

TOMO LO QUE TIENE PUNTO (4 PREGUNTAS)

UNIDAD 1

OBJETIVOS \Rightarrow ABSTRACCIÓN COMPLEJIDAD Y ADMINISTRAR / PROTEGER RECURSOS DE PC
 ESTADOS DE LOS PROCESOS (Planificadores) \Rightarrow Planificador Largo, Mediano, Corto, Alcanzar, como Planificador tiene un control de
 ALGORITMOS q' van al Planificador de corto alcance \Rightarrow No apropiativo (evita la concurrencia de los procesos), no apropiativo,
 MULTIPROGRAMACIÓN (Puede generar conflictos con la sincronización de procesos).

OBJETIVOS DE LA SINCRONIZACIÓN DE PROCESOS \Rightarrow Procesos no interfieren entre sí, controla el acceso a los recursos compartidos
EXCLUSIÓN MUTUA \Rightarrow Controlar el acceso a la región crítica (Porción de código del proceso permitiendo acceso a la zona compartida)

- \hookrightarrow **CONDICIONES** \Rightarrow Si un proceso quiere acceder, puede hacerlo en un tiempo finito y no a la vez.
- \hookrightarrow **PER SOFTWARE** \Rightarrow VARIABLE COMPARTIDA Y SUS PROBLEMAS (RESPONSA ACTIVA) PERO GUBERNABLE, DIFÍCIL DE IMPLEMENTAR

EXCLUSIÓN MUTUA POR HARDWARE \Rightarrow 2 TIPOS \Rightarrow VENTAJAS: DESESTABILIZA INTERFERENCIAS, DESVENTAJAS: PÉRDIDA EFICIENCIA

PROBLEMAS CUANDO HAY VARIOS PROCESADORES, FUNCIÓN SETTER-GETTER DE MEMORIA, ESPERAR INDEFINIDAMENTE, RESPONSA ACTIVA.

2. **SEMAFORO** \Rightarrow Cuentas q' poseen contador, FIFO, función UP DOWN, VENTAJAS: NO RESPONSA ACTIVA, CONFiable, DESVENTAJAS: DIFÍCIL EVITAR
 PROBLEMAS \Rightarrow " " encapsula todos los recursos compartidos, se accede desde métodos públicos del monitor

GARANTIZA q' solo este un proceso a la vez. VENTAJAS: NO RESPONSA ACTIVA, CONFiable, FÁCIL IMPLEMENTAR, PÉRDIDA PROPORCIONAL q' se
 PRODUCE DEADLOCK. DESVENTAJAS: NO ESTÁN HECHOS COMO SEMAFORO.

UNIDAD 2

3. **DEADLOCK** \Rightarrow BLOQUEO MUTUO DE PROCESOS **STARVATION** \Rightarrow PROCESOS q' REQUIEREN UN RECURSO DENTRO DE UN DEADLOCK.

MECANISMOS PARA RESOLVER \Rightarrow ANÁLISIS: NO HAY NADA

PREVENIR: PREVENIR LAS CONDICIONES DEL DEADLOCK \Rightarrow MUTUA EXCLUSIÓN (NO USAR SINCRONIZACIÓN DE PROCESOS) 3
 \hookrightarrow **TOMAR Y ESPERAR** (PREDIR LOS RECURSOS JUNTOS) 3
 \hookrightarrow **ALGORITMOS NO APROPIATIVOS** (LIBERAR RECURSOS) 3
 \hookrightarrow **ESFERA CIRCULAR** (PREDIR LOS RECURSOS EN ORDEN) 3

EVITAR: ASIGNAR RECURSOS A MEDIDA q' LOS PROCESOS LO VAN SOLICITANDO (ALGORITMO BANKER)

\hookrightarrow **VENTAJAS** = MINIMA SE PRODUCE DEADLOCK

\hookrightarrow **DESVENTAJAS** = HAY q' CONOCER DE TODOS LOS PROCESOS q' RECURSOS VAN SOLICITANDO.

DETECTAR Y ELIMINAR: ANALIZA SI SE PRODUCE UNA ESFERA CIRCULAR ENTRE LOS NOMBRES (NO HAY = NO HAY NADA / SI HAY =
 INTENTA ELIMINARLO LIBERANDO RECURSOS TODOS)

\hookrightarrow **VENTAJAS** = MENOS COSTOSO

\hookrightarrow **DESVENTAJAS** = SE PUEDE PRODUCIR DEADLOCK Y CONOCER DE TODOS LOS PROCESOS q' RECURSOS VAN SOLICITANDO

UNIDAD 3

7 CAPAS OSI \Rightarrow 1. FÍSICO (MEDIO FÍSICO AL QUE CADA UNO SE CONECTA LOS EQUIPOS)

2. Datos (Aquí se le da el valor a la dirección unitario físico)

3. **ANÁLISIS** (DIRECCIONA MI RUTA LÓGICA; AGROUPO LA CONVERSIÓN ENTRE EQUILIBRIOS)

4. TRANSPORT (PROTOCOLS q'ALABURAN q'AL MANSUKH SA RABIA / SA RCTOR)

5. Sección (MANTENIR ABIERTA LA COMUNICACIÓN ENTRE LOS DISPOSITIVOS DE RAD; SEPARAR Y CONTROLAR EL ROL DE LOS DIFERENTES EQUIPOS)

6. REPRESENTACION (REPRESENTA DADOS, TIPOS DE ARCHIVOS, ETC)

7. APLICACIÓN (LO MÁS CERCANO AL USUARIO) SE CUIDARÁ DE LA COMPOSICIÓN DE LA COMUNICACIÓN AL USUARIO)

ARQUITECTURAS \Rightarrow

RED: PERMITIR LA COMUNICACIÓN ENTRE EQUIPOS CONECTADOS EN LA MISMA RED

CLIENTA - SANCILDER: PROMIETE CRUZAR LOS DATOS EN UNA BASE DE DATOS Y ELIMINAR REDUNDANCIA E INCONSISTENCIA EN LOS DATOS Y ASEGURANDO QUE

3 CAPAS : PERMITA DISTRIBUIR LA CARGA DIVIDIENDO EL SERVIDOR

↳ ACTIVO/PASIVO: UNO TRABAJA/ EL OTRO SE ACTIVA CUANDO EL OTRO PUEDE FUNCIONAR, PERO NO COSTA

↳ ACTIVO/ACTIVO: APOYOS TRABAJANDO PERO TIENEN PROBLEMAS DE BALANCEO DE CARGA.

- SISTEMAS DISTRIBUIDOS \Rightarrow CONJUNTO DE COMPUTADORAS Q' SE INTEGRAN PARA OFERTAR LA VISIÓN DE UN SISTEMA ÚNICO

↳ **OBJETIVOS:** DISTRIBUIR CARGA DE TRABAJO, RECURSOS, ALTO RENDIMIENTO, RESPONSABILIDAD Y DISPONIBILIDAD

→ CARACTERÍSTICAS: TRANSPARENCIA, CONCURSITA, MODULAR, CONFIGURABLE Y CONSISTENTE

→ Tipos: CLUSTER (NO HAY COORDINADOR), P2P (NO HAY NODO COORDINADOR) / GRID (RES UN MAN)

DISTRIBUIR LA CARGA DE TRABAJO \Rightarrow INTEGRAR / TRANSFERIR UN PROCESO DE UN NODO A OTRO PARA BALANCEAR LA CARGA DE TRABAJO
(COMPARTIR TRABAJO CON VARIOS PC)

↳ POLÍTICA DE INFORMACIÓN (MEDIR CARGA DE TRABAJO Y REDUCIR INFORMACIÓN)

→ " " TRANSFERENCIA (BALANCEO DE CARGA) COMPARTIR CARGA Y OPTIMIZAR ACCESO A RECURSOS

→ " " SELECTION (APPROPRIATE / NO APPROPRIATE)

• " " IMPLANTASI (DESILIAN KESTABIL) DINAMIS & ADAPTIF

UNIDAD 4

REDES VIRTUAIS \Rightarrow PERMITE ORÇENAR RECURSOS DE UM SISTEMA DISTRIBUÍDO

PROMITE SABER q' PUEDES SER FICADO ANTES Y DESPUES (2 TIPOS =)

File) de import: usa CONTAINER, para no WORKBOOK q' tal processo se FIZEREM OUTROS DESPES

REGLO VECTORIAL: LA CURVA q' PROGRESO SE REALIZA ANTES O DESPUES.

SIN BENEFICIAÇÃO DE PROCESSOS VIA SEMEFAPO ⇒ NO, NO VEM MANEIRA COMBARTIDA, TODA DE FILMA, YASTAS GLOBOS.

RECLUSÃO NAUAFIDIANTE NA SORSA ⇒ SE BASE NA IL DO DO PRIMITIVO (SOLAR + MANSO); FUNÇÃO NAUAFIDIANTE NA SORSA

SINCRONIZACIÓN DE MÁQUINAS CUANDO HAYE MÚLTIPLES NODOS Y PROCESOS \Rightarrow CONTROL TRABAJO: NODO MONITOR (CONTROLA QUE LOS NODOS SE EJECUTEN EN ORDEN)

DESCRIPCIÓN DE CONTEXTO: NO HAY NINGÚN CONTROL, NO SE PUEDE TENER PARA ACCEDER AL RECURSO

11 Sin 11 : 6 11 11 11 / ni mo tefu; se acaade fu oridin de selctud.

UNIDAD 5/6

CACHE DE DATOS \Rightarrow SE ALMACENA TEMPORALMENTE EN MEMORIA

CONSISTENCIA DE LA CACHE \Rightarrow ES CONSISTENTE CUANDO LOS DATOS CACHADOS SON IGUALES A LOS REALES

MEMORIA DISTRIBUIDA \Rightarrow PERMITE GENERAR ^{SECCIONES} DE MEMORIA COMPARTIDA ENTRE NODOS DE UN SISTEMA DISTRIBUIDO

\rightarrow CENTRALIZADA: LOS NODOS HACEN LAS ACTUALIZACIONES EN UN SOLO NODO

\rightarrow DESCENTRALIZADA CON REPLICACIÓN: UN NODO COPIA LA INFO DE OTRO NODO JARDIN, COPIA A SU UBICACIÓN ORIGINAL

\rightarrow " CON MISIÓN: TOMA LA INFO DE UN NODO Y LOS CAMBIOS REALIZADOS SE GUARDAN EN EL MISMO NODO.

SISTEMA DE ARCHIVO DISTRIBUIDO \Rightarrow PERMITE GENERAR UN SIST DE ARCHIVO GLOBAL COMPARTIDO ENTRE NODOS DE UN SIST DISTRIBUIDO

REPLICACIÓN DE RECURSOS, ALGORITMO Y TIPOS \Rightarrow MANTENER COPIAS DE LOS RECURSOS Y MANTENER LAS COPIAS ACTUALIZADAS.

ALGORITMO DE COPIAS: GARANTIZA Q' TODAS LAS COPIAS ESTÉN ACTUALIZADAS.

\rightarrow ACTUALIZA TODO O NADA, PRIMERA COPIA, BASTINGO Y QUORUM (SE RESUELTA OPERACIÓN EN CASO DE LOS NODOS, SI HAY CAMBIO SE PROPAGAN LOS CAMBIOS)

UNIDAD 7

ASEGURAR LA CONFIABILIDAD Y FIABILIDAD \Rightarrow TRANSACCIONES DISTRIBUIDAS:

CONJUNTO DE OPERACIONES Q' GARANTIZAN AL ACOP

A - ATOMICIDAD (SE REALIZAN TODAS LAS OPERACIONES O NINGUNA)

C - CONSISTENCIA (SE ENCUENTRA EN UN ESTADO ESTABLE)

A - AISLACIÓN (LOS EFECTOS DENTRO DE LA OPERACIÓN NO SON VISIBILIZABLES)

P - PERMANENCIA (QUEDAN PERMANENTES LAS OPERACIONES COMPLETADAS)

TIPOS: TRANSACCIÓN 1 FASE \Rightarrow UN NODO LLAMA A OTRO NODO PARA HACER X TAREA (3 FUNCIONES: COMMIT, ABORTAR, ROLLBACK)

" 2 FASE \Rightarrow UN NODO (COORDINADOR) LLAMA A OTROS NODOS " " " EN PARALELO " " : PUNTO COMMIT, ROLLBACK, OBTENER RESPUESTA Y HACER COMMIT

MANEJO DE FALLOS \Rightarrow FALLO: CUANDO UN NODO NO PUEDE CUMPLIR SU FUNCIÓN (FALLO DEL SISTEMA/COMUNICACIÓN)

RECUPERACIÓN DE FALLOS:

- RECUPERACIÓN POR ABORTAR: ACTIVAR OPERACIONES INTERRUMPIDAS AL FALLO

" " ROLLBACK: REALIZA EL ROLLBACK PARA VOLVER A UN ESTADO NORMAL

- TOLERANCIA A FALLOS:

- FALLOS GARANTIZADO DE FALLO: HACER UN TRY CATCH, REALIZA X OPERACIONES PARA SEGUIR FUNCIONANDO

- COMPARTIMENTACIÓN DEL PROCESO: SE IMPIDE Q' OPERACIONES TIENAN Q' HACER CUANDO SE PRODUCE UN FALLO.

- RESILIENCIA " " : EL PROCESO YA SABE COMO ACTUAR ANTES UN FALLO.

SEGURIDAD EN SISTEMAS DISTRIBUIDOS \Rightarrow TIPOS:

- INTERFERENCIA: CORTAR EL MEDIO DE COMUNICACIÓN ENTRE NODOS

- INTERCEPCIÓN: ACCEDER AL MENSAJE Q' SE MANDA

- MODIFICAR/CREAR: MODIFICAR O CREAR MENSAJES ENTRE NODOS

