

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

**LENGUAJE DE PROGRAMACIÓN 1**

**6ta práctica (tipo b)**  
**(Segundo Semestre 2018)**

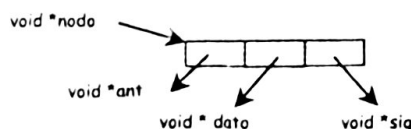
**Indicaciones Generales:**

- Duración: 110 minutos.
- Se podrá usar como material de consulta solo sus apuntes de clase.
- No se pueden emplear variables globales, estructuras, ni objetos (con excepción de los elementos de `iostream` e `iomanip`). Tampoco se podrán emplear las funciones `malloc`, `realloc`, `strdup` o `strtok`.
- Deberá modular correctamente el proyecto en archivos independientes. **LAS SOLUCIONES DEBERÁN DESARROLLARSE BAJO UN ESTRICTO DISEÑO DESCENDENTE.** Cada módulo **NO** debe sobrepasar las 20 líneas de código aproximadamente. El archivo `main.cpp` solo podrá contener la función `main` de cada proyecto. En cada archivo que implemente en los proyectos (.h y .cpp) deberá colocar un comentario en el que coloque claramente su nombre y código, de no hacerlo se le descontará 0.5 puntos por archivo. Se tomará en cuenta en la calificación el uso de comentarios relevantes.
- Los programas que presenten errores de sintaxis o de concepto se calificarán en base al 40% del puntaje de la pregunta. Los que no den resultados coherentes en base al 60%.
- La presentación, la ortografía y la gramática de los trabajos influirá en la calificación.

**Puntaje total: 20 puntos**

**Problema**

Se desea elaborar una biblioteca de funciones que permita crear y manejar una lista doblemente ligada (LDL) para el almacenamiento y búsqueda ordenada de datos genéricos. La biblioteca deberá manejar obligatoriamente punteros a funciones / permitirá crear una LDL leyendo los datos e insertándolos de forma ordenada en el árbol según un criterio especificado. También se deberá poder buscar un elemento en la lista y devolver un puntero al dato completo buscado. Finalmente, se podrá imprimir en orden todos los datos de la lista. La estructura deberá ser capaz de manejar cualquier tipo de dato, por lo que debe ser absolutamente independiente del tipo de dato que podrá manejar. Los nodos de la LDL deberán manejarse obligatoriamente en base a punteros genéricos como se muestra a continuación:



**Pregunta 1** (8 puntos)

Elabore y pruebe la biblioteca de funciones (archivos `listaDLGen.h` y `listaDLGen.cpp`) para manejar la lista descrita anteriormente. [NO SE PIDE IMPLEMENTAR UNA BIBLIOTECA ESTÁTICA]

La elaboración de la biblioteca deberá estar incluida en un proyecto que permita probarla. El proyecto, denominado "ListaGenérica" permitirá: leer desde la entrada estándar un conjunto de pares de datos (nombre y código numérico), construir con ellos una lista ordenada por el nombre, imprimir los pares ya ordenados y finalmente pedirle a usuario que ingrese un nombre, buscarlo en la lista, y si lo encuentra imprimir ambos datos.

**Pregunta 2** (12 puntos)

Se tiene un archivo CSV con la siguiente información de una lista de novelas:

Archivo: "novelas.csv"
Cien años de soledad.Gabriel García Marquez.-0.272209;0.237176;-0.48612
La ciudad y los perros.Mario Vargas Llosa;-0.186415;0.081194;-0.448023

En cada línea aparece el título de una novela, el nombre de su autor, y 300 valores reales, los que corresponden a un vector que representa dicha novela en un espacio vectorial de 300 dimensiones. Los valores de estos vectores han sido generados con algoritmos de aprendizaje automático, y tratan de

mantener una información numérica respecto al texto contenido en las novelas, por lo que serán útiles en una función de comparación que se plantea más adelante.

Se pide que elabore un proyecto, denominado "RecomendacionNovelas", el cual será usado para ordenar las novelas y permitir que un usuario busque recomendaciones de novelas para leer en base a un título específico, para lo cual el sistema usará la información de los autores y el vector de representación proporcionado. Por ello, debe realizar lo siguiente:

- El programa debe construir una lista doblemente enlazada a partir del archivo "novelas.csv". El archivo debe ser leído como tal, y no por redirección de la entrada estándar. Deberá emplear obligatoriamente las funciones de la biblioteca genérica implementada en la Pregunta 1. La información que se debe guardar por cada novela en cada será: título, autor y vector (arreglo de tipo float de 300 elementos). Use memoria dinámica y reserve espacios de memoria exactos. La lista debe quedar ordenada por el nombre del autor, y luego por el título de la novela. Finalmente, deberá grabar los datos ordenados (sólo el título y el autor) en un archivo de reporte denominado: "Lista de novelas.txt". Los datos deben aparecer correctamente tabulados y alineados con títulos y etiquetas que expliquen la naturaleza de los datos (no podrá usar el carácter '\t'). (4 puntos)
- El programa deberá solicitar al usuario por la entrada estándar que haga la consulta sobre una novela, solicitándole que introduzca un título. El objetivo será identificar otras novelas del mismo autor, para lo que debe implementar una funcionalidad que permita: (1) buscar la posición en la lista de la novela, (2) recorrer hacia la izquierda buscando novelas del mismo autor, (3) recorrer hacia la derecha buscando novelas del mismo autor. Como la lista está ordenada por autores, los pasos 2 y 3 deben aprovechar la característica de doble enlace de la lista generada. Se debe mostrar el resultado en la salida estándar, y en caso no se encuentre la novela mostrar un mensaje al usuario apropiado. La solicitud de títulos debe realizarse hasta que el usuario escriba "FIN". (4 puntos)

En este punto se espera mejorar la funcionalidad de recomendación de novelas usando el vector numérico, por lo que debe implementar un nuevo proyecto: "RecomendacionNovelasPorSimilitud" y copie toda la información del proyecto anterior. En este caso, se mostrará la novela más cercana al título consultado aprovechando los vectores numéricos, los cuales serán comparados a partir del coseno del ángulo que forman entre ellos, según la siguiente fórmula:

$$\text{similitud} = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

Donde A y B son dos vectores a comparar, y 'n' es igual a 300 según este problema. Es decir, debe: (1) obtener el vector del título a buscar, (2) recorrer toda la lista y calcular la similitud de dicho vector contra el resto de novelas, (3) identificar la novela cuyo valor de similitud sea el mayor (el rango de valores a calcular va entre 0 a 1). Finalmente, debe imprimir en la salida estándar el título y autor de la novela a recomendar, además de las novelas del mismo autor obtenidas en el paso anterior. (4 puntos).

### Anotaciones finales

Al finalizar el laboratorio, comprima<sup>1</sup> la carpeta "Laboratorio6" en un archivo con nombre <código del alumno con 8 dígitos>.zip y súbalo a la intranet del curso, en el enlace Documentos, en la carpeta \Laboratorio6\<código del horario>\<aula>.

Profesores del curso: Arturo Oncevay, Miguel Guanira

San Miguel, 2 de noviembre del 2018.

<sup>1</sup> Para evitar problemas en la corrección de la prueba, utilice el programa de compresión que viene por defecto en el Windows (Zip) no 7z.