

Laboratorio 3

Electrónica IV

05/05/2025

Conceptos involucrados

- Patrón de programación ADT (Abstract Data Type)
-

Ejercicio a Desarrollar

Se deberá crear un repositorio git con acceso público que tenga el código fuente de un proyecto en C que cumpla con los siguientes requisitos:

- Se definirá un archivo de **makefile** para poder compilar el proyecto con el comando **make all**.
- Todo el código fuente (archivos **.c** y **.h**) deberá ubicarse en la carpeta **src**.
- El ejecutable deberá ubicarse en la carpeta **build/bin** y los archivos objeto intermedios en la carpeta **build/obj**.
- El programa principal estará compuesto solo por la función **main**, que se ubicará en el archivo **main.c**.
- El programa mostrará por pantalla los datos del alumno en formato JSON utilizando la función **serializar** del modulo **alumno**.
- La estructura de dato con los atributos de un alumno (nombre, apellido y documento) deberá ser privada del archivo **alumno.c**.
- En el archivo **alumno.h** se definirá una función **CrearAlumno** que recibirá como parámetros una cadena con el apellido, una cadena con el nombre y el documento del alumno a crear. La función retorna un puntero a la estructura de datos asignada para almacenar los datos del alumno, o el puntero nulo si no se pudo crear la estructura.
- Se deberá poder crear alumnos en forma estática o dinámica. Para esto se pueden plantear dos soluciones: dos funciones diferentes con los mismos parámetros o compilación condicional utilizando las funciones del preprocesador.
- En el archivo **alumno.h** se definirá una función **SerializarAlumno** que recibirá como parámetros un puntero a la estructura con los datos alumno, un puntero un vector de caracteres para generar el resultado y la cantidad de bytes disponibles en la cadena de resultado. La función retorna la longitud de la cadena generada, o -1 si el espacio no es suficiente.
- El modulo **alumno** tendrá dos función internas para serializar los campos de texto y numéricos. La función **serializar** utilizará estas funciones interna.
- Se documentará todo el código utilizando doxygen. El comando **make doc** deberá generar la documentación en HTML en la carpeta **build/doc**.
- Se deberá mantener un formato consistente en todo el código desarrollado (posición de llaves, formatos de identificadores y niveles de indentado).
- Se deberá evitar la repetición de código utilizando, adecuadamente, macros y archivos de cabecera.