ISFPP 2023 – PROGRAMACION ORIENTADA A OBJETOS

Alumnos:

Cardarilli, Martin

Cornejo, Maia

Pacheco, Melisa

INTRODUCCION

Este proyecto se enfoca en examinar los Tipos de Datos Abstractos (TAD), explorando su comportamiento, propiedades y cómo estas relaciones pueden ser empleadas para abordar desafíos de la vida cotidiana en nuestra sociedad mediante el desarrollo de software.

Análisis de las estructuras seleccionadas

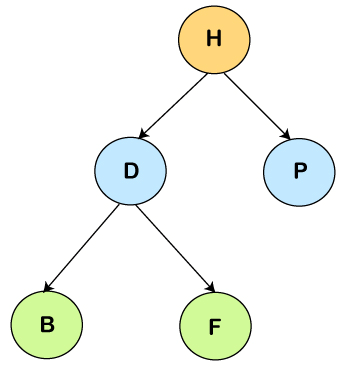
La aplicación del principio de abstracción al diseño de estructuras de datos conduce a la creación de Tipos de Datos Abstractos (TDA). Un TDA se caracteriza por un conjunto de operaciones, generalmente denominadas interfaz pública, que representa el comportamiento del TDA, mientras que la implementación interna constituye la parte privada del TDA. Nos centraremos en TDAs que exhiben comportamientos organizativos específicos.

En el sistema de transporte de colectivos, las TDAs seleccionadas son:

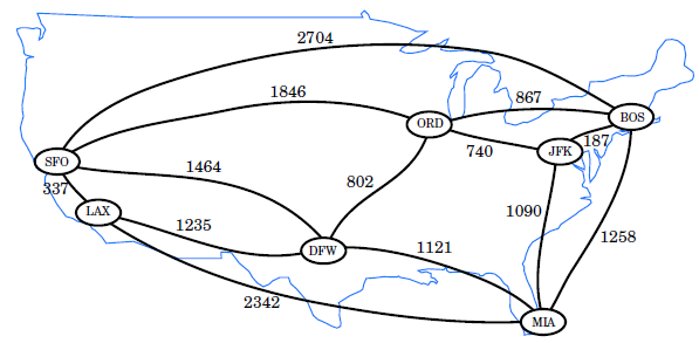
ArrayList (implementación de List)

TreeMap (implementación de Map)

AdjacencyMapGraph de JGraphT (implementación de Graph)

En el ámbito de la informática, una lista (List) es un TAD que organiza elementos en una secuencia ordenada. En este contexto, la implementación específica es un ArrayList, que es una estructura de datos que proporciona acceso rápido a los elementos mediante índices por lo que se utilizo para cargar los Tramos.

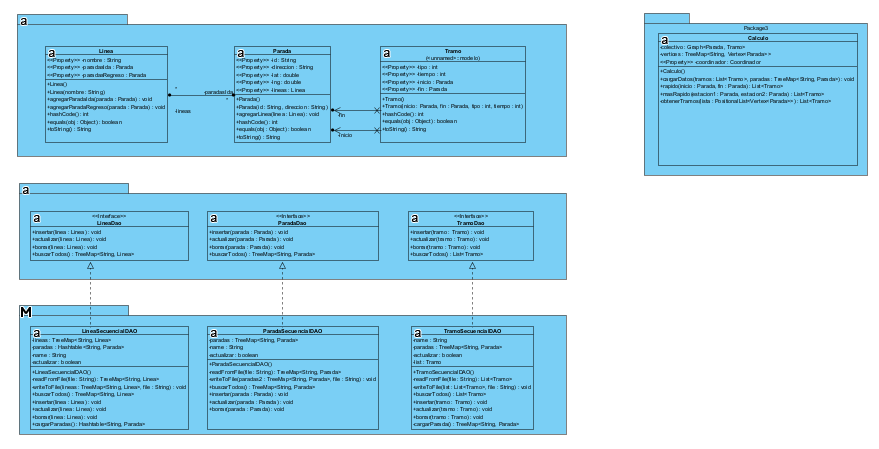
Por otro lado, un mapa (Map) es un TAD que asocia claves con valores, permitiendo la recuperación eficiente de un valor dado una clave. En nuestra implementación, se ha utilizado TreeMap, que organiza las claves de manera ordenada según su comparación natural o mediante un comparador proporcionado. Se utilizo Map para cargar las paradas y las líneas.



Además, se ha incorporado el uso de un grafo (Graph) mediante la implementación de AdjacencyMapGraph de JGraphT. Este grafo permite representar las relaciones y conexiones entre diferentes puntos en el sistema de transporte, facilitando la modelización de la red de colectivos. Cada nodo en el grafo representa una parada o ubicación, y las aristas representan las conexiones entre estas ubicaciones.

Estas estructuras son esenciales en el sistema de colectivos para organizar, gestionar y analizar datos de manera eficiente. La ArrayList se utiliza para almacenar información en secuencia, TreeMap para la asociación eficiente de claves con valores, y AdjacencyMapGraph para representar las relaciones espaciales entre ubicaciones, brindando una base sólida para la planificación y optimización de rutas en el contexto del sistema de transporte de colectivos.

Diagrama de clase - UML



Implementación

Manual de Usuario - Aplicación de Consultas de ColectivoInterfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamente

Descripción de la Aplicación:

La aplicación de consultas de transporte colectivo proporciona a los usuarios la capacidad de planificar rutas y obtener información detallada sobre el recorrido entre dos paradas de colectivo. Utiliza un sistema de consultas eficiente que tiene en cuenta la velocidad y las conexiones disponibles.

Configuración:

Líneas, Paradas y Tramos:

Para acceder a la información de las líneas de colectivo, paradas y tramos, seleccione la opción "Datos" en el menú y elija entre " Líneas," "Paradas" o "Tramos."

Realizar Consulta de Ruta:

En el menú "Aplicación," seleccione la opción "Mas rápido" para encontrar la ruta más rápida entre dos paradas.

Elija la parada de origen y destino en los menús desplegables.

Ingrese la hora de llegada al origen para obtener un recorrido considerando el tiempo actual.

Instrucciones para el Uso:

Consulta de Ruta:

Seleccione "Mas rápido" en el menú de la aplicación.

Elija las paradas de origen y destino de la lista desplegable.

Ingrese la hora de llegada al origen.

Haga clic en el botón "Mas rápido" para obtener la ruta más rápida.

Visualización del Resultado:

La información detallada del recorrido se mostrará en la parte inferior de la ventana.

Se proporciona un desglose del tiempo total, tiempo en colectivo y tiempo caminando.

Exploración del Mapa:

Seleccione "Datos" en el menú y elija "Líneas" para modificar, insertar o eliminar líneas.

El mapa muestra las estaciones de colectivo y la ruta más rápida entre dos paradas.

Salir de la Aplicación:

Para cerrar la aplicación, seleccione "Aplicación" en el menú y elija "Salir."

Notas Adicionales:

La barra de progreso muestra el avance del cálculo de la ruta más rápida.

Puede explorar el mapa arrastrándolo y usando la rueda del mouse para hacer zoom.

¡Gracias por utilizar nuestra aplicación de consultas de transporte colectivo! Siempre estamos trabajando para mejorar y brindar la mejor experiencia posible.

Errores detectados, posibles mejoras y extensiones

Errores:

No logramos hacer la modificación de Paradas.

El programa se rompe si se seleccionan dos paradas iguales.

Posibles mejoras y extensiones:

En un futuro también se podría incorporar un mapa que solamente muestre todas las paradas.

Que se pueda distinguir los tramos caminando de los tramos en colectivo.

Agregarle congestión y manejo de pasajeros.

Geolocalización en tiempo real para manejar las horas de salida y llegada.

Conclusión:

En resumen, la implementación de estructuras de datos robustas y la creación de una aplicación de consultas eficiente han sido los pilares de este proyecto. La utilización de TDAs ha demostrado ser esencial para organizar, gestionar y analizar datos de manera efectiva en el contexto del sistema de transporte de colectivos. La identificación de posibles mejoras abre la puerta a futuras iteraciones que continuarán elevando la calidad y la usabilidad de la aplicación.