



FORMATO 4

SÍLABO-MODELO

CURSO: CC4P1 PROGRAMACIÓN CONCURRENTES Y DISTRIBUIDA

I. INFORMACIÓN GENERAL

CODIGO	: CC4P1 PROGRAMACIÓN CONCURRENTES Y DISTRIBUIDA
CICLO	: 6
CREDITOS	: 04 (CUATRO)
HORAS POR SEMANA	: 06 (TEORÍA: 02, LABORATORIO:04)
PRERREQUISITOS	: CC642 - PROGRAMACIÓN PARALELA
CONDICION	: OBLIGATORIO
ÁREA ACADÉMICA	: Ciencias de la Computación
PROFESOR	: Yuri Nuñez Medrano
E-MAIL	: ynunezm@uni.edu.pe

II. SUMILLA DEL CURSO

El Curso está directamente ligada con la Programación de Sistemas Distribuidos y su desempeño en los Sistemas Operativos y por ello La Programación Concurrente está ligada a los conceptos de multiproceso y multiprocesador, por lo que se estudiará las técnicas básicas de gestión de la concurrencia, como por ejemplo los mecanismos de sincronización y comunicación entre procesos. Referente a la Programación Distribuida, está relacionada con el desarrollo de sistemas en los cuales la red es una infraestructura crítica. Por tanto, el alumno aprenderá los fundamentos básicos de programación de cliente/servidor bajo los protocolos UDP/TCP, hasta escalar al nivel más elevado automatizando las operaciones del sistema. En este contexto, se trabajarán los paradigmas RPC/RMI y la tecnología de componentes, una panorámica del tipo de aplicaciones y Sistemas Distribuidos Modernos (Clustering, Grid, Cloud, P2P).

III. COMPETENCIAS DE LA ASIGNATURA

Al finalizar la asignatura, el estudiante:

- Programar en entornos de red con arquitectura cliente/servidor.
- Aplicar las diferentes técnicas de comunicación entre procesos y grupos de procesos distribuidos.
- Programar aplicaciones para entornos distribuidos.
- Resolver problemas complejos y de respuesta en tiempo real utilizando conceptos y herramientas de programación concurrente y tiempo real, planificando sus tareas, así como un uso eficiente de la memoria.



FORMATO 4

IV. UNIDADES DE APRENDIZAJE

1. **INTRODUCCIÓN A LOS SISTEMAS DISTRIBUIDOS: / Semana 1**
I Conceptos fundamentales de Sistemas Concurrentes y Sistemas Distribuidos.
2. **INTRODUCCION A CONCURRENCIA Y SUS TIPOS / Semana 2**
I Conceptos Generales, II: Procesos e Hilos, III: Modelos, IV tiempo real.
3. **COMUNICACIÓN POR MEMORIA COMPARTIDA: / Semana 3**
I. Semáforos, II. Regiones Críticas Condicionales, III. Monitores. IV Sincronización.
4. **COMUNICACIÓN CON SOCKETS / Semana 4**
I: Cliente y Servidor, II: Diferentes tipos de comunicaciones.
5. **PASO DE MENSAJES / Semana 5**
I. Protocolo de Comunicación
6. **EL MODELO CLIENTE/SERVIDOR / Semana 6**
I: Arquitectura, II: Clientes y Servidores, III: Plataforma, IV: Modelos 2-Tier 3-Tier, Multi Tier
7. **(PARCIAL) / Semana 8**
8. **MIDDLEWARE / Semana 9**
I: Llamada a un Procedimiento remoto (RPC), II: Middleware Orientado a Mensajes (MOM), III: Peer-to-peer, IV: Servicio de directorio, V: Seguridad
9. **SISTEMAS DISTRIBUTIVOS DE ARCHIVO / Semana 10**
I: Diseño, II: Implementación, III: Tendencia
10. **OBJETOS DISTRIBUIDOS MODELOS DE COMPONENTES / Semana 11**
I: Objetos y Componentes, II: Beneficios, III: Modelos de Componentes
11. **CONCURRENCIA EN REDES / Semana 12**
I: Balance de Carga, II: Alta Disponibilidad
12. **TECNOLOGÍAS DISTRIBUIDAS MODERNAS / Semana 13**
I. Clustering, II. Tecnologías Grid y Cloud (virtualización), III. Servicios Web, IV. Computación móvil y ubicua, 5. Redes P2P.
13. **SINCRONIZACIÓN EN SISTEMAS DISTRIBUIDOS / Semana 14**
I. Relojes físicos y lógicos, II. Estados globales, III. Exclusión mutua distribuida, IV. Elecciones.
14. **TRANSACCIONES DISTRIBUIDAS / Semana 15**
I. Transacciones, II. Control de concurrencia en transacciones, III. Control optimista de concurrencia, IV. Ordenación por marcas de tiempo, V. Recuperación de transacciones.
15. **(FINAL). / Semana 16**

V. LABORATORIOS Y EXPERIENCIAS PRÁCTICAS

Laboratorio 1: Desarrollo de Comunicación
Laboratorio 2: Desarrollo de Sincronización
Laboratorio 3: Desarrollo Memoria Compartida
Laboratorio 4: Comunicación y Paso de Mensajes



FORMATO 4

Laboratorio 5: Comunicación del Sistema Distribuido

Laboratorio 6: Comunicación y Concurrencia

VI. METODOLOGÍA

El curso se desarrolla en sesiones de teoría, práctica y laboratorio de cómputo. En las sesiones de teoría, el docente presenta los conceptos, teoremas y aplicaciones. En las sesiones prácticas, se resuelven diversos problemas y se analiza su solución. En las sesiones de laboratorio se usa el lenguaje de programación Java para resolver problemas y analizar su solución. Al final del curso el alumno debe desarrollar el examen final. En todas las sesiones se promueve la participación activa del alumno.

VII. FÓRMULA DE EVALUACIÓN

Cálculo del Promedio Final: $PF = (EP + EF + PL1) / 3$

EP: Examen Parcial EF: Examen Final PC1: Promedio de Laboratorios menos la menor Nota

VIII. BIBLIOGRAFÍA *

1. Andrew S. Tanenbaum. Sistemas Operativos Distribuidos. Publicado por Prentice-Hall, 1996.
2. Andrew S. Tanenbaum. Redes de Ordenadores. Publicado por Prentice-Hall, 1991.
3. Robert Orfali, Dan Harkey, Jeri Edwards. Client/Server Survival Guide. 3er ed. Publicado por John Wiley, 1999.
4. George Coulouris, Jean Dollimore and Tim Kindberg. Distributed Systems: Concepts and Design. 4ta ed. Publicado por Addison-Wesley. 2005.
5. Tom Sheldon . Lan Times - Guía de Interoperabilidad. Publicado por Osborne/McGraw-Hill, 1995.
6. Tom Sheldon. Lan Times - Enciclopedia de Redes – Networking. Publicado por McGraw-Hill 1994.
7. Reaz Hoque . Corba 3 developers guide - IDG Books, 1998.

IMPORTANTE Enviar el formato al email: acreditación.....@uni.edu.pe