

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 1

VALOR: 3.0

II CUATRIMESTRE 2025

Estructura de Datos- Proyecto 1

I Objetivos de aprendizaje

- Comprender y aplicar las API de colecciones de Java para la manipulación eficiente de estructuras de datos como listas, conjuntos, mapas y colas.
- Diseñar e implementar soluciones recursivas a problemas computacionales, identificando casos base y pasos recursivos correctamente.
- Comparar, analizar e implementar algoritmos de ordenamiento (como burbuja, selección, inserción, quicksort y mergesort), evaluando su eficiencia en términos de complejidad temporal y espacial.
- Seleccionar e integrar estructuras de datos adecuadas según el tipo de problema y los requerimientos de eficiencia.

II Objetivos de la actividad académica

- Evaluar la capacidad del estudiante para utilizar adecuadamente las colecciones de Java, seleccionando la estructura adecuada según el problema propuesto (por ejemplo, ArrayList, HashSet, HashMap, etc.).
- Comprobar que el estudiante entiende y aplica correctamente la recursividad, incluyendo la formulación de funciones recursivas y la resolución de problemas con estructuras anidadas o jerárquicas.
- Medir la habilidad del estudiante para implementar e interpretar algoritmos de ordenamiento.

III Temas de estudio

- La API de colecciones y recursividad.
- Algoritmos de ordenamiento.

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 llevar un registro de las principales celebraciones que se realizan en diferentes países.

El programa deberá contener las siguientes funcionalidades:

Registro de celebraciones:

En esta pantalla se podrá incluir los datos generales de cada celebración, con los siguientes campos:

Nombre del campo	Tipo de dato
ID de la celebración	Identificador consecutivo numérico generado por el programa (no editable).
Fecha	Campo para seleccionar una fecha. La librería utilizada para este campo debe ser incluida en el proyecto al entregarlo.
Descripción	Texto con la descripción de la celebración.
País	Texto con el nombre del país al que corresponde la celebración.

La información de las celebraciones con los campos anteriormente especificados, se guardarán en una clase “Celebracion” y cada una de estas se agregará a una única colección utilizando la clase **ArrayList** de Java.

Listado de celebraciones:

En esta pantalla se listan todas las celebraciones registradas en la colección (con todos sus campos).

En esta opción se debe utilizar una clase de tipo **Iterator** para recorrer la colección.

Los datos deben ser mostrados usando un componente tipo tabla (un JTable o similar).

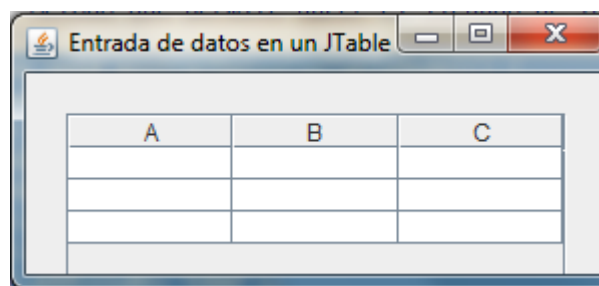


Figura 1

Búsqueda y edición de una celebración:

Para buscar una celebración determinada, el usuario debe digitar un país y hacer clic en un botón Buscar.

El programa debe buscar todas las celebraciones que pertenezcan al país digitado. La búsqueda debe seleccionar todos los registros cuyo valor en el campo País, contenga el texto digitado en la búsqueda.

Por ejemplo, si el usuario digita "ric" como texto a buscar, el programa devolverá registros existentes como "Puerto Rico" y "Costa Rica". La búsqueda no es de tipo Case Sensitive.

Luego de la búsqueda, se deben mostrar los registros encontrados (con todos sus campos) desplegándolos en un componente tipo tabla (ver Figura 1).

De los registros desplegados, el usuario podrá seleccionar un único registro para su edición. Al seleccionar un registro, se mostrarán en pantalla los campos del registro a editar con los valores actuales para poder ser editados.

La edición no será dentro de la misma tabla que contiene los registros encontrados, sino que en un bloque aparte se mostrarán cada uno de los campos a editar similar a la pantalla de registro de eventos, considerando lo especificado en la columna de tipo de dato. Por ejemplo, en el caso de la Fecha, se mostrará el control de fecha o calendario igual al de la pantalla de registro, mostrando como seleccionado el valor actual del campo fecha.

El campo ID del evento no podrá ser editado.

Habrà un botón GUARDAR para guardar los cambios.

Listado de países invertidos:

Debe utilizar **recursividad** para esta operación.

En esta opción, el programa obtendrá los diferentes países existentes en todos los registros de celebraciones.

El programa tomará los países obtenidos y los mostrará en un componente tipo tabla (ver Figura 1) pero con el texto invertido.

Por ejemplo, si uno de los países obtenidos es "Finlandia", se mostrará el texto "aidnalniF".

La inversión del texto se debe hacer con una función recursiva.

Ordenamiento de la lista de celebraciones:

En esta pantalla habrá dos tipos de ordenamiento:

1. Se listarán las celebraciones ordenadas ascendentemente por el campo País como primer criterio y el campo Fecha como segundo criterio (considerando el formato año-mes-día). Esto utilizando el algoritmo "Inserción".

Proyecto 1.
Valor relativo: 40 puntos

Fecha de entrega:
Valor: 3.0

2. Se listarán las celebraciones ordenadas descendientemente por el campo País como primer criterio y el campo Fecha como segundo criterio (considerando el formato año-mes-día). Esto utilizando el algoritmo "Merge Sort".

Para el ordenamiento no se permite utilizar el método Sort de la colección ArrayList o de ninguna otra colección.

Los registros ordenados (con todos sus campos) se mostrarán en un componente tipo tabla (ver Figura 1), según el tipo de ordenamiento indicado por el usuario.

VII Rúbrica o escala de evaluación

NO.	INDICADORES POR EVALUAR	CUMPLIMIENTO		PUNTOS
		Cumple	No cumple	
REGISTRO DE CELEBRACIONES				
1.	La pantalla del registro de celebraciones incluye los 4 campos indicados en el enunciado con sus correspondientes tipos de dato según lo especificado para cada campo. Los 4 campos se almacenan en clases llamadas “Celebración” y cada una de estas se agrega a una única colección utilizando la clase ArrayList de Java. 1 punto por cada uno.			4
LISTADO DE CELEBRACIONES				
2.	En esta pantalla se listan todas las celebraciones registradas en la colección de tipo ArrayList (con todos sus 4 campos según el enunciado en el Registro de celebraciones). Los datos se despliegan en un componente tipo tabla (como en la Figura 1 de este enunciado). Se utiliza una clase de tipo Iterator para recorrer la colección ArrayListList. Nota: debe cumplir con todos los enunciados anteriores, sino no, la nota será 0.			4
BÚSQUEDA Y EDICIÓN DE CELEBRACIONES				
3.	El programa busca todas las celebraciones que pertenezcan al país digitado por el usuario. La búsqueda debe seleccionar todos los registros cuyo valor en el campo País, contenga el texto digitado en la búsqueda. La búsqueda no es de tipo Case Sensitive. Luego de la búsqueda, se deben mostrar los registros encontrados (con todos sus campos) desplegándolos en un componente tipo tabla (como en la Figura 1 de este enunciado). Nota: debe cumplir con todos los enunciados anteriores, sino no, la nota será 0.			4
4.	De los registros desplegados en el punto anterior, el usuario podrá seleccionar un único registro para su edición. Al seleccionar un registro, se mostrarán en pantalla los campos del registro a editar con sus valores actuales.			4

	<p>La edición no será dentro de la misma tabla que contiene los registros encontrados, sino que en un bloque aparte se mostrarán cada uno de los campos a editar similar a la pantalla de registro de celebraciones, considerando lo especificado en la columna de tipo de dato. Por ejemplo, en el caso de la Fecha, se mostrará el control de fecha o calendario igual al de la pantalla de registro, mostrando como seleccionado el valor actual del campo fecha.</p> <p>El campo ID del evento no podrá ser editado.</p> <p>Habrà un botón GUARDAR para guardar los cambios.</p> <p>1 punto por cada campo que se muestre y guarde correctamente.</p>			
	LISTADO DE PAÍSES INVERTIDOS			
5.	<p>Debe utilizar recursividad para esta operación.</p> <p>En esta opción, el programa obtendrá los diferentes países existentes en todos los registros de celebraciones.</p> <p>El programa tomará los países obtenidos y los mostrará en un componente tipo tabla (como en la Figura 1 de este enunciado) pero con el texto invertido.</p> <p>Por ejemplo, si uno de los países obtenidos es "Finlandia", se mostrará el texto "aidnalniF".</p> <p>La inversión del texto se debe hacer con una función recursiva.</p>			4
	ORDENAMIENTO DE LA LISTA DE CELEBRACIONES			
7.	<p>Se listarán las celebraciones ordenadas ascendentemente por el campo País como primer criterio y el campo Fecha como segundo criterio (considerando el formato año-mes-día). Esto utilizando el algoritmo "Inserción".</p> <p>Para el ordenamiento no se permite utilizar el método Sort de la colección ArrayList o de ninguna otra colección.</p> <p>Los registros ordenados (con todos sus campos) se mostrarán en un componente tipo tabla (como en la Figura 1 de este enunciado), según el tipo de ordenamiento indicado por el usuario.</p> <p>Nota: debe cumplir con todos los enunciados anteriores, sino no, la nota será 0.</p>			4

8.	<p>Se listarán las celebraciones ordenadas descendientemente por el campo País como primer criterio y el campo Fecha como segundo criterio (considerando el formato año-mes-día). Esto utilizando el algoritmo "Merge Sort".</p> <p>Para el ordenamiento no se permite utilizar el método Sort de la colección ArrayList o de ninguna otra colección.</p> <p>Los registros ordenados (con todos sus campos) se mostrarán en un componente tipo tabla (como en la Figura 1 de este enunciado), según el tipo de ordenamiento indicado por el usuario.</p> <p>Nota: debe cumplir con todos los enunciados anteriores, sino no, la nota será 0.</p>			4
GENERALIDADES				
9.	Utiliza modo gráfico para todas las funcionalidades sin excepción.			10
16.	No utiliza cuadros de diálogo (MessageBox o similares) para dar o solicitar información al usuario, ni para indicar el resultado de una operación exitosa. Solo se usan a menos que sean excepciones del programa o que el usuario omitió algo necesario para la operación que quiere realizar.			2