

Universidad Nacional Autónoma de México
Facultad de Ciencias
Computación Distribuida
README

☁ García Santamaría José Luis 316174646
jose_luis@ciencias.unam.mx

☺ Reyes Martínez Antonio 316184931
antonioreyes21@ciencias.unam.mx

12 de noviembre de 2020

Todos nuestros archivos se encuentran en la carpeta **src**, dentro de ésta encontramos el archivo para el algoritmo NodoConsenso, también en la carpeta **/src/Canales** también encontramos la clase CanalRecorridos.py, por otro lado cabe mencionar que la especificación de la implementación de nuestra función *consenso* está comentada en el código como documentación.

En general, para correr nuestra práctica es necesario realizar desde la terminal (una vez situados en la carpeta **src**) los siguientes comandos:

python3 -i "Nombre del nodo".py
y una vez dentro del interprete poner el comando
envi.run()

En general podemos asegurar que implementamos 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 ( $v$ )
```

Figure 1. A Protocol for Crash Failures ($f < n$)