

**Facultad de Ciencias, UNAM**

**Materia:** Computación Distribuida

**Profesor:** Mauricio Riva Palacio Orozco

**Ayudantes:** Alan Alexis Martínez López y Adrián Felipe Fernández Romero

**Práctica 6 - Detección de Fallas en Sistemas Distribuidos**

**Descripción:**

Los detectores de fallas son una herramienta esencial en los sistemas distribuidos, ya que permiten mantener la integridad del sistema cuando uno o varios procesos dejan de responder.

En esta práctica implementarán un detector de fallas sencillo, basado en el siguiente esquema, cuyo propósito es identificar procesos que pueden haber fallado durante la ejecución del algoritmo.

```
(1) init:  $suspected_i \leftarrow \emptyset$ .  
  
(2) repeat for  $\beta$  time units  
(3)   for each  $j \notin suspected_i$  do send INQUIRY( $i$ ) to  $p_j$  end for;  
(4)    $crashed_i[1..n] \leftarrow [true, \dots, true]$ ;  
(5)   set  $timer_i$  to  $\Delta$   
(6) end repeat.  
  
(7) when INQUIRY( $j$ ) is received do send ECHO( $i$ ) to  $p_j$ .  
  
(8) when ECHO( $j$ ) is received do  $crashed_i[j] \leftarrow false$ .  
  
(9) when  $timer_i$  expires do  $suspected_i \leftarrow \{x \mid crashed_i[x]\}$ .
```

## Facultad de Ciencias, UNAM

**Materia:** Computación Distribuida

**Profesor:** Mauricio Riva Palacio Orozco

**Ayudantes:** Alan Alexis Martínez López y Adrián Felipe Fernández Romero

### Práctica 6 - Detección de Fallas en Sistemas Distribuidos

#### Desarrollo

Para esta práctica deberán extender el trabajo realizado en la Práctica 4 (algoritmo de consenso).

Su tarea consiste en implementar una función que utilice el detector de fallas previamente descrito y, además, modificar su función de consenso para que se apoye en dicho detector durante su ejecución.

Cada proceso deberá contar con dos nuevas variables:

- **Suspected<sub>i</sub>**: mantiene el conjunto de procesos que el proceso  $i$  sospecha que han fallado.
- **Crashed<sub>i</sub>**: indica qué procesos se consideran caídos en la ronda actual.

En esta práctica no habrá pruebas automáticas, ya que el comportamiento depende directamente del consenso de la Práctica 4. Sin embargo, deberán asegurarse de que al finalizar la ejecución cada proceso imprima su variable **suspected<sub>i</sub>**.

Si la implementación es correcta, esta variable deberá coincidir en todos los procesos correctos (aquellos que completan la ejecución y alcanzan el consenso).

#### Preguntas

Responde las siguientes preguntas y justifica tus respuestas (excepto en la última):

1. ¿A qué clase pertenece este detector de fallos?
2. ¿Qué cambios tendrías que realizar para que el detector pudiera ejecutarse indefinidamente cada  $k$  rondas?
3. ¿Tienes alguna sugerencia, queja o comentario sobre el laboratorio?  
Este es el momento para expresarlo (no habrá represalias).

**Facultad de Ciencias, UNAM**

**Materia:** Computación Distribuida

**Profesor:** Mauricio Riva Palacio Orozco

**Ayudantes:** Alan Alexis Martínez López y Adrián Felipe Fernández Romero

**Práctica 6 - Detección de Fallas en Sistemas Distribuidos**

**Lineamientos de entrega**

Dentro de la entrega deberán incluir un archivo **README** que contenga:

- El número y nombre de la práctica: **Práctica 6**
- Los nombres de los integrantes y sus números de cuenta
- Una breve descripción de la implementación del detector de fallas y de las modificaciones realizadas al algoritmo de consenso

Solo **un integrante** deberá subir la práctica. El otro integrante únicamente deberá marcarla como entregada antes de la fecha límite.

El archivo final deberá subirse **comprimido en formato .zip** con el siguiente nombre:

**Practica6-ApellidoPaterno1ApellidoMaterno1-ApellidoPaterno2ApellidoMaterno2**