

Trabajo práctico número 3
“LinkedList”
Programación I – Laboratorio I.
Tecnicatura Superior en Programación.
UTN-FRA

Autores: *Marina Cardozo*

Revisores: *Laura Carelli*

Versión : 3



Esta obra está bajo una [Licencia Creative Commons Atribución-CompartirIgual 4.0 Internacional](http://creativecommons.org/licenses/by-sa/4.0/).

Índice de contenido

1 Enunciado	3
2 Biblioteca LinkedList	3
3 Cómo realizar y entregar el trabajo práctico	4
3.1 Objetivo.....	4

1 Enunciado

Una aerolínea requiere un sistema para administrar los pasajeros de cada vuelo, para lo cual se deberá desarrollar una solución implementando la biblioteca **LinkedList**. Se deberá modelar la entidad que representa un pasajero con todos sus datos asociados de tal manera que la misma permita interactuar con las estructuras de datos almacenadas en los archivos.

Se deberá reutilizar la biblioteca del tp2 de la entidad pasajero en este tp, realizándole las modificaciones pertinentes para que funcione correctamente con los archivos y se utilice memoria dinámica.

Nota: Si el tp2 no fue entregado el tp3 no será evaluado en la primera instancia. Se corregirá junto con los recuperatorios.

El programa deberá tener el siguiente menú de opciones:

1. Cargar los datos de los pasajeros desde el archivo data.csv (modo texto).
2. Cargar los datos de los pasajeros desde el archivo data.csv (modo binario).
3. Alta de pasajero
4. Modificar datos de pasajero
5. Baja de pasajero
6. Listar pasajeros
7. Ordenar pasajeros
8. Guardar los datos de los pasajeros en el archivo data.csv (modo texto).
9. Guardar los datos de los pasajeros en el archivo data.csv (modo binario).
10. Salir

2 Aclaraciones

-Se debe contemplar que si no hay ningún pasajero cargado el id inicial debe ser 1. También se tendrá que asegurar que los ids no se dupliquen por más que hayan sido borrados.

-No es necesario que los archivos estén en memoria para que el programa funcione. El programa puede iniciar dando de alta pasajeros, luego cargar el archivo en memoria y mantener a los pasajeros cargados en un principio.

-No se puede salir del programa si es que **no** se realizó algún guardado de archivo.

-Deberán utilizar varias bibliotecas, el desarrollo de las funciones de pedidos de dato para la carga de pasajeros no pueden estar en la biblioteca correspondiente a la entidad pasajeros.

3 Biblioteca LinkedList

Las funciones del LinkedList que deben utilizarse como mínimo son las siguientes:

LinkedList* **ll_newLinkedList**(void) -> Crea y retorna un nuevo LinkedList. Es el constructor, ya que en él crearemos la struct y daremos valores iniciales a los campos.

`void ll_deleteLinkedList(LinkedList* self) -> Elimina el LinkedList`

`void ll_add(LinkedList* self , void* element) -> Agrega un elemento al final de LinkedList..`

`void ll_remove(LinkedList* self , int index) -> Elimina un elemento en LinkedList, en el índice especificado.`

`void* ll_get(LinkedList* self , int index) -> Retorna un puntero al elemento que se encuentra en el índice especificado.`

`int ll_len(LinkedList* self) -> Retorna el tamaño del LinkedList.`

4 Cómo realizar y entregar el trabajo práctico

El trabajo práctico deberá ser entregado en el repositorio de GIT correspondiente al TP3 de la materia.

El mismo consistirá en el proyecto de Eclipse con el programa funcionando y comentado, respetando las reglas de estilo de la cátedra. La compilación no deberá arrojar mensajes de error ni de warnings.

El proyecto deberá contar con la biblioteca LinkedList la cual se proveerá como una biblioteca estática, y se deberá hacer uso de dichas funciones para resolver la lógica del programa.

3.1 Objetivo

El objetivo del siguiente trabajo es que el alumno sea capaz de demostrar que puede integrar los conocimientos aprendidos durante la cursada en un caso real. Los conocimientos necesarios para la realización del TP son los siguientes:

- Manejo de punteros.
- Manejo de linkedList.
- Manejo de estructuras.
- Manejo de memoria dinámica.