



Casa abierta al tiempo

UNIVERSIDAD AUTONOMA METROPOLITANA

PROGRAMA DE ESTUDIOS

UNIDAD	IZTAPALAPA	DIVISION	CIENCIAS BASICAS E INGENIERIA	1 / 3
NOMBRE DEL PLAN LICENCIATURA EN COMPUTACION				
CLAVE	UNIDAD DE ENSEÑANZA-APRENDIZAJE		CRED.	8
2151111	PROGRAMACION CONCURRENTE		TIPO	OBL.
H.TEOR. 3.0	SERIACION		TRIM.	
H.PRAC. 2.0			VII	
	2151115			

OBJETIVO(S) :

Objetivo General:

Al final de la UEA el alumno será capaz de resolver problemas utilizando conceptos y técnicas de programación concurrente.

Objetivos Específicos:

Al final de la UEA el alumno será capaz de:

- Identificar la necesidad de uso de un modelo de programación concurrente.
- Desarrollar habilidades para la construcción de programas concurrentes, desde la concepción, el diseño y la implementación.
- Comprender los modelos de programación de memoria compartida y paso de mensajes, así como sus principales herramientas de programación.

CONTENIDO SINTETICO:

1. Introducción (1 semana).
 - 1.1. Conceptos básicos.
 - 1.2. Programación secuencial vs. programación concurrente.
 - 1.2.1. Ventajas.
 - 1.2.2. Desventajas.
 - 1.2.3. Infraestructura.
2. Procesos e hilos (2 semanas).
 - 2.1. Creación y comunicación de procesos.
 - 2.2. Creación y comunicación de hilos.
3. Modelo de programación de memoria compartida (4 semanas).



Casa abierta al tiempo

UNIVERSIDAD AUTONOMA METROPOLITANA

ADECUACION
PRESENTADA AL COLEGIO ACADEMICO
EN SU SESION NUM. 383

EL SECRETARIO DEL COLEGIO

CLAVE 2151111

PROGRAMACION CONCURRENTE

- 3.1. Concepción del modelo.
 - 3.2. Mecanismos de sincronización.
 - 3.2.1. Candados.
 - 3.2.2. Semáforos.
 - 3.2.3. Barreras.
 - 3.3. Herramientas de programación.
 - 3.3.1. Bibliotecas de manejo explícito de hilos.
 - 3.3.2. Bibliotecas de manejo implícito de hilos.
 - 3.4. Construcción de aplicaciones.
4. Modelo de programación de paso de mensajes (4 semanas).
- 4.1. Concepción del modelo.
 - 4.2. Comunicación entre procesos.
 - 4.2.1. Síncrona.
 - 4.2.2. Asíncrona.
 - 4.2.3. Punto a punto.
 - 4.2.4. Multipunto.
 - 4.3. Bibliotecas para envío y recepción de mensajes.
 - 4.4. Construcción de aplicaciones.

MODALIDADES DE CONDUCCION DEL PROCESO DE ENSEÑANZA-APRENDIZAJE:

El profesor utilizará la clase magistral para exponer los temas del curso propiciando la participación activa y corresponsable en el proceso de enseñanza-aprendizaje, el pensamiento crítico, la disciplina y el rigor en el trabajo académico, así como la capacidad para aprender por sí mismo.

Para lograr lo anterior se podrán desarrollar actividades tales como tareas, trabajos de investigación y exposición de temas.

Se deberán desarrollar exhaustivamente ejemplos y ejercicios sobre los temas abordados.

En las horas de práctica, el profesor utilizará la modalidad de taller para que los alumnos resuelvan problemas de manera individual o grupal.

MODALIDADES DE EVALUACION:

Evaluación Global:

La evaluación global de esta UEA incluirá evaluaciones periódicas y, a juicio del profesor, una evaluación terminal.

Se sugiere que las evaluaciones periódicas sean un mínimo de dos escritas y una oral.

**UNIVERSIDAD AUTONOMA METROPOLITANA**

ADECUACION
PRESENTADA AL COLEGIO ACADEMICO
EN SU SESION NÚM. 383

V. Wang
EL SECRETARIO DEL COLEGIO

CLAVE 2151111

PROGRAMACION CONCURRENTE

El profesor seleccionará los elementos de evaluación periódica de entre los siguientes: evaluaciones, participación en clase, tareas, programas, trabajos de investigación y presentaciones de temas.

Los factores de ponderación quedarán a juicio del profesor y se darán a conocer al inicio del curso.

Evaluación de Recuperación:

La evaluación de recuperación de esta UEA podrá ser global o complementaria, a juicio del profesor.

BIBLIOGRAFIA NECESARIA O RECOMENDABLE:

1. Ben-Ari, M., (2006), Principles of Concurrent and Distributed Programming, 2a ed., Addison-Wesley, EUA.
2. Doug, L., (1999), Concurrent Programming in Java: Design Principles and Pattern, 2a ed.; Prentice Hall, EUA.
3. Goetz, B., Peierls, T., Bloch, J., Bowbeer, J., Holmes, Lea, D., (2006), Java Concurrency in Practice; Addison-Wesley, EUA.
4. Herlihy, M., Shavit, N., (2008), The Art of Multiprocessor Programming, Morgan Kaufmann, EUA.
5. Hughes, C., Hughes, T., (2008), Professional Multicore Programming: Design and Implementation for C++ Developers, Wrox, EUA.
6. Wellings, A., (2004), Concurrent and Real-Time Programming in Java, Wiley, EUA.



Casa abierta al tiempo

UNIVERSIDAD AUTONOMA METROPOLITANA

ADECUACION
PRESENTADA AL COLEGIO ACADEMICO
EN SU SESION NUM. 383

[Handwritten Signature]
EL SECRETARIO DEL COLEGIO