

Apellido y nombre: \_\_\_\_\_ Legajo: \_\_\_\_\_

**Control policial***Temas evaluados: Resolución de problemas, estructuras de datos, estructuras enlazadas.**Contexto*

Usted es parte de un equipo de desarrolladores que presta servicios profesionales a un distrito de la provincia de Buenos Aires. Su trabajo se centra en el desarrollo de aplicaciones que colaboren a resolver los problemas de seguridad en la zona de influencia

*Descripción*

Su tarea, en esta ocasión, consiste en colaborar con la división de tránsito y las fuerzas de seguridad una aplicación para controlar el paso de vehículos en diez zonas en la que está dividido el distrito. Entre las múltiples aplicaciones que surgen por el control vehicular, en este caso en particular se hace necesario establecer si vehículos sospechosos de algún ilícito pasaron por la zona, en caso de haber sido así poder establecer el recorrido por la zona

*Planteamiento del problema*

Las fuerzas de seguridad de la zona disponen de los datos de un vehículo sospechado de un ilícito. Cuenta con la siguiente información:

Patente	Delito que se sospecha	Hora del ilícito
LLDDDL	Cadena 50 caracteres	HHMMSS

Por otro lado, la división de tránsito dispone de un vector de listas con punteros, para cada una de las diez zonas, a lista de vehículos ordenados por hora de registro:

El vector tiene el número de la zona [1..10], denominación del lugar (cadena 30 caracteres), puntero a una estructura enlazada. La lista con la patente del vehículo registrado LLDDDL, hora de registro HHMMSS, ordenada por este campo.

*Observaciones*

- ✓ Las zonas son consecutivas pero el recorrido de un vehículo puede hacerse en cualquiera de las dos direcciones.
- ✓ Un vehículo puede pasar más de una vez por una zona y/o no pasar en alguna otra. Es decir, si pasa no necesariamente pasa por todas las zonas, y cuando pasa esa situación no es única en la zona

*Se dispone*

- ✓ La función de biblioteca cuyo prototipo es ***Nodo\* buscar(Nodo\* &, TipoDeDato)*** que busca en una estructura enlazada un nodo conteniendo el dato cuyo tipo se describe y retorna el puntero al nodo que contiene el dato buscado o NULL en caso de no encontrarlo. Por cada invocación solo retorna un nodo en caso de estar, recordar que la misma patente buscada puede repetirse más de una vez en cada una de las listas. Observar el modo en que se pasan los parámetros
- ✓ La función de biblioteca ***Nodo\* insertarOrdenado(Nodo\* &, tipoDeDato)***

Las funciones de biblioteca pueden utilizarse sin desarrollas SI u SOLO SI se las invoca según el prototipo

*Se pide*

1. Declarar todas las estructuras de datos involucradas en la aplicación que se propone (2 puntos).
2. Codificar el prototipo la función ***struct VehiculoSospechoso recorrido(lista de parámetros)*** que reciba las estructuras de datos con los vehículos registrados, la patente del vehículo sospechoso, el ilícito y retorna una struct con la patente del vehículo, el ilícito del que se sospecha pudo haber participado y un puntero a una lista con el recorrido del mismo y la siguiente información: la denominación de la zona y la hora de registro, ordenado por esto y solo en caso de que la hora de registro sea posterior a la hora del ilícito, caso contrario decida un valor de para ese campo y justifique la elección. (2 puntos).
3. Codificar o diagramar la función descrita en (2). (5 puntos).
4. Respecto al vector de lista del registro de automóviles:
  - a. Que estructura alternativa se puede proponer para asegurar el orden cronológico sin recorrer la lista en la carga. Justifique (1 punto)

Apellido y nombre: \_\_\_\_\_ Legajo: \_\_\_\_\_