

Computacion Distribuida

Readme 03

Para el 10 noviembre 2020

Profesora: Karla Rocio Vargas Rodriguez

Francisco Daniel Cruz Torres
eragond@ciencias.unam.mx
316194099

Uriel Garcia Luna Bobadilla
uriel_06@ciencias.unam.mx
316641902

10 noviembre 2020

Para esta práctica debíamos implementar con `simpy` el siguiente algoritmo de consenso:

```

Function Consensus( $v_i$ )
 $V_i \leftarrow [\perp, \dots, v_i, \dots, \perp]$ ;  $New_i \leftarrow \{(v_i, i)\}$ ;
when  $r = 1, 2, \dots, f + 1$  do %  $r$ : round number %
  begin_round
    ( $\alpha$ ) if ( $New_i \neq \emptyset$ ) then foreach  $j \neq i$ : send ( $New_i$ ) to  $p_j$  endif;
    let  $rec\_from[j]$  = set received from  $p_j$  during  $r$  ( $\emptyset$  if no msg);
     $New_i \leftarrow \emptyset$ ;
    foreach  $j \neq i$ : foreach  $(v, k) \in rec\_from[j]$ :
    ( $\beta$ ) if ( $V_i[k] = \perp$ ) then
       $V_i[k] \leftarrow v$ ;  $New_i \leftarrow New_i \cup \{(v, k)\}$  endif
    end_round;
    let  $v$  = first non- $\perp$  value of  $V_i$ ;
    return ( $v$ )

```

La manera de hacerlo fue como en practicas pasadas con nuestra interfaz `Nodo`, esta implementación de nodo respetamos los atributos que nos brindaron los cuales incluían todos los que vienen en el algoritmo y dos nuevos: un booleano que indica si el nodo falla y un atributo líder en el cual guardaremos el valor acordado después del consenso.

Las observaciones que consideramos más importantes en nuestra implementación es que los procesos que fallen lo harán desde el principio y no mandaràn nada a ningún nodo, bajo esto planteamos la lógica de en cada ronda recibir nuestro *rec_form* ya que dentro de nuestra ronda actual hacemos un while hasta que recibamos $n - f$ mensajes, donde n son el número de vecinos que tenemos y f los procesos que fallan ya que como los procesos que fallan siempre lo hacen en la primera ronda podemos asegurar que recibiremos mensajes de los demás procesos, de igual manera como todos los procesos mandan mensaje cuando *New* sea vacío significará que todos los nodos han mandado su mensaje y reenviado cuando reciben uno nuevo por lo cual terminaremos la ronda