

# Diagrama estructural final

Graph
Hash nodos
<ul style="list-style-type: none"><li>- initialize()</li><li>- add()</li><li>- delete()</li><li>- node(key)</li><li>- nodes()</li><li>- inspect()</li></ul>

Node
key Set in_edges Set out_edges Hash properties
<ul style="list-style-type: none"><li>- initialize(key,properties)</li><li>- inspect()</li><li>- connect_to()</li><li>- disconnect_from()</li><li>- edges()</li><li>- nodes()</li><li>- connect_endpoint()</li><li>- properties</li><li>- set(key, value)</li><li>- get (key)</li></ul>

Edge
Node from-node Node to_node Hash properties
<ul style="list-style-type: none"><li>- inspect()</li><li>- to_s()</li><li>- properties()</li></ul>

# Proyecto Estructura de Datos

Proyecto de Circuitos - Estructura de datos

Gabriel González Mañanán

2007-6542

Profesor Rodrigo Orizondo

# Estructura de datos

El circuito está representado a través de un grafo el cual tendrá sus respectivos métodos que harán posible insertar nodos, eliminar nodos, buscar nodos, contar cantidad de nodos y retornar de manera humanamente-leíble el grafo completo.

La estructura principal será la clase Grafo, la cual contendrá los métodos de inicialización e impresión del mismo, conjuntamente con las clases:

- Nodos: La clase que contendrá los elementos del circuito.
- Aristas: Una clase que servirá como la representación de una conexión de un nodo a un nodo.

# Diagrama estructural inicial

Grafo
<ul style="list-style-type: none"><li>- add(node)</li><li>- delete(node)</li><li>- find(node)</li><li>- ver()</li></ul>

Nodo
<ul style="list-style-type: none"><li>- initialize(id,impedancia,propiedades)</li><li>- conectados_a()</li><li>- conectados_de()</li><li>- descendientes()</li><li>- ancestros()</li><li>- set_propiedad()</li><li>- get_propiedad()</li><li>- ver()</li></ul>

Vertice
<ul style="list-style-type: none"><li>- initialize(desde,hasta,propiedades)</li><li>- direccion</li><li>- ver()</li></ul>

# Implementación

Para la realización de este proyecto escogí Ruby como lenguaje para la implementación.

El proyecto consta de 4 archivos “.rb”, 3 archivos correspondientes a las tres clases previamente explicadas y archivo de prueba (test.rb) que instancia los objetos y demuestra las diferentes funcionalidades de creación del grafo (circuito)