

Planeación del curso: Programación Concurrente

Trim 2014-O

I.- Información general

Nombre de la UEA: PROGRAMACION CONCURRENTE

Clave de la UEA: 2151111

Grupo: CG01

Horario de clases teóricas y prácticas:

Teoría: Martes de 10:00 – 11:00, Salón: B-103

Jueves de 10:00 – 12:00, Salón: B-103

Laboratorio: Viernes de 11:00 a 13:00, Salón: AT-220

Nombre del profesor: **Graciela Román Alonso** (Cubículo T-215)

email: grac@xanum.uam.mx

Horario de asesorías: Martes, Miércoles y Jueves de 11:00 a 14:00

II.- Contenido del curso:

a) Objetivos

Generales

Que al final del curso el alumno sea capaz de resolver problemas utilizando conceptos y técnicas de programación concurrente.

Específicos

Que al final del curso el alumno sea 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.

b) Contenido

● Introducción (1 semana).

- Conceptos básicos.
- Programación secuencial vs. programación concurrente.
 - Ventajas.
 - Desventajas.
 - Infraestructura.

● Procesos e hilos (2 semanas).

- Creación y comunicación de procesos.
- Creación y comunicación de hilos.
- **Modelo de programación de memoria compartida (4 semanas).**
 - Concepción del modelo.
 - Mecanismos de sincronización.
 - Candados.
 - Semáforos.
 - Barreras.
 - Herramientas de programación.
 - Bibliotecas de manejo explícito de hilos.
 - Bibliotecas de manejo implícito de hilos.
 - Construcción de aplicaciones.
- **Modelo de programación de paso de mensajes (4 semanas).**
 - Concepción del modelo.
 - Comunicación entre procesos.
 - Síncrona.
 - Asíncrona.
 - Punto a punto.
 - Multipunto.
 - Bibliotecas para envío y recepción de mensajes.
 - Construcción de aplicaciones.

c) Bibliografía:

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

III.- Evaluación del curso:

Fechas de exámenes parciales:

- Primer examen parcial: Jueves 2 de Octubre, semana 4

- Segundo examen parcial: Jueves 30 de Octubre, semana 8
- Tercer examen parcial: Jueves 27 de Noviembre, semana 11 BIS

Calificación Final:

⤴ Exámenes:	50%
⤴ Prácticas:	20%
⤴ Tareas y Programas:	15%
⤴ Proyecto Final:	15%

Se aplicarán 3 exámenes parciales que deberán ser presentados de manera obligatoria.
 Requisito para aprobar el curso: Tener un **promedio aprobatorio** de los 3 exámenes y haber entregado el proyecto final funcionando.

No se aplicará examen global.

Uso del lenguaje C + OpenMP + MPI

Escala:

6	a	7.4	S
7.5	a	8.4	B
8.5	a	10	MB