

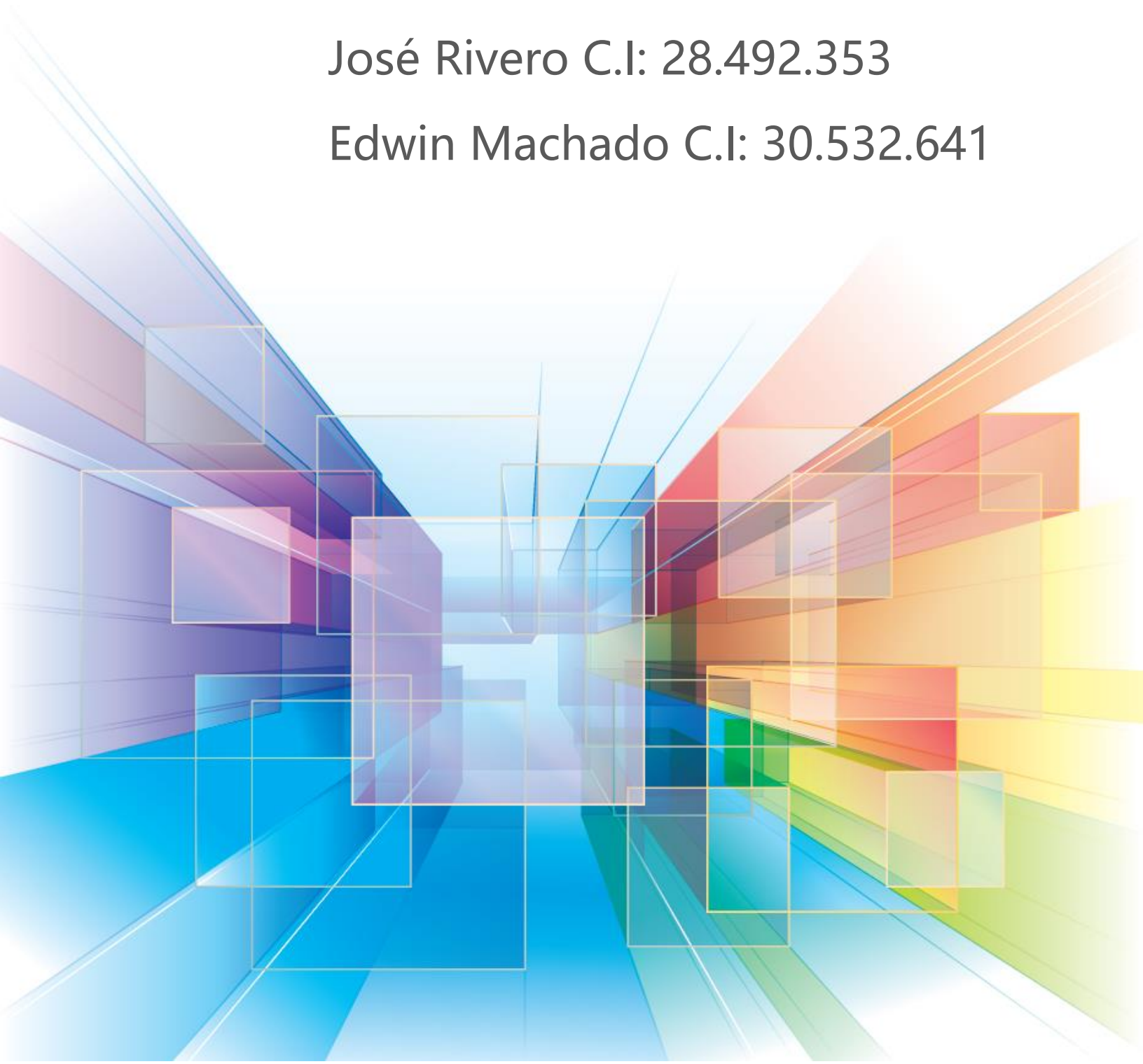
# Informe 2

Arquitectura del computador

Por:

José Rivero C.I: 28.492.353

Edwin Machado C.I: 30.532.641



## ¿Qué es el paralelismo?

El paralelismo es una forma de computación en la que múltiples cálculos pueden llevarse a cabo simultáneamente. Este enfoque se basa en la idea de dividir problemas grandes en problemas más pequeños que se resuelven de manera simultánea. Existen varios tipos de paralelismo, incluyendo el de nivel de bit, el de nivel de instrucción, el de datos y el de tarea. El paralelismo ha sido utilizado durante muchos años, especialmente en la computación de alto rendimiento.

## CUDA

CUDA son las siglas de Compute Unified Device Architecture (Arquitectura Unificada de Dispositivos de Cómputo), y se refiere a una plataforma de computación en paralelo desarrollada por Nvidia. Esta plataforma incluye un compilador y un conjunto de herramientas de desarrollo que permiten a los programadores utilizar una variación del lenguaje de programación C, conocida como CUDA C, para codificar algoritmos específicamente diseñados para ejecutarse en las unidades de procesamiento gráfico (GPU) de Nvidia.

CUDA busca aprovechar las ventajas de las unidades de procesamiento gráfico (GPU) en comparación con las unidades de procesamiento central (CPU) de propósito general. Se logra mediante la explotación del paralelismo inherente a los múltiples núcleos de las GPU, lo que permite el lanzamiento de un gran número de hilos simultáneos. Cuando una aplicación está diseñada con numerosos hilos que realizan tareas independientes, como ocurre naturalmente en el procesamiento gráfico, la GPU puede ofrecer un rendimiento destacado en campos que van desde la biología computacional hasta la criptografía, entre otros ejemplos.

## OPEN MP

OpenMP es una interfaz de programación de aplicaciones (API) diseñada para la programación multiproceso de memoria compartida en diversas plataformas. Esta interfaz posibilita la introducción de concurrencia en programas escritos en C, C++ y Fortran, basándose en el modelo de ejecución fork-join. OpenMP está disponible en varias arquitecturas, incluyendo plataformas Unix y Microsoft Windows. Se compone de directivas de compilador, rutinas de biblioteca y variables de entorno que afectan el comportamiento durante la ejecución.

En esencia, OpenMP sirve como un modelo de programación paralela para sistemas de memoria compartida y memoria distribuida en multiprocesadores. Este modelo se fundamenta en la disposición de múltiples hilos de ejecución concurrentes que pueden acceder a variables almacenadas en áreas de

memoria a las que cada hilo tiene acceso. OpenMP simplifica la programación al evitar el intercambio explícito de datos entre distintos procesadores mediante el uso de directivas de compilación.

## **MPI (Interfaz de paso de mensajes)**

La Interfaz de Paso de Mensajes (MPI), acrónimo de Message Passing Interface en inglés, es un estándar que establece la sintaxis y la semántica de las funciones de una biblioteca de paso de mensajes. Esta biblioteca está diseñada para ser utilizada en programas que aprovechan la existencia de múltiples procesadores y suele emplearse en lenguajes como C, C++, Fortran y Ada. MPI es un "estándar por consenso" creado en un foro abierto que incluía a proveedores de hardware, investigadores, académicos, desarrolladores de bibliotecas de software y usuarios, representando a más de 40 organizaciones.

Aunque MPI no es un estándar emitido por una organización como ISO o IEEE, es ampliamente utilizado debido a su desarrollo colaborativo y al hecho de ser de código libre. La ventaja clave de MPI radica en la portabilidad de los programas que utilizan esta biblioteca, ya que ha sido implementada para casi todas las arquitecturas de memoria distribuida. Además, ofrece rapidez, ya que cada implementación de la biblioteca ha sido optimizada para el hardware en el que se ejecuta.