

TÉCNICAS DE PROGRAMACIÓN
SEXTO LABORATORIO
SEMESTRE ACADÉMICO 2025-1

Horario: Todos

Duración: 110 minutos
Elaborado por los profesores del curso.

ADVERTENCIAS:

- Todo dispositivo electrónico (teléfono, tableta, computadora u otro) deberá permanecer apagado durante la evaluación en su mochila.
- Coloque todo aquello que no sean útiles de uso autorizado durante la evaluación en la parte delantera del aula, por ejemplo, mochila, maletín, cartera o similar, y procure que contenga todas sus propiedades. La apropiada identificación de las pertenencias es su responsabilidad.
- Si se detecta omisión a los dos puntos anteriores, la evaluación será considerada nula y podrá conllevar el inicio de un procedimiento disciplinario en determinados casos.
- Es su responsabilidad tomar las precauciones necesarias para no requerir la utilización de servicios higiénicos: durante la evaluación, no podrá acceder a ellos, de tener alguna emergencia comunicárselo a su jefe de práctica.
- Quienes deseen retirarse del aula y dar por concluida su evaluación no lo podrán hacer dentro de la primera mitad del tiempo de duración destinado a ella.

INDICACIONES:

- No se pueden usar apuntes de clase ni calculadoras.
- LAS SOLUCIONES DEBERÁN DESARROLLARSE BAJO UN ESTRICTO DISEÑO DESCENDENTE, por lo que NO SE CALIFICARÁN aquellos módulos que son llamados por otros que estén incompletos. Cada módulo no debe sobrepasar las 30 líneas de código aproximadamente.
- **NO SE PUEDEN EMPLEAR ARCHIVOS DE DATOS NI ARREGLOS AUXILIARES A LOS INDICADOS EN LA PRUEBA.** Tampoco podrá emplear VARIABLES GLOBALES y NO podrá implementar funciones en el archivo main.cpp, las funciones o tareas se deberán implementar en archivos independientes (.h y .cpp). **DE NO ACATAR ESTA INDICACIÓN SE COLOCARÁ COMO NOTA FINAL PARA ESTA PRUEBA DE CERO. NO SE HARÁN EXCEPCIONES.**
- No se calificará el código puesto como comentario ni aquellas funciones que no son invocadas en el proyecto o aquellas que su invocación esté comentada.
- En la calificación se tomará en cuenta el buen uso de los nombres de los identificadores, y el eficaz uso de comentarios.
- **DEBE COLOCAR SU NOMBRE Y CÓDIGO EN EL ARCHIVO main.cpp, DE LO CONTRARIO SE LE DESCOTARÁ 0.5 PUNTOS. NO SE HARÁN EXCEPCIONES.**
- **DEBE COLOCAR UN COMENTARIO AL INICIO DEL ARCHIVO main.cpp CON UNA DESCRIPCIÓN DE LO QUE HACE EL PROGRAMA. ESTA DESCRIPCIÓN NO DEBE SER GENÉRICA, DEBE SER ESPECÍFICA DE LO QUE HARÁ EL PROGRAMA. SE LE DESCOTARÁ 0.5 PUNTOS EN SU NOTA FINAL SI NO SE COLOCA ESTE COMENTARIO O LO QUE SE EXPRESE EN ÉL NO SEA ESPECÍFICO. NO SE HARÁN EXCEPCIONES.**
- **NO PUEDE EMPLEAR LA BIBLIOTECA "cstring", "string.h" o SIMILARES. TAMPOCO SE PUEDEN EMPLEAR LAS FUNCIONES get() ni getline() PARA LEE CADENAS COMPLETAS.**
- **NO SE PERMITIRÁ LA LECTURA DE DATOS CARÁCTER POR CARÁCTER.**
- **TODA OPERACIÓN DE BÚSQUEDA DEBE REALIZARSE EN FUNCIONES INDEPENDIENTES Y DEBE CONSIDERAR QUE EL DATO BUSCADO NO SE ENCUENTRE.**

INDICACIONES INICIALES

Siga estrictamente las indicaciones que a continuación se detallan:

- En la unidad de trabajo indicada por los Jefes de práctica (si trabaja en otra unidad, no se calificará el laboratorio y se le asignará como nota cero) Cree allí una carpeta con el nombre "Laboratorio_06_2025_1_CO_PA_PN_" donde **CO** indica: Código del alumno, **PA** indica: Primer Apellido del alumno y **PN** primer nombre (de no colocar este requerimiento **se le descontará 2 puntos de la nota final**). **Allí colocará los proyectos solicitados en la prueba.**

La finalidad de este laboratorio es la de reforzar los conceptos de arreglos y cadenas de caracteres.

Cree un proyecto en NetBeans con el nombre: "Laboratorio06_2025_1", dentro de la carpeta anterior (de no respetar este nombre se le descontarán dos puntos de su nota final – NO SE HARÁN EXCEPCIONES) y en él desarrolle el programa que resuelva el problema que se describe a continuación. El proyecto debe estar dentro de la carpeta creada previamente.

DEBE LEER TODA LA PRUEBA ANTES DE EMPEZAR A DESARROLLAR EL PROGRAMA

El Ministerio de Transportes lleva un registro de las infracciones de tránsito cometidas por diferentes empresas, estos registros están almacenados en diferentes archivos de textos como los que se muestran a continuación:

TablaDeInfracciones.txt

2007	425.23	No conducir por el carril de extremo derecho de la calzada un vehiculo de ...
1029	586.45	Conducir un vehiculo especial que no se ajuste a las exigencias reglamentarias...
...

Según lo que se ve en el archivo, en cada línea aparecen las características de una infracción de tránsito, primero aparece el código de la infracción, el valor de la multa y la descripción de la infracción.

InfraccionesCometidas.txt

1/12/2020	G760-721	89314225	2022			
23/07/2023	P474-593	89941095	2060	P	22/08/2023	
...			

El archivo contiene en cada línea las características de una infracción cometida. Primero aparece la fecha en la que se cometió, le sigue la placa del vehículo de la empresa que cometió la infracción, el DNI de representante legal de la empresa y el código de la infracción. En caso la infracción se haya pagado aparece P seguido de la fecha de pago, de no haberse pagado no aparecen estos dos últimos datos.

El Ministerio de Transportes desea controlar las infracciones pagadas y pendientes de pago, por esta razón se le solicita emitir un reporte similar al siguiente:

MINISTERIO DE TRANSPORTES LISTADO DE INFRACCIONES							
No.	DNI	PLACAS	MULTA MAS ANTIGUA	ULTIMA MULTA PAGADA	TOTAL EN MULTAS	MULTAS PAGADA	MULTAS PENDIENTES
1)	14658855	P310-495/G222-555/G123-789	23/12/2021	09/05/2025	3789.25	2723.09	1066.09
...
17)	42773344	F345-345/M135-791	20/08/2023	NO HAY REGISTRO	5878.80	0.00	5878.80
...
TOTALES				

Las tareas por realizar son las siguientes, las cuales debe desarrollarlas en el orden que se indican y cada una en una función independiente:

- (6 puntos) Leer los datos del archivo “*TablaDeInfracciones.txt*” y con ellos llenar en los siguientes arreglos: *arrCodigoInfraccion*, *arrMultasInfraccion*. Para realizar esta labor, conforme vaya leyendo los datos deberá insertar los datos de manera ordenada ascendentemente por el código de la infracción. El archivo solo se puede leer una vez.
- (2 puntos) En un archivo de textos denominado “*PruebaTablaInf.txt*” verifique el correcto llenado de los datos. Éstos deben tener un título que indique de qué datos se trata, encabezados simples encima de cada columna que indique lo que contienen y los datos deben tener una alineación y formato apropiado de acuerdo con el tipo de dato.
- (7 puntos) Haciendo uso del archivo “*InfraccionesCometidas.txt*”, complete las cantidades que se mostrarán en el reporte en los siguientes arreglos: *arrDni*, *arrMasMultaAntigua*, *arrUltimaPendiente*, *arrTotalMultas*, *arrTotalPagadas* y *arrTotalPendientes*. Conforme vaya leyendo los datos guarde en un nuevo archivo de textos denominado “*PlacasRegistradas.txt*” el DNI y la placa de la infracción, en una sola línea ambos datos y separado por un espacio en blanco. En el proceso de llenado de los arreglos no debe repetir los dni, por eso primero debe buscar el DNI en el arreglo si lo encuentra solo debe acumular o modificar los datos en el arreglo si no lo encuentra debe inicializar los datos en los arreglos con los datos leídos. La búsqueda debe ser binaria.
- (5 puntos) Emita el reporte solicitado en un archivo denominado “*ReporteDeMultas.txt*” . Para imprimir las placas debe implementar las siguientes funciones y realizar el siguiente proceso:

Función: *obtenerPlaca* (*const char *listaDePlacas, int n, char *placa*); donde:

- *listaDePlacas* es una cadena de la forma “P222-555/G111-111/M555-777” (pueden haber más o menos placas).
- *n* es un valor entero como 3, 1 o 4, etc.
- *placa* es una cadena no inicializada

La función debe contar el número de placas que contiene la cadena “*listaDePlacas*”. Si el valor de “*n*” es mayor que el número de placas o de 1, la función no realiza acción alguna, de lo contrario debe copiar los caracteres de la *n*-ésima placa en la cadena “*placa*”. Dese cuenta que todas las placas tienen la misma cantidad de caracteres.

Función: *agregarPlaca* ((*char *listaDePlacas, const char *placa*); donde:

- *listaDePlacas* es una cadena de la forma “P222-555/G111-111/M555-777”.
- *placa* es una cadena de la forma “M353-868”.

La función debe contar el número de placas que contiene la cadena “*listaDePlacas*”. Luego, usando la función “*obtenerPlaca*”, debe extraer una a una las placas de la cadena “*listaDePlacas*”. Cada placa extraída debe compararse con la “*placa*” que ingresa como parámetro, si encuentra que son iguales la función no realiza acción alguna, de lo contrario se debe concatenar la “*placa*” a la “*listaDePlacas*” anteponiéndole el carácter ‘/’.

El proceso para imprimir las placas en el reporte debe tomar del arreglo “*arrDni*,” el DNI que se va a imprimir y extraer del archivo “*PlacasRegistradas.txt*” una a una las placas que le pertenecen a la empresa y forme el texto a imprimir empleando las funciones antes descritas. Podrá recorrer el archivo tantas veces como empresas existan.

Las tareas que no se desarrollen en el orden solicitado o que se desarrolle sin que haya terminado la anterior no se calificarán.

Al finalizar el examen, comprima la carpeta que contiene el proyecto empleando el programa Zip que viene por defecto en el Windows, no se aceptarán los trabajos compactados con otros programas como RAR, WinRAR, 7zip o similares y súbalo a la tarea programada en PAIDEIA para este examen.

CRITERIOS DE CALIFICACIÓN:
1. Si el programa entregado presenta más de tres errores de sintaxis serán calificados sobre la mitad del puntaje.
2. Si el programa no muestra los resultados o los muestren y no sean correctos, no podrán tener más del 75% de la nota.
3. Se descontará 15% de la nota si el programa define variables con nombres que no tengan sentido.
4. Se descontará 15% de la nota si no se colocan comentarios relevantes, incluyendo un encabezado al inicio del programa en el que se indique el nombre del autor, la fecha, y una descripción de lo que hace programa).
5. No se calificarán aquellas funciones implementadas en el archivo main.cpp

San Miguel, lunes 2 de junio del 2025