



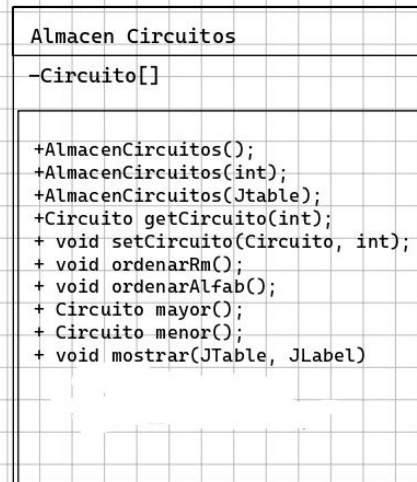
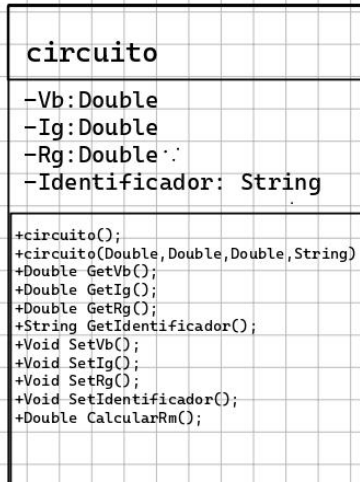
**TECNOLOGICO  
NACIONAL DE MÉXICO**

**Instituto Tecnológico de La Laguna  
Ingeniería en Sistemas Computacionales  
Programación Orientada a Objetos**

**Proyecto 01**

**Profesor:** Ing. Armando López Ciseña  
**Nombre del alumno:** Grecia Genesis Robles Picazo  
**Matrícula:** 25130266  
**Fecha:** 07/10/2025

## Diagramas de Clase del Proyecto 01



Circuito	Vb	Ig	Rg	Rm
X - 18	480	6	5	75
L - 25	312	7	8	37
A - 96	187	2	6	88
M - 74	240	4	3	57
G - 39	36	2	15	3
Suma/Prom	251	4.2	37	259

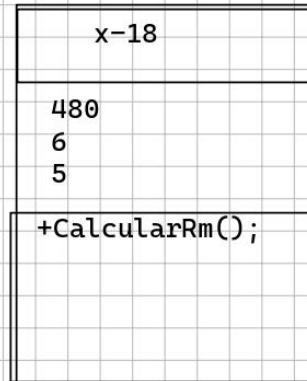
$$R_m = \frac{V_b - I_g R_g}{I_g}$$

Vb : 480

Ig : 6

Rg : 5

Rm=75



$$480 - (6 \times 5) / 6 = 75$$

## Descripción de las Clases

### Clase Circuito

Esta clase representa a cada circuito individual que se usa en la aplicación. Contiene los datos principales que identifican a un circuito, como su nombre o identificador, el voltaje máximo a medir (**Vb**), la corriente del medidor (**Ig**) y la resistencia interna (**Rg**). Además, esta clase tiene un método que realiza el cálculo de la **resistencia multiplicadora (Rm)** usando la fórmula establecida.

El propósito de esta clase es **encapsular los datos y el cálculo**, para que el programa no trabaje directamente con valores sueltos, sino con objetos que guardan toda la información de un circuito de forma ordenada y reutilizable.

### Clase AlmacenCircuitos

Esta clase se encarga de manejar **todos los circuitos juntos**, es decir, un conjunto o arreglo de objetos de la clase Circuito. Su función principal es **organizar, calcular y mostrar** la información completa dentro del JTable del programa.

Aquí es donde se incluyen los métodos para:

- Ordenar los circuitos de forma **alfabética o por Rm (descendente)**.
- Encontrar el circuito con el **mayor y menor valor de Rm**.
- **Mostrar los resultados** en la tabla de manera limpia y sin errores.

### Clase Frame Principal

Esta clase representa la **parte visual del programa**, es decir, la ventana donde el usuario interactúa. Aquí se colocan los botones, las etiquetas, el JTable y la imagen de asignación. Desde esta ventana se pueden ejecutar todas las acciones principales:

- Cargar datos de ejemplo.
- Generar valores aleatorios.
- Calcular los resultados.
- Ordenar la información.
- Borrar los datos.

El Frame sirve como **conexión entre el usuario y la lógica del programa**, ya que los botones llaman a los métodos de las otras clases (Circuito y AlmacenCircuitos) para realizar los cálculos y mostrar los resultados en pantalla.

### **Modificación del constructor AlmacenCircuitos y el método Mostrar() de AlmacenCircuitos:**

En el **constructor de la clase AlmacenCircuitos** se crea el arreglo donde se van a guardar los objetos de tipo Circuito. Este constructor recibe como parámetro la cantidad de circuitos que se van a manejar, y con eso se define el tamaño del arreglo. Así, desde que se crea el objeto de AlmacenCircuitos, ya se reserva el espacio necesario para guardar los circuitos que se van a usar más adelante.

Por otro lado, el **método mostrar()** sirve para mostrar los datos de los circuitos en la tabla (JTable). En este método se agregaron validaciones que antes no teníamos en clase, para evitar errores cuando la tabla tiene celdas vacías o cuando no todas las filas están llenas.

Antes, se usaba el **getRowCount()** para recorrer las filas, pero descubrí que eso solo cuenta cuántas filas hay, no si realmente tienen información, por eso, si alguna celda estaba vacía o incompleta, el programa intentaba leer algo que no existía y marcaba error o mostraba datos incorrectos.

Para arreglar eso, se agregaron validaciones que revisan si las celdas tienen datos antes de convertir sus valores. De esta forma, solo se toman en cuenta las filas que están completas y se ignoran las vacías. Además, se evita que el programa se trabe por intentar leer una celda sin valor.

En pocas palabras, esas validaciones se pusieron para que el método **mostrar()** funcione mejor y no falle cuando el usuario deja espacios vacíos o la tabla no está completamente llena. Así los datos que se muestran son correctos y el programa es más estable.