

# Computación Distribuida

Semestre 2023-1.

Práctica 1: Consenso

Erick Arroyo 318163790

## Descripción:

El programa esta desarrollado en lenguaje **C**, utilizando la libreria OpenMpi. El propósito del programa es representar el comportamiento del algoritmo de consenso de dos etapas. El programa se comporta como sigue: El usuario indica el número de procesadores (nodos) que se ejecutarán, a partir de ello el programa determina que una tercera parte de los procesadores serán considerados traidores (cuyo comportamiento indica el planteamiento del algoritmo). En una primer instancia el programa determina la lealtad de cada comandante para después mandar su decisión al resto de comandantes. Aclarando, los comandantes leales mandan la misma desición a el resto de los comandantes, y los traidores mandan distintas desiciones. Eso para la primer etapa. Para la segunda, el comportamiento es similar, es decir, los comandantes leales mandan las decisiones del resto de comandantes, tal y como las recibieron, y los traidores corrompen esas listas de decisiones. Para determinar la decisión mayoritaria, cada comandante determina la decisión del resto de ellos, a partir de la información dada en los arreglos. Por último, el programa sigue el esquema de decisión dado por el profesor. section\*Instalación:

## Prerequisitos:

Tener instalado la versión **+4.0.3** de Open MPI

## Descarga:

Descargar archivo **Practica01\_Arroyo\_Erick.zip**, recuperado de: <https://classroom.google.com/u/0/c/NTI2MTY1NjcxMzAw/a/NTI3NjA4NjQ5NzY5/details?hl=es>

## Ejecución:

**Paso 1:** Descomprimir el archivo **Practica01\_Arroyo\_Erick.zip** en el directorio deseado.

**Paso 2:** Acceder hasta el directorio **src** del archivo descomprimido, desde el simbolo del sistema.

**Paso 3:** Ejecutar el comando **mpicc -o consenso Consenso.c**, seguido del comando **mpirun -np [#nodos] -oversubscribe ./Consenso**.

## Bug's:

En caso de no tener un buen funcionamiento al ejecutar el proyecto, favor de mandar un correo con una explicación del bug, versiones de su entorno y los pasos para replicar el error, a alguna de las siguientes direcciones: [erickarroyo@ciencias.unam.mx](mailto:erickarroyo@ciencias.unam.mx).

## Observaciones:

El código esta documentado, por lo que se pretende que eso baste para el entendimiento del mismo. Sin embargo, solo aclaremos como se determina el número de traidores. Entonces, esto se hace a partir del número total de comandantes, es decir, uno de cada tres comandantes será designado como traidor, y a su vez, en número total de nodos es determinado por el usuario por medio del comando: **mpirun -np [numero\_de\_nodos] -oversubscribe ./Consenso**