

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

PROGRAMACIÓN 2
5ta práctica (tipo b)
Primer Semestre 2024

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 de C que gestionen memoria como `malloc`, `realloc`, `memset`, `strdup`, `strtok` o similares, 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 **"CO_PA_PN_Lab05_2024_1"** 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). **Allí colocará el proyecto solicitado en la prueba.**

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 dos archivos del tipo CSV, con la estructura que se describen a continuación:

Pedidos3#.csv
2,14268463,BHD6079
7,42302422,RWW7975
9,11750801,VDL9379
...

Núm. de pedido, DNI , Cód. Libro

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

```
#include "BibliotecaGenerica.h"
#include "FuncionesEnteras.h"
#include "FuncionesRegistros.h"

using namespace std;
int main(int argc, char** argv) {
    void *pedidos1,*pedidos2,*pedidosfinal;

    crealista(pedidos1,leenum,"RegistroNum1.txt");
    imprimelista(pedidos1,imprimenum,"Repnum1.txt");
    crealista(pedidos2,leenum,"RegistroNum2.txt");
    imprimelista(pedidos2,imprimenum,"Repnum2.txt");
    combinalista(pedidos1,pedidos2,pedidosfinal,cmpnum);
    imprimelista(pedidosfinal,imprimenum,"RepnumFinal.txt");

    crealista(pedidos1,leeregistro,"Pedidos31.csv");
    imprimelista(pedidos1,imprimeregistro,"Repreg1.txt");
    crealista(pedidos2,leeregistro,"Pedidos32.csv");
    imprimelista(pedidos2,imprimeregistro,"Repreg2.txt");
    combinalista(pedidos1,pedidos2,pedidosfinal,cmpregistro);
    imprimelista(pedidosfinal,imprimeregistro,"RepregFinal.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:

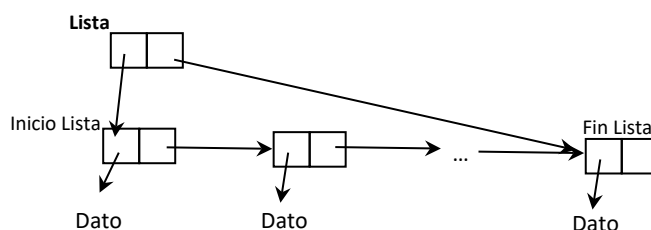


Figura No. 1

Esta estructura debe contar con una biblioteca denominada **BibliotecaGenerica**, con las funciones necesarias para su soporte, como son las funciones **crealista**, **insertalista** (coloca un dato genérico al final de la lista), **quitalista** (devuelve un dato genérico, eliminando el nodo inicial de la lista), **listavacia**, **combinalista** e **imprimelista**. La función **crealista** recibe el nombre del archivo a ser cargado en la lista. Con el archivo **Pedidos31.csv** debe crear la lista **pedidos1**, con el archivo **Pedidos32.csv** debe crear la lista **pedidos2**, ambos archivos están ordenados por tal motivo al insertar al **final** de la lista se obtendrán 2 listas ordenadas. Luego con estas listas cargadas y ordenadas, debe realizar una combinación o **merge** de estas estructuras utilizando la función **combinalista**, de tal forma que se genere una **nueva lista ordenada**. Para este proceso solo puede recorrer una vez cada lista de pedidos. La lista unificada debe almacenarse en la lista **pedidosfinal**, siguiendo la estructura de la figura1. En caso exista el mismo valor en la lista 1 y 2, solo debe considerar uno de ellos en la lista final. La función **imprimelista** recibe el nombre del archivo donde realizará la impresión de la lista según lo solicitado en cada pregunta.

Pregunta 1 (5 puntos)

Desarrolle la biblioteca **BibliotecaGenerica**, que brinde soporte a la creación e impresión de las listas que aparecen en el main de acuerdo con la figura1. Para probar el buen funcionamiento de esta biblioteca genérica, desarrolle la biblioteca **FuncionesEnteras** la cual debe trabajar **solo** con los enteros, contenidos en los archivos RegistroNum#.txt. Los números deben insertarse siempre al final de la lista e imprimirse de la siguiente forma:

```
4
6
7
8
...
```

Pregunta 2 (8 puntos)

Desarrolle en la biblioteca **BibliotecaGenerica**, la función **combinalista** que realice la operación descrita en el proceso de combinación de listas. Recuerde que las funciones de inserción y retiro de nodos de la lista **NO** deben recorrer la estructura, para eso debe hacer un manejo adecuado de los punteros. **Esta pregunta no tiene validez si no se desarrolla la pregunta anterior.** Luego del proceso de combinación de listas debe emitir el siguiente reporte:

Reporte de números Final

```
1
2
3
4
5
6
7
...
```

Pregunta 3 (7 puntos)

Una vez comprobado el buen funcionamiento de su biblioteca genérica, desarrolle la biblioteca **FuncionesRegistros**, donde deberá cargar todos los datos de los archivos Pedidos3#.csv en cada lista respectiva, la información se encuentra ordenada por el número del pedido. Al igual que las preguntas anteriores, luego de cargar las dos listas debe combinarlas en una sola y emitir un reporte similar al que se muestra a continuación:

Reporte de Registros Final

1	69200503	ZQY9966
2	14268463	BHD6079
3	81082011	QRH6838
4	88080249	NEO9165
5	27109544	CMP1150
6	57431952	CRR5443
7	42302422	RWW7975
8	83265244	RSK1164
9	11750801	VDL9379
10	37751890	KWL0415
...		

Para esta pregunta debe emplear las mismas funciones de la biblioteca **BibliotecaGenerica** empleadas en los archivos de números, **en caso contrario su solución no tiene valor.**

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.

Erick Huiza
Miguel Guanira

Heider Sánchez

San Miguel, 3 de mayo del 2024.