

IMPORTANTE:

- **Crear un proyecto con su Nombre y Apellido.**
- **Realizar todas las funciones que se indican.**
- **Añadir comentarios a su código identificando cada inciso.**

Como parte de una pasantía en la UTN FR Mar del Plata, se nos ha encomendado realizar un pequeño módulo para organizar información existente en un archivo de Estudiantes de la Tecnicatura Universitaria en Programación. El archivo consta de la siguiente estructura de datos:

```
typedef struct {  
    int legajo;  
    char nombre[20];  
    char apellido [20];  
    char dni[10];  
    int edad;  
    char genero;  
    int cantMateriasAprobadas; /// indica el total de materias aprobadas del estudiante en la carrera (max 20)  
    float promedioSinAplazos; /// indica el promedio de estudiante, sin tener en cuenta aplazos  
    float promedioConAplazos; /// indica el promedio de estudiante, teniendo en cuenta aplazos  
}estudiante;
```

A partir de dicha información, se administrará una lista simplemente vinculada con la siguiente estructura:

```
typedef struct _nodo{  
    estudiante dato;  
    struct _nodo* siguiente;  
} nodoEstudiante;
```

Contamos con el siguiente código desarrollado por un pasante que ya no se encuentra en el proyecto:

```
nodoEstudiante * inicListaEstudiante ()  
{  
    return NULL;  
}
```

Se nos pide desarrollar la siguiente funcionalidad:

1. Utilizando el archivo que se le entregó ("EstudiantesTUP.dat"), hacer una función que recorra el mismo de forma secuencial y construya una lista de estudiantes. Para realizar esto, deberá codificar las funciones:
 - a. Una función que **crea un *nodoEstudiante*, inicialice sus datos y retorne un puntero al mismo.**
 - b. Una función que agregue **un nuevo nodo al final de una la lista de forma recursiva.**
 - c. Un subprograma, que por cada registro en el archivo, cree un nodo y lo agregue a la lista. La función recibe la lista vacía y retorna la lista cargada.
2. Realizar una función que muestre la lista cargada previamente **de forma recursiva**. La misma deberá estar modularizada de acuerdo a la responsabilidad de cada estructura de datos.
3. Se desea **averiguar qué porcentaje de estudiantes** poseen un promedio superior a un valor determinado. Para lo cual deberá realizar:
 - a. Una función que cuente la cantidad de estudiantes que tiene la lista de **forma recursiva.**
 - b. Una función que averigüe, **de forma recursiva, que cantidad de estudiantes poseen un promedio con aplazos superior a un valor determinado.**
 - c. Un subprograma, que deberá invocar a las funciones anteriores (3.a y 3.b), **calcule el porcentaje** e informe por pantalla los resultados obtenidos.

UTN FRMDP - TUP - Laboratorio 2
Primer Parcial - 2 de octubre de 2020

4. Se quiere obtener a los 10 mejores promedios de la lista de estudiantes, con el fin de seleccionar a los mismos para un programa de ayudantías. Dichos estudiantes deberán alojarse en un **arreglo de estudiantes**. Para lo cual deberá realizar:
 - a. Hacer una función que busque al estudiante **de mayor promedio sin aplazos** en la lista y **retorna el legajo** del mismo.
 - b. Hacer una función que elimine el nodo de una lista que coincida con el **legajo** pasado por parámetro. **La función recibe un puntero doble a la lista y retorna una estructura de tipo estudiante.**
 - c. Un subprograma, que pase los estudiantes **de la lista a un arreglo de estudiantes, ordenándolos según el promedio sin aplazos (de mayor a menor)**. Para esto, invocará a las funciones anteriores (4.a y 4.b) y agregará en el arreglo el estudiante.
 5. Realizar una función que muestre el arreglo obtenido del punto 4, **en forma recursiva**. Esta función deberá mostrar el arreglo empezando **desde el promedio más bajo**.
 6. Se desea crear una **Pila (implementada con una lista de estudiantes)**, con aquellos estudiantes que **hayan aprobado más de 10 materias** de la carrera y que tengan un **promedio sin aplazos superior a 8 (ocho)**. Para lo cual deberá realizar:
 - a. **Las funciones necesarias para inicializar, agregar y mostrar la Pila de estudiantes (implementada con las funciones de listas que ya realizó, si necesita más, códifíquelas).**
 - b. Una función que **copiará de la lista principal a la Pila**, los estudiantes que cumplan con las condiciones antes mencionadas.
- Tenga en consideración la responsabilidad de cada estructura de datos trabajada. Realizar en el main las invocaciones que considere necesarias para probar el funcionamiento de este apartado.
7. Hacer una función main() que invoque a los modulos anteriores y demuestre el funcionamiento del programa.
 - a. Para hacer esto, cree las variables que considere necesarias e invoque las funciones (de forma directa o indirecta) como corresponde en cada caso.
 - b. Muestre los resultados cada vez que sea necesario.
 - c. Si lo considera, cree un menú de opciones para ejecutar cada función o subprograma.
 - d. A fin de identificar cada inciso, comente su código indicando a qué apartado corresponde, por ejemplo: // Apartado 3.a

El desarrollo de la función main es de carácter obligatorio y sumará puntos extras en caso de necesitarlos.

Apartado	1			2	3			4			5	6		7	
	a	b	c		a	b	c	a	b	c		a	b	main	compila
Puntaje	5	7,5	7,5	2,5	5	7,5	7,5	5	12,5	10	5	15	10	Si/No	Si/No

Tabla de puntuación:

Obtenido	10	20	30	40	50	60	70	80	90	100
Nota	1	2	3	4	5	6	7	8	9	10
Condición	Desaprobado					Aprobado				