

Estructuras de Datos y Algoritmos

Práctico de máquina 1 - Año 2023

Fecha de entrega: Jueves 31 de agosto de 2023 hasta las 8 hs.

La empresa de paquetería “*El Revoleo*” necesita un sistema capaz de manejar la información de los envíos realizados. La información que se mantiene sobre los mismos es: código de envío, que es único e identifica de forma unívoca toda la información asociada al mismo, D.N.I. del receptor, nombre y apellido del receptor, domicilio del receptor, D.N.I. del remitente, nombre y apellido del remitente, fecha en que se realizó el envío y la fecha en que se recibió. La empresa necesita poder realizar consultas sobre un envío dado, ingresar la información de nuevos envíos, modificar datos de un envío y eliminar los datos de aquellos que no se considere mantener.

Para almacenar la información planteada se utilizará la siguiente estructura:

- Lista Secuencial Ordenada con examinación secuencial (LSO).

Para solucionar las necesidades de la empresa deberá diseñar un programa en Lenguaje C que implemente los operadores necesarios para el manejo de la estructura propuesta.

La solución completa deberá presentar un menú que permita las siguientes operaciones: **ingreso de nuevos envíos, eliminar envíos existentes, modificar datos de un envío, consultar la información completa asociada al envío, Memorización Previa** y la opción **Mostrar Estructura**.

La opción **Mostrar Estructura** debe mostrar por pantalla todo el contenido de la estructura listando la información completa de los elementos presentes en ella, en el orden en que están almacenados.

Memorización Previa: Con esta opción deberá cargarse correctamente en la estructura el contenido del archivo de texto “*Envios.txt*” que contiene información de envíos y será provisto por la cátedra (disponible en la *página web de la materia*).

El archivo de texto “*Envios.txt*” contiene un dato de la nupla en cada línea (renglón) del mismo. Un ejemplo de esa información se muestra a continuación:

AD09912	/*código del primer envío*/
21695670	/*DNI receptor*/
Jorge Icaza	/*nombre y apellido receptor*/
4 DE JUNIO 100	/*domicilio receptor*/
28499429	/*DNI remitente*/
Jose Rondeau	/*nombre y apellido remitente*/
2020-03-03	/*fecha envío*/
2020-03-18	/*fecha recepción*/
.	.
.	.
AA07040	/*código del n-ésimo envío*/
32982224	/*DNI receptor*/
Jorge Luis Borges	/*nombre y apellido receptor*/
SARMIENTO 1335	/*domicilio receptor*/
24363665	/*DNI remitente*/
Jose de San Martin	/*nombre y apellido remitente*/
2020-03-04	/*fecha envío*/
2020-03-19	/*fecha recepción*/

Consideraciones a tener en cuenta:

- Se espera un máximo de 300 envíos.
- La confirmación de la baja de un elemento deberá realizarse por pantalla.



- El código del envío es una secuencia de 7 caracteres alfanuméricos.
- Los números de DNI son enteros.
- El campo nombre y apellido puede contener un máximo de 80 caracteres en cada caso.
- El domicilio puede contener un máximo de 80 caracteres.
- Las fechas de envío y recepción son una cadena de caracteres con el formato (AAAA-MM-DD).
- El ingreso de datos **no debe ser sensible a mayúsculas y minúsculas**, esto significa que al buscar un código de un envío deberá ser reconocido independientemente de cómo se ingresen las letras del mismo (aa12345 = Aa12345 = AA12345 = aA12345).
- El programa deberá desarrollarse en Lenguaje C, utilizando como entorno de desarrollo para tal fin **Code::Blocks** (disponible en www.codeblocks.org).

Ejemplo de rutina para Lectura de Datos

El código que se presenta a continuación es una guía para programar una rutina que permita leer datos desde un archivo de texto. **Deberá adaptarlo a la situación planteada.**

```
int Lectura_datos()
{
    .... //declaraciones
    FILE *fp;
    if (( fp = fopen ( "Envios.txt" , "r" ) )!=NULL)
        return 0;
    else {
        while (! (feof(fp))) {
            fscanf(fp, "%s", aux.codigo);
            fscanf(fp, "%d", &aux.dni_rec);
            fscanf(fp, "%s", aux.nombre_rec);
            fscanf(fp, "%s", aux.dom_rec);
            fscanf(fp, "%d", &aux.dni_env);
            fscanf(fp, "%s", aux.nombre_env);
            fscanf(fp, "%s", &aux.fecha_env);
            fscanf(fp, "%s", &aux.fecha_rec);
            //realizar el procesamiento del envío
            ....
        }
        fclose(fp);
        return 1;
    }
}
```

Importante:

- Los grupos deben ser de 2 integrantes.
- Los códigos fuente entregados que no compilen o estén incompletos respecto de la funcionalidad solicitada no serán revisados.
- La entrega del práctico se realiza por medio de la página de la materia y se debe enviar el archivo fuente del programa.
- El nombre del archivo deberá estar conformado de la siguiente manera: **PnroP-GruponroG** donde *nroP* es reemplazado por el número de práctico que se entrega y *nroG* por el número del grupo al que pertenece el programa. Por ejemplo, el nombre P1-Grupo22.c corresponde al práctico de máquina 1 enviado por el grupo 22. **Los programas cuyos nombres no respeten estas reglas de conformación no serán aceptados.**

