo que se registró de primero (usando una estructura de cola, por lo tanto, el usuario no selecciona el artículo a eliminar). Luego, se refresca la lista de artículos en la pantalla.			3
TRASLADO DE ARTÍCULOS				
6.	El usuario puede seleccionar el departamento origen y el destino. Se valida que la cola de departamentos tenga al menos 2 elementos. Se valida que el departamento origen no esté vacío. Cada artículo por trasladar se saca de la cola origen y se incluye en la cola destino.			4
ELIMINACIÓN DE DEPARTAMENTOS				
7.	La eliminación se realiza únicamente si el último elemento de la pila no contiene ningún artículo asociado (es decir, su cola de artículos está vacía). Luego de eliminado el departamento, se refresca la lista de departamentos. En caso de existir al menos un artículo, se muestra un mensaje indicando que no se puede eliminar el departamento debido a que tiene artículos relacionados.			2
GENERALIDADES				
8.	Utiliza modo gráfico para todas las funcionalidades sin excepción.			6
9.	No utiliza cuadros de diálogo (tipo MessageBox u otro control similar) para dar ni para solicitar información al usuario, a menos que sean excepciones del programa o que el usuario omitió algo necesario para la operación que quiere realizar.			1