



Algoritmo y Estructura de Datos II

**“Mapa de transporte público”**  
**Grupo 12**

Integrantes:

Grupo 31:

Frias Juan Gabriel.

Grupo 61:

Fernandez Federico.

Acosta Celio Elizeo.

Grupo 21:

Gauna Gaston Leonel.

## **Tema y contexto**

En otro universo existe un planeta muy parecido a la Tierra y en él una ciudad hermosa a la cual nombraron “Corrientes”.

En plena urbanización comenzaron a funcionar los colectivos, pero al principio nadie sabía a dónde iría a parar si se subía a uno de ellos.

Para resolver este problema, se desarrolló un sistema que permite conocer qué colectivos conectan distintos barrios y cómo se relacionan entre sí.

## **Objetivo**

El objetivo del proyecto es crear un programa que permita al usuario seleccionar un barrio de origen y uno de destino, mostrando el camino más corto y qué líneas de colectivo conectan ambos puntos, también implementar una función que, en caso de que no haya un colectivo directo desde el punto A hasta el punto B, muestre qué colectivos habrá que tomar y en qué barrios para poder llegar al destino.

Además, el sistema debe permitir consultar la información general de todos los colectivos, choferes y horarios almacenados en archivos.

De esta manera, el programa busca organizar la información del transporte urbano y facilitar el acceso a ella de forma clara y estructurada.

## **Decisiones de diseño**

Estructura elegida para el mapa de barrios: Grafo ponderado

Uso: representar los barrios (nodos) y las distancias entre ellos (arcos con peso).

Motivo: un grafo es la estructura ideal cuando se necesita modelar conexiones entre distintos puntos, permitiendo analizar relaciones como cercanía o rutas posibles.

Alternativas consideradas:

Se evaluó usar una matriz simple o listas de adyacencia por separado, pero estas opciones no permitían representar tan fácilmente la relación bidireccional entre barrios ni manejar distancias.

Por eso, se eligió un grafo ponderado no dirigido, ya que refleja de manera natural el sistema de transporte urbano, donde las calles conectan barrios en ambos sentidos y las distancias son valores importantes.

Estructura elegida para la gestión de datos: Archivos binarios

Uso: almacenar la información de colectivos, choferes y horarios.

Motivo: los archivos binarios permiten guardar registros estructurados con lectura y escritura eficiente, manteniendo los datos incluso después de cerrar el programa.

Alternativas consideradas:

Se consideró usar archivos de texto, pero se descartaron porque no permiten un acceso directo a los registros ni el manejo sencillo de tipos de datos como números enteros y estructuras compuestas.

Los archivos binarios son más adecuados para proyectos de tipo “base de datos” donde se trabaja con varios campos por registro.

Estructura elegida para el listado: Corte de control

Uso: generar un listado de colectivos agrupando su información (número, choferes, horarios, ).

Motivo: la técnica de corte de control permite procesar archivos ordenados agrupando datos de forma automática sin estructuras adicionales complejas.

Alternativas consideradas:

Se pensó en recorrer todo el archivo y usar condicionales simples, pero eso resultaba menos eficiente y más confuso al agrupar datos.

El corte de control fue la opción más clara y ordenada para mostrar la información.

### **Desarrollo obtenido**

El sistema implementa las siguientes funcionalidades:

Carga de barrios y conexiones: se ingresan los barrios, las líneas de colectivo que pasan por cada uno y las distancias entre ellos.

Consulta de recorrido: el usuario puede seleccionar un barrio “A” y un barrio “B” para conocer qué colectivos los conectan.

Gestión de archivos binarios: se guarda y muestra la información de los colectivos, choferes y horarios.

Listado de colectivos: se aplica la técnica de corte de control para mostrar de forma agrupada la información de los colectivos disponibles.

Interfaz de menú principal: integra todas las funcionalidades anteriores, permitiendo navegar entre las opciones del sistema de manera sencilla.

### **Uso de IA**

En el desarrollo del grafo y el algoritmo de rutas:

Para el desarrollo de la estructura de datos del grafo ponderado, así como para la implementación del algoritmo de Dijkstra y las funciones asociadas al cálculo de rutas y líneas de transporte, se utilizó asistencia de la herramienta **ChatGPT** exclusivamente como apoyo técnico. Toda la lógica fue comprendida en profundidad, adaptada y modificada para ajustarse a las necesidades específicas del proyecto, incluyendo la carga del mapa, la representación de los arcos con pesos y colectivos, y la reconstrucción de rutas óptimas.

Cada función fue revisada, integrada al proyecto y probada mediante casos de prueba personalizados para validar su funcionamiento correcto. La IA se empleó como guía para resolver dudas puntuales de sintaxis, estructura y organización del código, pero la implementación final fue ajustada manualmente y verificada de forma exhaustiva para garantizar su coherencia con los requisitos del trabajo integrador.