

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

PROGRAMACIÓN 2
4ta práctica (tipo b)
Primer Semestre 2025

Indicaciones Generales:

- Duración: 110 minutos.

NO SE PERMITE EL USO DE APUNTES DE CLASE, FOTOCOPIAS NI MATERIAL IMPRESO

- No se pueden emplear **variables globales, NI OBJETOS** (con excepción de los elementos de `iostream`, `omanip` y `fstream`). **NO PUEDE UTILIZAR LA CLASE `string`**. Tampoco se podrán emplear las funciones `malloc`, `realloc`, `memset`, `strtok` o `strdup`, igualmente no se puede emplear cualquier función contenida en las bibliotecas `stdio.h`, `cstdio` o similares y que puedan estar también definidas en otras bibliotecas. **NO PODRÁ EMPLEAR PLANTILLAS EN ESTE LABORATORIO**
- Deberá modular correctamente el proyecto en archivos independientes. LAS SOLUCIONES DEBERÁN DESARROLLARSE BAJO UN ESTRICTO DISEÑO DESCENDENTE. **Cada función NO debe sobrepasar las 20 líneas de código aproximadamente**. El archivo `main.cpp` solo podrá contener la función `main` de cada proyecto y el código contenido en él solo podrá estar conformado por tareas implementadas como funciones. En el archivo `main.cpp` deberá colocar un comentario en el que coloque claramente su nombre y código, **de no hacerlo se le descontará 0.5 puntos en la nota final**.
- El código comentado NO SE CALIFICARÁ. De igual manera NO SE CALIFICARÁ el código de una función si esta función no es llamada en ninguna parte del proyecto o su llamado está comentado.
- Los programas que presenten errores de sintaxis o de concepto se calificarán en base al 40% de puntaje de la pregunta. Los que no muestren resultados o que estos no sean coherentes en base al 60%.
- Se tomará en cuenta en la calificación el uso de comentarios relevantes.

SE LES RECUERDA QUE, DE ACUERDO AL REGLAMENTO DISCIPLINARIO DE NUESTRA INSTITUCIÓN, CONSTITUYE UNA FALTA GRAVE COPIAR DEL TRABAJO REALIZADO POR OTRA PERSONA O COMETER PLAGIO.

**NO SE HARÁN EXCEPCIONES ANTE CUALQUIER TRASGRESIÓN DE LAS
INDICACIONES DADAS EN LA PRUEBA**

- **Puntaje total:** 20 puntos.

INDICACIONES INICIALES

Cree un proyecto de C++ en NetBeans siguiendo estrictamente las indicaciones que a continuación se detallan:

- La unidad de trabajo será **t:** (Si lo coloca en otra unidad, no se calificará su laboratorio y se le asignará como nota cero)
- Cree allí una carpeta con el nombre **"Lab04_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 3 puntos de la nota final).

Cuestionario:

La finalidad principal de este laboratorio es la de reforzar los conceptos contenidos en el capítulo 5 del curso: "Listas Genéricas". En este laboratorio se trabajará con punteros sin tipo, así como la implementación de TAD's.

Deberá elaborar un proyecto denominado **"ListaGenerica"** y en él desarrollará el programa que dé solución al problema planteado. **DE NO COLOCAR ESTE REQUERIMIENTO SE LE DESCONTARÁ 3 PUNTOS DE LA NOTA FINAL.**

Se tienen tres archivos del tipo CSV, los cuales se describen a continuación:

RegistroDeFaltas1.csv									
26522329	U5W-825	22/11/2019	101	SANCHEZ DEL CASTILLO	FIGUEROA	ROSARIO			
47844324	X7J-719	14/09/2019	101	VILCA LEE	NESTOR	HUGO			
53374153	R2T-476	5/09/2019	101	SERRANO	IBEROS	FRANCISCO			
...									
licencia, placa, fecha, infracción, nombre del conductor									

Con esta información, la función "main" del proyecto estará compuesto por el siguiente código:

```
#include "BibliotecaGenerica.h"
#include "BibliotecaRegistros.h"
#include "BibliotecaEnteros.h"

using namespace std;

int main(int argc, char** argv) {
    void *lista;

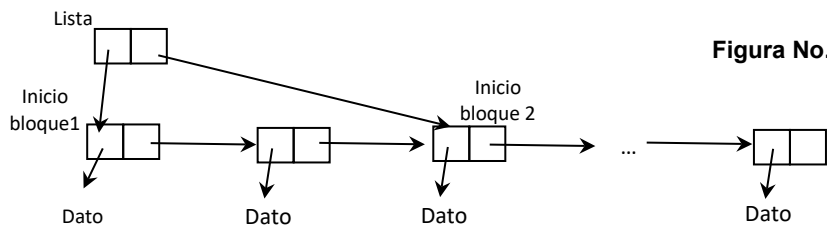
    creaLista(lista, leenum, clasificaentero, "numeros.txt");
    imprimelistas(lista, imprimenum, "Repnum.txt");

    creaLista(lista, leeregistro, clasificaregistro, "RegistroDeFaltas1.csv");
    imprimelistas(lista, imprimeregistro, "Repfalta.txt");

    return 0;
}
```

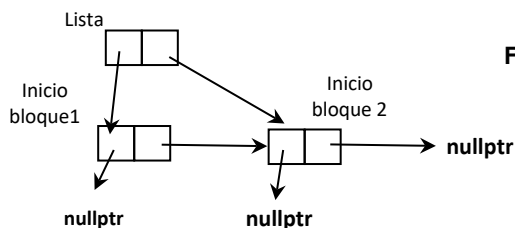
**NO PUEDE
CAMBIAR
ESTE CÓDIGO**

Para solucionar este problema se le solicita que defina una lista genérica simplemente ligada. La misma debe tener la siguiente estructura:



Esta estructura debe contar con una biblioteca **BibliotecaListaGenerica**, con las funciones necesarias para su soporte, como son: **creaLista**, **generalista**, **insertaLista**, e **imprimeLista**.

- La función **creaLista** recibe el nombre del archivo a ser cargado en la lista e invoca a las funciones **generalista** e **insertaLista**.
- La función **generalista** se encargará de crear la lista vacía de forma que quede lista para insertar los datos de acuerdo con los bloques definidos. La lista inicial debe tener la estructura que se muestra en la Figura 2.



- La función **insertaLista** se encargará de llenar la lista de acuerdo con el bloque que le corresponde al dato a ser insertado, para esta tarea debe utilizar el puntero a función **clasifica**. Los nodos de inicio de cada bloque también deben contener datos. **El orden de los datos en cada bloque es el mismo que tiene en el archivo leído. Recuerde que luego de cada inserción la lista con sus dos bloques debe estar siempre unida.**
- La función **imprimeLista** recibe el nombre del archivo donde realizará la impresión de la lista según lo solicitado en cada pregunta. Para esta operación se debe recorrer la lista completamente considerando que ambos bloques siempre están unidos.

Pregunta 1 (14 puntos)

Desarrolle la biblioteca **BibliotecaListaGenerica**, que brinde soporte a las listas que aparecen en el main de acuerdo con la Figura 1. Para probar el buen funcionamiento de esta biblioteca genérica, desarrolle la biblioteca **BibliotecaEnteros** la cual debe trabajar **solo** con números enteros. Los números deben insertarse en cada bloque de la lista de la siguiente forma: los números menores a 10 van en el bloque 1 y el resto en el bloque 2. Finalmente debe imprimir la lista en el archivo brindado como parámetro de la función. De la siguiente forma:

4
5
6
10
20
33
90
...

- (3.0 puntos) Desarrolle la función **generaLista**
- (4.0 puntos) Desarrolle la función **insertaLista**
- (2.0 puntos) Desarrolle la función **creaLista**
- (2.0 puntos) Desarrolle la función **imprimeLista**
- (3.0 puntos) Desarrolle los punteros a función propios del archivo con números

Pregunta 2 (6 puntos)

Una vez comprobado el buen funcionamiento de su biblioteca genérica, desarrolle la biblioteca **BibliotecaRegistros**, donde deberá cargar todos los datos del archivo Registro de Faltas en la lista genérica de acuerdo con el tipo de falta de la siguiente forma: si la falta inicia con 1 debe insertarse en el primer bloque, el resto debe insertarse en el segundo bloque. Finalmente debe emitir un reporte similar al que se muestra a continuación:

Reporte

FECHA	LICENCIA	NOMBRE	FALTA
2019/11/22	26522329	SANCHEZ DEL CASTILLO FIORELLA ROSARIO	101
2019/09/14	47844324	VILCA LEE NESTOR HUGO	101
2019/09/05	53374153	SERRANO IBEROS FRANCISCO	101
2019/05/09	19568447	VEGA SORIA HENRRY	102
2019/01/20	25518120	RONDON CABRERA ALEXANDER ADOLFO	103
...			

Al finalizar la práctica, **comprima** la carpeta dada en las indicaciones iniciales 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.**

Profesores del curso:
Rony Cueva
Erick Huiza
Miguel Guanira

Erasmus Gómez
Andrés Melgar

San Miguel, 9 de mayo del 2025.