

Profesor: Luis Germán Pérez Hernández

Ayudante: Daniel Michel Tavera

Laboratorio: Isabel Espino Gutiérrez

**Fecha de entrega: jueves 18 de abril de 2024**

### **Descripción de la Práctica:**

El alumno deberá implementar el 'Algoritmo del Rey' visto en clase como sigue:

- Cada nodo determina de forma aleatoria su plan inicial (ataque o retirada).
- Un subconjunto de nodos simulara ser traidores, enviando información arbitraria al momento de la comunicación (solamente se simulan fallos bizantinos, no fallos por caída).
- Se realizan rondas de comunicación hasta que se alcance un consenso o transcurra un número máximo de rondas sin lograrlo.
- Cada nodo reporta el resultado, indicando también si es leal o traidor.

Queda a discreción del alumno la forma de determinar cuántos y cuáles nodos serán traidores, además de establecer los parámetros del algoritmo:

- Número máximo de rondas antes de declarar un fracaso.
- Número máximo de traidores a tolerar (y, por tanto, margen de votos necesarios para tomar la mayoría como válida)
- Etc.

Finalmente, también se deberá establecer la manera de elegir al rey después de cada ronda en la que no se alcance consenso.

Es muy importante asegurar que todos los nodos lleguen a la misma conclusión (consideren al mismo nodo como rey) para que el algoritmo funcione.

**Requisitos de entrega:**

Guardar el código fuente del programa en un archivo con el nombre

*“Práctica2NombreApellido”*

(por ejemplo Practica2DanielMichel.c)

Realizar también un reporte en pdf con el mismo nombre, en el cual se indique lo siguiente:

- Una descripción de cómo se desarrolló la práctica y cómo funciona la solución implementada. Si su solución consiste de más de un archivo, mencionarlo aquí.
- La forma de compilar y operar el programa, incluyendo qué entradas se esperan del usuario y qué salidas arroja.
- Cualquier otro comentario o aclaración que consideren pertinente.

Subir los archivos al Classroom antes de las 23:59 horas de la fecha de entrega.

NO se recibirán entregas después de esa fecha.