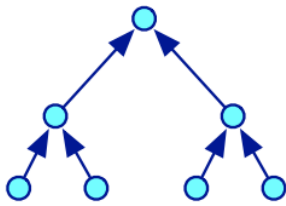


Instrucciones: (1) Coloque su nombre, paralelo y número de matrícula en el espacio indicado, (2) Lea cada pregunta detenidamente, (3) Conteste claramente cada pregunta.
 NOTA: NO se permite el uso de celulares.

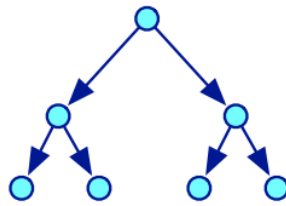
Nombre, paralelo y matrícula: _____

Pregunta	1	2	3	4	Total:
Puntos	6	4	7	9	26
Puntaje					

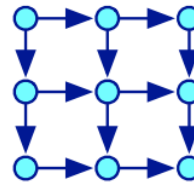
1. Grafos de dependencias de tareas. Considere los cuatro grafos de dependencias de tareas mostrados a continuación. Asuma que a cada tarea le toma una unidad de tiempo en ejecutarse y que el costo de comunicación entre las tareas es cero (0). Para cada grafo, conteste las preguntas planteadas abajo.



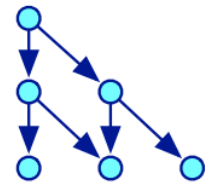
(A)



(B)



(C)



(D)

(a) (2 pts) ¿Cuál es la longitud de la ruta crítica?

(A)

(B)

(C)

(D)

(b) (4 pts) ¿Cuál es el máximo grado de concurrencia?

(A)

(B)

(C)

(D)

2. MapReduce, Hadoop, Spark, Kafka, Big Data

- (a) (4 pts) ¿Cómo hacen sistemas como MapReduce, GFS, Spark o Kafka para saber que un nodo que forma parte del sistema ha fallado? ¿Cuál es la limitación de este mecanismo?

3. Comunicación entre procesos

(a) (3 pts) ¿Qué hace un middleware RMI o RPC para que las llamadas o invocaciones remotas parezcan locales?

(b) (4 pts) El middleware RPC utilizado por el Network File System (NFS) utiliza semánticas de invocación “al-menos-una-vez”. ¿Qué puede hacer un servidor NFS para sobreponerse a esta debilidad del middleware? Nota: No es necesario que conozca ningún detalle de la implementación de NFS para contestar esta pregunta.

4. Confiabilidad y tolerancia a fallos

(a) (9 pts) La mayor parte de los mecanismos de tolerancia a fallos utilizan algún tipo de *redundancia*. Escoja al menos tres (3) de los siguientes mecanismos de tolerancia a fallas, y explique dónde/cómo tiene redundancia.

- Ejecución especulativa (también llamadas, tareas de backup) en MapReduce
- Linaje en Spark
- Replicación primario-secundario en bases de datos RDBMS
- Confiabilidad de TCP
- Confiabilidad en Wireless Sensor Networks (WSNs)