



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

## Tarea 2

### INTEGRANTES

Torres Valencia Kevin Jair - 318331818

Aguilera Moreno Adrián - 421005200

iiiiiii HEAD Natalia Abigail Pérez Romero - 31814426

===== Natalia Abigail Pérez Romero - 318144265

llllllll 8b75eb6b2d65790e47fe8ab8cbe88bfec584c480

### PROFESOR

Miguel Ángel Piña Avelino

### AYUDANTE

Pablo Gerardo González López

### ASIGNATURA

Computación Distribuida

12 de septiembre de 2022

iiiiiii HEAD

1. (1 puntos) Considera el algoritmo de Flooding visto en clase. Demuestra el siguiente corolario: Todo proceso  $p_i$  que ejecuta el algoritmo de **Flooding**, recibe  $M$  en tiempo a lo más el diámetro  $diam(G)$  de la gráfica del sistema distribuido.

=====

1. `LLLLLL 8b75eb6b2d65790e47fe8ab8cbe88bfec584c480 HHHHHH HEAD`

2. (1 punto) Considera el algoritmo de BroadcastTree visto en clase. ¿Cuál sería el peor caso en complejidad de tiempo para el algoritmo BroadcastTree? Explica detalladamente.

=====

2. `8b75eb6b2d65790e47fe8ab8cbe88bfec584c480` `HEAD`

3. (2 puntos) Considera el algoritmo de BroadConvergecast visto en clase.

1. Prueba el siguiente lema: ¡¡Todo proceso  $p_i$  a profundidad  $D$ , recibe el mensaje  $\langle START \rangle$  en tiempo  $D$ !!.
2. Prueba el siguiente lema: ¡¡Todo proceso  $p_i$  a profundidad  $D$  envía su mensaje a la raíz en el tiempo  $D + 2 * altura(p)$ !!

=====

3. `8b75eb6b2d65790e47fe8ab8cbe88bfec584c480` `HEAD`

4. (1 punto) ¿Se basan los algoritmos de BroadcastTree y ConvergeCast en el conocimiento acerca del número de nodos en el sistema? ¿Por qué?

=====



4. `8b75eb6b2d65790e47fe8ab8cbe88bfec584c480` `HEAD`

5. (3 puntos) Generaliza el algoritmo de Broadconvergecast para:

1. Construya un árbol generador, es decir, inicialmente cada proceso tendrá sus variables  $PADRE = \perp$  e  $HIJOS = \emptyset$  y conforme el algoritmo vaya avanzando en el número de rondas, esas variables se vayan actualizando. El proceso raíz (distinguido) debe conocer el momento en que se terminó de construir este árbol generador.
2. Suponga que cada proceso tiene una entrada distinta para reportar algo (pueden ser información de sensores, lecturas, etc.) A partir del algoritmo anterior, indica las modificaciones que se tendrían que hacer en el algoritmo, para que se recolecte la información de estos procesos y la raíz tenga todas la entradas. Analiza la complejidad en bits, es decir, el total de bits que son enviados sobre los canales de comunicación

(hint: Cada mensaje de información puede tomar  $k$  bits)

=====

5. `8b75eb6b2d65790e47fe8ab8cbe88bfec584c480` `HEAD`

6. (2 puntos) Da un algoritmo distribuido para contar el número de procesos en cada capa de un árbol enraizado  $T$  de forma separada. Al final la raíz reportará el número de procesos por capa. Analiza la complejidad de tiempo y la complejidad de mensajes de tu algoritmo. =====

6. `8b75eb6b2d65790e47fe8ab8cbe88bfec584c480`