

Nombre y Apellido:
Profesor:

5 de Julio de 2025
Cantidad de hojas:

Escenario

Se dispone de un conjunto de ciudades conectadas por carreteras dirigidas, con distancias conocidas en kilómetros. Además, hay un grupo de camiones, cada uno identificado por un ID, con una velocidad propia (en km/h) y una autonomía máxima (en kilómetros). Cada camión puede encontrarse inicialmente en una ciudad cualquiera.

Tu tarea es desarrollar un sistema que permita almacenar y gestionar la información de ciudades, carreteras y camiones, y que proporcione la funcionalidad para consultar rutas óptimas según el tiempo de viaje de cada camión, considerando además las restricciones de posición inicial y autonomía.

Funcionalidades a Implementar

1. Método principal de rutas

Debes implementar el siguiente método:

List<Recorrido> ruta(String ciudadOrigen, String ciudadDestino)

donde para cada camión registrado en el sistema se debe calcular la ruta más rápida (en menor tiempo total) entre la ciudad de origen y la de destino, considerando la velocidad individual de cada camión.

La clase Recorrido debe tener los siguientes campos:

- camionID: String
- tiempo: float (en horas)
- rutaRecorrida: List<Ciudad> (lista ordenada de ciudades recorridas desde el origen al destino)

2. Método extendido con posición inicial y autonomía

Debes implementar también el siguiente método:

List<Recorrido> rutaExtendida(String ciudadOrigen, String ciudadDestino)

Este método debe considerar los siguientes aspectos adicionales respecto al método ruta:

- Posición inicial propia: Cada camión puede encontrarse en una ciudad distinta al origen solicitado en la consulta. Por lo tanto, cada camión deberá calcular primero la ruta más rápida desde su ciudad actual hasta la ciudad de origen especificada, y luego continuar desde allí hasta la ciudad de destino. Si el camión ya se encuentra en la ciudad de origen, comenzará directamente la ruta hacia el destino.
- Autonomía máxima: Cada camión dispone de una autonomía máxima (en kilómetros), que limita la cantidad total de kilómetros que puede recorrer en todo el trayecto (desde su ciudad inicial, pasando por el origen de la consulta y hasta el destino final). Si la ruta necesaria supera la autonomía máxima del

Nombre y Apellido:
Profesor:

5 de Julio de 2025
Cantidad de hojas:

camión, dicho camión debe ser omitido del resultado. Se asume que los camiones no pueden recuperar autonomía hasta llegar al destino. El método debe devolver, para cada camión que pueda completar el recorrido dentro de su autonomía, un objeto Recorrido con la ruta óptima (en menor tiempo total), considerando su velocidad individual y la restricción de autonomía.

3. Programa y Test cases

La clase principal se denomina "Main", y tiene su correspondiente método "main". En éste:

1. Lee y carga los datos
2. Ejecuta las consultas especificadas e imprime por consola los resultados

Implementa el o los **Casos de Prueba** necesarios para verificar el correcto funcionamiento de los métodos implementados para resolver las funcionalidades listadas más arriba (ruta y rutaExtendida).

Notas adicionales

- Usa cualquier estructura o método visto en el curso, o derivados.
- Si no hay ruta posible para un camión, omítelo del resultado.
- Se provee el código necesario para cargar las estructuras de datos necesarias para resolver el problema.
- Solo se deben modificar y/o agregar archivos de código en el paquete "uy.edu.ucu.aed.parcial" (directorío: uy.edu.ucu.aed.parcial).