

Universidad Tecnológica Nacional Facultad Regional Avellaneda										
Técnico Universitario en Programación - Técnico Universitario en Sistemas Informáticos										
Materia: Laboratorio de computación I										
Apellido:					Fecha:	26/11/2021				
Nombre:					Docente ⁽²⁾ :	Morel / Passucci				
División:	1°D				Nota ⁽²⁾ :					
Legajo					Firma ⁽²⁾ :					
Instancia ⁽¹⁾ :	PP		RPP		SP	X	RSP		FIN	

(1) Las instancias válidas son: 1^{er} Parcial (PP), Recuperatorio 1^{er} Parcial (RPP), 2^{do} Parcial (SP), Recuperatorio 2^{do} Parcial (RSP), Final (FIN) Marque con una cruz.

(2) Campos a ser completados por el docente.

Proyecto Eclipse y Github:

Crear un nuevo proyecto en Eclipse que se llame “SP_Labo1_ApellidoNombre”. Los commits se harán en un nuevo repositorio de GitHub llamado “labo1_SP_1D”.

NOTAS IMPORTANTES:

- El parcial es incremental, se deben realizar los puntos en orden. Hacer el punto 1 bien significa tener habilitado el paso para continuar con el punto 2, y así con los demás.
- De ser necesario, los profesores pueden solicitar la defensa oral del parcial.

Desarrollar en ANSI C:

Realizar un programa con un menú de opciones:

1. Leer de un archivo .csv los datos de los perritos y guardarlos de manera dinámica en un LinkedList. Para ello se deberá crear la entidad:

ePerrito:

- id (entero)
- nombre (cadena de caracteres con 21)
- peso (flotante)
- edad (entero)
- raza (cadena de caracteres con 21)

2. Listar a todos los perritos ordenados de manera ascendente por nombre.

NOTA: Utilizar constructores, getters y setters.

3. Calcular en un nuevo campo (de **ePerrito**) la “cantidadComidaRacion” (flotante) de cada uno de los perritos leídos. El criterio para calcularlo es: **peso * 23**

Para hacer este cálculo, se deberá desarrollar la función **ll_map** en la biblioteca LinkedList, la cual recibe la lista y una función.

```
int ll_map(LinkedList* this, int (*pFunc)(void*));
```

La función **ll_map** ejecutará la función recibida como parámetro por cada ítem de la lista, y le pasará, en cada llamada, uno de los ítems.

```
int ePerrito_laQueMapea(void*);
```

De esta manera, la función pasada como parámetro podrá realizar un cálculo con el ítem recibido, en este caso, calcular la ración de comida.

4. Listar los perritos con sus raciones de comida.

5. Una vez cargados los campos cantidadComidaRacion, generar un listado con los perritos que sean de raza “Galgo”, tengan más de 10 años de edad y coman menos de 200 gramos de comida diaria.

Para hacer esto, se deberá desarrollar la función **ll_filter** en la biblioteca LinkedList, la cual recibe la lista y una función.

```
LinkedList* ll_filter(LinkedList* this, int (*pFunc)(void*));
```

```
int ePerrito_laQueFiltra(void*);
```

6. Guardar en un archivo llamado “**galgosFlaquitos.csv**” los resultados de la búsqueda anterior (**formato texto**).