# Practica 1

## Inicialización y estructuras principales

Pantalla de computadora con letras

El contenido generado por IA puede ser incorrecto.

Se marcan todos los autos como libres al iniciar server.

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

## Ciclo principal del servidor y recepción de comandos

El servidor se mantiene activo en un ciclo infinito, esperando conexiones TCP:  
Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Continua con los casos de las peticiones “estado”, “viaje”, “viaje\_terminado” .  
Recibe el mensaje del cliente y lo interpreta con un sistema de condicionales que prioriza los comandos más largos para evitar ambigüedades:

if (strncmp(buffer, "viaje\_terminado", 15) == 0) { ... }

else if (strncmp(buffer, "viaje\_cancelado", 15) == 0) { ... }

else if (strncmp(buffer, "viaje", 5) == 0) { ... }

else if (strncmp(buffer, "estado", 6) == 0) { ... }

else { /\* Servicio desconocido \*/ }

# evidencias Texto El contenido generado por IA puede ser incorrecto.

# conclusión

La práctica consolidó mi entendimiento del modelo cliente-servidor utilizando sockets TCP en C, con un enfoque en el diseño y la validación de protocolos de comunicación. Logré implementar un servidor capaz de manejar múltiples tipos de peticiones.