



# CUUDS

NOMBRE DEL ALUMNO: José Carlos Toledo Pérez

NOMBRE DEL PROFESOR: Andrés Alejandro Reyes Molina

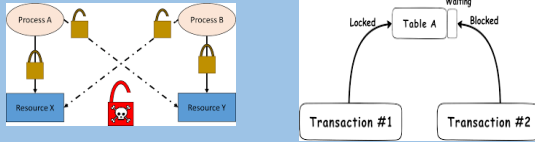
MATERIA: Base de datos II

CUATRIMESTRE: 8

TIPO DE TRABAJO: Super nota

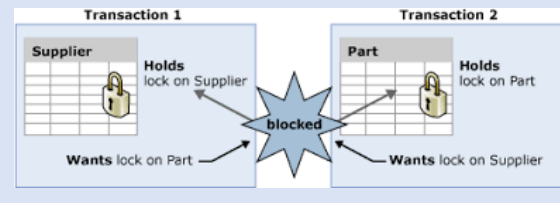
## 2.4 BLOQUEOS

Un bloqueo en general es cuando una acción que debe ser realizada está esperando a un evento. Para manejar los bloqueos hay distintos acercamientos: prevención, detección, y recuperación.



### 2.4.1 BLOQUEO MORTAL

En sistemas operativos, el bloqueo mutuo (también conocido como interbloqueo, traba mortal, deadlock, abrazo mortal) es el bloqueo permanente de un conjunto de procesos o hilos de ejecución en un sistema concurrente que compiten por recursos del sistema o bien se comunican entre ellos.



### 2.4.2 SOLUCIONES

El control de concurrencia y detección y manejo de bloqueos es un área de mucho estudio en las bases de datos distribuidas, a pesar de esto no hay ningún algoritmo aceptado para solucionar el problema.



### 2.4.3 BLOQUEO DE DOS FASES (2PL)

Antes de implementar un algoritmo de control de concurrencia 2PL es necesario considerar distintos factores como cuál es la unidad atómica más pequeña que el sistema permite bloquear, cual es el intervalo de sincronización para todas las copias de un elemento, donde se deben colocar las tablas con la información de los bloqueos y por último que tan probable es que ocurra por los factores anteriores un bloqueo mutuo.

### 2.4.4 TIME-STAMP

Cada transacción realizada se le asigna un timestamp (literalmente: sello de tiempo) único en el nodo que se originó. Este sello se adjunta a cada petición de lectura y escritura. En el caso de que se dé un conflicto de que dos operaciones de escritura traten de acceder al mismo elemento, este se resuelve serializandolo respecto a los sellos que tengan. A pesar de que existen varios algoritmos de control de concurrencia basados en timestamps, muy pocos son utilizados en aplicaciones comerciales.

### 2.5 SERIABILIDAD

Una forma de evitar los problemas de interferencia es no permitir que las transacciones se intercalen. Una ejecución en la cual ninguna de dos transacciones se intercala se conoce como serial. Para ser más precisos, una ejecución se dice que es serial si, para todo par de transacciones, todas las operaciones de una transacción se ejecutan antes de cualquier operación de la otra transacción.

### 2.6 DEAD LOCK

El deadlock, conocido también como abrazo mortal o bloqueo mutuo en programación, se refiere a la situación en la que dos o más tareas, como procesos o hilos, se pausan esperándose la una a la otra para poder llevar a cabo la continuación o finalización de su trabajo.

### 2.6.1 CAUSAS DEL DEADLOCK

Dentro de las características del deadlock en programación, se encuentra que este puede presentarse por diversas situaciones, como puede ser que varios procesos compitan entre sí por la asignación de uno o más recursos determinados. Esta destaca como la causa más común de los bloqueos mutuos.