

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**  
**FACULTAD DE CIENCIAS E INGENIERÍA**

**PROGRAMACIÓN 2**  
**5ta práctica (tipo b)**  
**Segundo 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 `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 **"Lab05\_2024\_2\_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 dos archivos del tipo CSV, con la estructura que se describen a continuación:

Repartidores.csv
JNV387,Justino Norabuena Virginia Karina,Motocicleta
PRT150,Pairazaman Raffo Tatiana Delicia,Bicicleta
...

Código del repartidor, Nombre, Unidad

Ordenes.csv
12484697,2,AD-546,LAF361
12484697,1,PO-751,LAF361
...

DNI, Cantidad, Plato, Código del repartidor

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

```
#include "BibliotecaGenerica.h"
#include "ListaconEnteros.h"
#include "ListaconRegistros.h"

using namespace std;

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

    crealista(lista,leenumeros,"numeros1.txt");
    cargalista(lista,compruebanumero,leeregnumeros,"numeros2.txt");
    muestralista(lista,imprimenumeros,"repnum.txt");

    crealista(lista,leeregistros,"repartidores.csv");
    cargalista(lista,compruebaregistro,leeordenes,"ordenes.csv");
    muestralista(lista,imprimeregistros,"reregistros.txt");

    return 0;
}
```

**NO PUEDE  
CAMBIAR  
ESTE CÓDIGO**

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

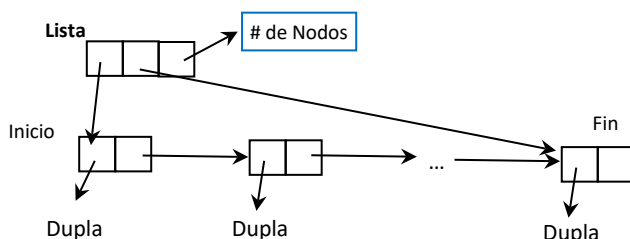


Figura No. 1

Esta estructura debe contar con una biblioteca denominada **BibliotecaGenerica**, con las funciones necesarias para su soporte, como son las funciones **construir**, **insertafinal**, **eslistavacia**, **crealista**, **cargalista** y **muestralista**. La función **crealista** colocará los repartidores en la lista, para esta operación cargará la información en la posición 0 de la dupla que se muestra en la figura 2, debe generar el arreglo de ordenes en la posición 1, si así lo cree necesario. La función **cargalista** colocará las órdenes del repartidor correspondiente en el arreglo fijo de 20 elementos que se encuentra definido en la posición 1 de la dupla. La función **muestralista** recibe el nombre del archivo donde realizará la impresión de la lista y el contenido del arreglo, según lo solicitado en cada pregunta.

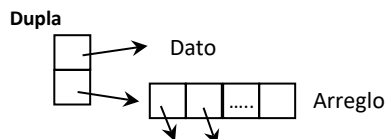


Figura No. 2

### Pregunta 1 (5 puntos)

Desarrolle la biblioteca **BibliotecaGenerica**, que brinde soporte a la creación y llenado (función **crealista**) de la lista que aparece en el main de acuerdo con la figura 1. Para probar el buen funcionamiento de esta biblioteca genérica, desarrolle la biblioteca **ListaconEnteros** la cual debe trabajar **solo** con números

enteros, contenidos en el archivo **numeros1.txt**. Los números deben ser insertados empleando la estructura dupla de la figura 2, para este caso deben registrarse en la posición 0 de la dupla. A continuación, un ejemplo del archivo a emplearse:

```
2
4
6
...
```

### Pregunta 2 (5 puntos)

Desarrolle en la biblioteca **BibliotecaGenerica**, la función **cargalista** que se encarga de registrar los datos contenidos en el archivo **numeros2.txt** dentro del arreglo de la dupla. Como se trata de un archivo diferente debe emplear un puntero a función con tipo específico para esta operación (**leeregnumeros**). Como se aprecia el archivo de números cuenta con 2 campos, el primero sirve como clave a buscarse en la dupla[0], por lo tanto, para comparar estos valores debe utilizar un puntero a función denominado **compruebanumero**. Una vez ubicada la dupla, debe insertarse segundo valor del archivo de texto en el arreglo correspondiente. **Esta pregunta no tiene validez si no se desarrolla la pregunta anterior.** A continuación, un ejemplo del archivo de texto:

```
2 4
2 2
4 1
6 2
9 4
...
```

### Pregunta 3 (3 puntos)

Desarrolle en la biblioteca **BibliotecaGenerica**, la función **muestralista** que se encarga de mostrar los datos registrados en la lista, recuerde considerar ambos valores contenidos en la dupla de acuerdo con el siguiente reporte:

```
2
4 2 1 9

4
1 3 2

6
2 9
...
```

### Pregunta 4 (2 puntos)

Una vez culminada la biblioteca **BibliotecaGenerica**, desarrolle la biblioteca **ListaconRegistros** la cual debe trabajar con los datos contenidos en el archivo **Repartidores.csv**. Como primera funcionalidad debe implementar el puntero a función **leeregistros** que se muestra en el main, que complementando a la función genérica **crealista** insertará los registros en cada nodo, considerando siempre el uso de la estructura dupla de la figura 2, para este caso deben emplearse la posición 0 de la dupla, ya que estos valores servirán como llave de búsqueda para la posterior carga de las órdenes. En este caso los valores a registrarse son: Código del repartidor, Nombre, Unidad.

### Pregunta 5 (3 puntos)

Desarrolle en la biblioteca **ListaconRegistros**, las funciones que ayudarán a registrar los datos contenidos en el archivo **Ordenes.csv** dentro del arreglo de la dupla complementando la función genérica **cargalista**.

Como se trata de un archivo diferente al de repartidores debe emplear un puntero a función específico para esta operación (**leeordenes**). Para ubicar la orden en el repartidor correspondiente debe emplear un puntero a función **compruebaregistro**. Una vez ubicada la dupla, debe insertarse los datos de la orden en el arreglo correspondiente. En este caso los valores a registrarse son: DNI, Cantidad, Plato. **Esta pregunta no tiene validez si no se desarrolla la pregunta anterior.**

**Pregunta 6 (2 puntos)**

Desarrolle en la biblioteca **ListaconRegistros**, la función **imprimeregistro** que complementando la función **muestralista** se encarga de imprimir los datos registrados en la lista, recuerde considerar ambos valores contenidos en la dupla de acuerdo con el siguiente reporte:

JNV387	Justino Norabuena Virginia Karina
Ordenes:	
22598563	1 BE-265
22598563	1 BR-592
PRT150	Pairazaman Raffo Tatiana Delicia
Ordenes:	
68114999	3 BR-926
68114999	1 EN-419
68114999	1 AD-518
...	

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, 25 de octubre del 2024.