

Primer Proyecto de Algoritmos y Estructuras de Datos 2, Primer semestre 2021

Integrantes:

Granados Chacón Josué Guillermo

Rojas Diaz Yordan Andrés

## **Breve descripción del problema**

Durante este proyecto se desarrollará una aplicación que se asemeje a un lenguaje de programación. Este pseudo lenguaje tendrá el nombre de C! y su sintaxis derivará del lenguaje C, esta aplicación tendrá una interfaz en la que se podrá visualizar los valores solicitados por el código, un log para el código desarrollado en la aplicación y el estado actual de la memoria RAM.

En este pseudo lenguaje se contará con cierto tipo de datos, como es el caso de int, float, long, char, con cada tipo teniendo un cierto número de bytes que se les asignara información. Estos datos se manejarán como copias, menos el caso de las struct.

Las variables para C! se crearán en un servidor conectado por sockets a la interfaz. Este servidor recibirá instrucciones en formato json de parte de la interfaz y mediante estas asignará la cantidad de bytes correspondiente a cada una de las peticiones de la interfaz.

Ambos, el servidor y la interfaz serán programados en C++

## **Descripción de las estructuras de datos utilizadas**

Linked List

Es una estructura lineal de datos, cuyos elementos no están guardados en espacios continuos de la memoria. Estos elementos están conectados al utilizar pointers

# **Descripción de los algoritmos desarrollados**

## **Conexión Servidor-Cliente**

El servidor es programado en c++. Utiliza la qt para crear un servidor que va a escuchar en un puerto predefinido, este servidor al escuchar una solicitud de conexión, la acepta para crear un socket. Para el cliente, de igual manera es creado mediante qt y su sistema de servers, este envía y recibe los datos enviados por el server.

## **Generar Json**

Mediante el uso de las herramientas ofrecidas por qt para el uso de json, el sistema lee línea por línea las peticiones para crear una variable escritas por el usuario y las escribe en formato json para que sean recibidas por el servidor.

## **Asignar espacio de memoria**

Al recibir un archivo json, el servidor creará nuevas variables dentro de un espacio de la memoria preestablecido, el tamaño de la variable será establecido por el archivo json al igual que su tipo. Después de esto el servidor enviará un pointer para que sea asignado por la interfaz.

# **Problemas encontrados**

## **Leer el código creado por el usuario línea por línea**

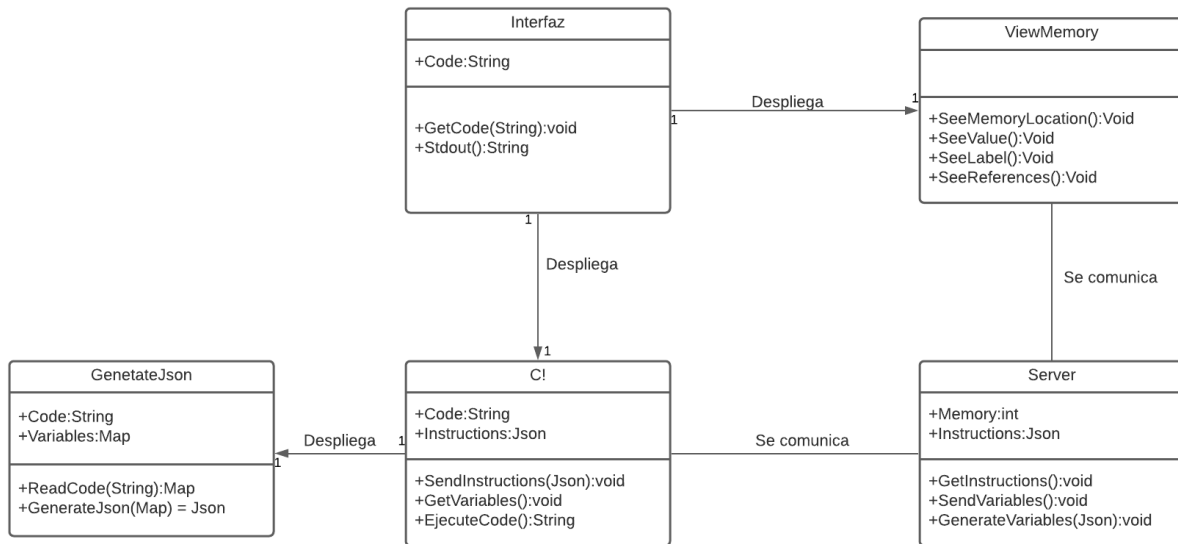
### **Mandar la variable creada por los sockets**

Se intento mandar la dirección de memoria del servidor a la interfaz, pero los pointers no señalan al lugar deseado

## **Generar archivo json en C++**

No se encuentra una librería para generar Json en c++. La encontrada fue jsoncpp, pero esta no se pudo descargar debido a un error que esta posee.

# Diagrama de Clases



**Enlace al Repositorio Github:** <https://github.com/YordanRD/Proyecto1Datos2.git>

**Enlace al Jira:**

<https://datos2.atlassian.net/secure/RapidBoard.jspa?rapidView=1&projectKey=P1D2&selectedIssue=P1D2-4#>