



PROGRAMACIÓN DEL SÍLABO DE APRENDIZAJE

I. DATOS GENERALES

- Nombre de la asignatura : Algoritmos y Programación Paralela
- Código del curso : IIS 0343
- Cantidad de horas semanal : 06 (Teoría: 02 | Práctica: 02 | Laboratorio: 02)
- Créditos : 04
- Escuela Profesional : Ingeniería en Informática y Sistemas
- Ciclo académico : IV ciclo
- Docente responsable : Msc. Ana Silvia Cori Morón
- Duración : 17 semanas
- Fecha de inicio : 20 de agosto del 2018
- Régimen : Flexible

II. APORTE DE LA ASIGNATURA AL PERFIL PROFESIONAL

Tiene como propósito brindar al futuro profesional de Ingeniería en Informática y Sistemas conocimientos sobre la posibilidad de realizar múltiples tareas simultáneamente mediante hardware, que no es inmediatamente traducida a software, pues las aplicaciones deben ser diseñadas para aprovechar estas nuevas capacidades, mediante el uso de hebras y/o procesos, porque las arquitecturas de computadores están teniendo a incluir cada vez más núcleos y/o procesadores por máquina como método de incrementar la capacidad computacional de cada unidad.

III. SUMILLA

La asignatura de Algoritmos y Programación Paralela es de carácter teórico - práctico. En la asignatura se desarrollarán conceptos básicos acerca de concurrencia y paralelismo, Programas concurrentes, Propiedades de la programación concurrente, Exclusión mutua. Mecanismos de Comunicación y sincronización en memoria compartida. Introducción a la Programación Paralela. Clasificación Lógica del Paralelismo. Clasificación Física de Computadores Paralelos. Paradigmas de Programación Paralela. Diseño de Programas Paralelos.

IV. LOGRO DE LA ASIGNATURA

Al finalizar la asignatura, el alumno desarrolla aplicaciones computacionales utilizando mecanismos de sincronización que permitan gestionar correctamente el uso de memoria compartida aplicando las métricas de complejidad de los algoritmos concurrentes y paralelos.

V. PROGRAMACIÓN DE CONTENIDOS

UNIDADE DE APRENDIZAJE I: PRIMERAS APROXIMACIONES AL PROCESAMIENTO PARALELO

SEMANA	CONTENIDOS CONCEPTUAL	PROCEDIMENTAL	ACTITUDINAL
01	Conceptos básicos de algoritmos. Concurrencia vs Paralelismo. Programación concurrente. Programación distribuida y paralela. Arquitecturas hardware. Metodología y notación de la programación concurrente y paralela. Condiciones de Bernstein e indeterminismo. Grafos de precedencia.	• Distingue la diferencia entre concurrencia y paralelismo. • Realiza el pseudocódigo de los grafos de precedencia. • Utiliza las condiciones de Bernstein para determinar si una serie de procesos se pueden ejecutar de forma concurrente.	• Asiste puntualmente a las clases teóricas y prácticas. • Participa de forma espontánea en la solución de ejercicios.
02			
03	Programa y proceso. Procesos e hilos, representación y ciclo de vida de los procesos, concepto y estructura de un proceso e hilo. Interacción entre procesos.	• Conocimiento de procesos e hilos. • Algoritmos de sincronización y exclusión mutua.	

SEMANA	CONTENIDOS CONCEPTUAL	PROCEDIMENTAL	ACTITUDINAL
	Planificación entre procesos.		
04	Sincronización y exclusión mutua. Primitivas de sincronización para variables compartidas. Modelos de problemas clásicos.		
05	Algoritmos de exclusión mutua con memoria compartida. Espera ocupada. Algoritmos no eficientes. Primer intento, segundo intento, tercer intento, cuarto intento.	• Conocer los algoritmos de sincronización y exclusión mutua con variables compartidas.	
06	Analogía de los esquimales. Quinto intento. Algoritmo de Dekker. Algoritmo de Peterson.		
07	Algoritmo incorrecto de Hyman. Algoritmos de exclusión mutua para n procesos.	• Implementación de los algoritmos de exclusión mutua para n procesos.	
08	Algoritmo de Knuth y Kesell. Algoritmo de Eisenberg-McGuire. Algoritmo de Lamport. Soluciones hardware.		
09			

EXAMEN PARCIAL I

ESTRATEGIAS DIDACTICAS

- Conferencia para la teoría.
- Participación activa en clase en la solución de ejercicios.
- Talleres grupales.
- Prácticas.

TIEMPO: 08 SEMANAS

UNIDAD DE APRENDIZAJE II: PRIMITIVAS DE SINCRONIZACIÓN BASADAS EN MEMORIA COMPARTIDA

SEMANA	CONTENIDOS CONCEPTUAL	PROCEDIMENTAL	ACTITUDINAL
10	Semáforos, conceptos, clasificación y resolución de problemas usando semáforos. Las operaciones wait y signal. Exclusión mutua y condición de sincronización con semáforos.	• Comprende y resuelve problemas con semáforos.	• Asiste puntualmente a las clases teóricas y prácticas.
11	El problema y algoritmo de los filósofos comensales. El problema y algoritmo del productor consumidor.	• Comprende los problemas tipos.	
12	El problema y algoritmo del barbero durmiente. El algoritmo de los lectores y escritores. Inconvenientes con el mecanismo de semáforos.		

SEMANA	CONTENIDOS CONCEPTUAL	PROCEDIMENTAL	ACTITUDINAL
13	Monitores, introducción, concepto, sincronización de monitores. Las operaciones delay, resume y empty. Algoritmos con monitores. Implementación de monitores con semáforos.	• Comprende y resuelve problemas con monitores	
14	Paso de mensajes. Canales de comunicación. Espera selectiva. Paso de mensajes asíncrono. Resolución de problemas utilizando paso de mensajes asíncrono.	• Usa correctamente el mecanismo de paso de mensajes en Pascal-FC y en Java.	• Participa de forma esporádica en la solución de ejercicios.
15	Paso de mensajes sincrónico. Comunicación mediante canales en Pascal-FC. Espera selectiva. Resolución de problemas usando canales. Paso de mensajes sincrónico con Java.		
16	EXPOSICIONES DE PROYECTOS DE INVESTIGACIÓN		
17	EXAMEN PARCIAL II		

ESTRATEGIAS DIDACTICAS

- Conferencia para la teoría.
- Participación activa en clase en la solución de ejercicios.
- Talleres grupales.
- Prácticas.

TIEMPO: 09 SEMANAS

VI. SISTEMA DE EVALUACIÓN

PROCEDIMIENTOS		PORCENTAJE	PONDERACIÓN
TEORÍA	Exámenes parciales	50%	10
60%	Asistencia, evaluación continua, esfuerzo personal	10%	2
PRÁCTICA	Trabajo de investigación	20%	4
40%	Implementación de algoritmos en laboratorio	20%	4

OBSERVACIÓN:

- La calificación será vigesimal (de 00 a 20) siendo la nota mínima de aprobación 11 (once).
- Solo para el promedio final se considerará 0.5 a favor del estudiante.
- Los alumnos que no se presenten a rendir sus evaluaciones en las fechas indicadas y definidas en clase tendrán la calificación 00. En caso la inasistencia sea debidamente justificada mediante los canales respectivos, según normas vigentes de la universidad, se reprogramará dicha evaluación.
- La evaluación de los estudiantes es permanente y el contenido de los exámenes es de todo el desarrollo del curso a la fecha de aplicación de la evaluación. La evaluación del estudiante es continua y no necesariamente requiere de aviso previo (excepto los exámenes parciales), siempre que las evaluaciones se realicen en horarios establecidos para el curso.
- El porcentaje de inasistencia a clases y/o laboratorio de más del 30% será causal de desaprobación del curso.

El promedio final del curso se obtiene de acuerdo a los criterios de evaluación que se señalaron anteriormente.

VII. ESTRATEGIAS METODOLÓGICAS

CLASES TEÓRICAS

- Método : Analítico y deductivo.
- Técnica : Exposición teórica en aula propiciando el análisis y participación activa del alumno.
- Procedimiento : Planteamiento de problemas y la búsqueda de soluciones mediante el planteamiento de algoritmos computacionales. La complejidad de los problemas será progresiva, de acuerdo al avance del curso.

PRÁCTICAS DE LABORATORIO Y/O EN AULA

- Método : Analítico y deductivo.
- Técnica : Implementación de algoritmos mediante el uso de lenguajes de programación de alto nivel.
- Procedimiento : Planteamiento de problemas y la búsqueda de soluciones mediante el planteamiento de algoritmos computacionales. La complejidad de los problemas será progresiva, de acuerdo al avance del curso.

VIII. MATERIALES EDUCATIVOS Y OTROS RECURSOS DIDÁCTICOS

- Se utilizará material impreso y en formato digital.
- Pizarra acrílica, plumones, proyector multimedia y el computador.
- Prácticas y guías de laboratorio dirigidas.

IX. BIBLIOGRAFÍA BÁSICA

- Cornei, E. (2010) Lenguaje de programación Java. Lima: Macro.
- Drozdek, A. (2007) Estructuras de datos y algoritmos con Java. México: Thomson

X. BIBLIOGRAFÍA COMPLEMENTARIA

- Pérez J. (2006). *Programación concurrente*. Segunda edición. España: Editorial Rueda.
- Jorba J. & Suppi R. Programación concurrente.
- Vallejo D., Gonzales C. & Albusac J. (2014). *Programación Concurrente y Tiempo Real*. Segunda edición. España: Editrix Editorial
- Aguller J. & Leiss E. (2004). *Introducción a la Computación Paralela*. Primera edición. Venezuela: Gráficas Quinto.
- Art B. (1982). *Principles of Concurrent Programming*. First edition. USA: Prentice-Hall International
- Quinn M. (2003). *Parallel programming in C with MPI and OpenMP*. Singapore: McGraw Hill Education
- Palma J., Garrido M., Sánchez F. & Quesada A. (2006). *Programación Concurrente*. Primera edición. España: Paraninfo
- Almeida F., Gimenez D., Mantas J. & Vidal A. (2008). *Introducción a la programación paralela*. Primera Edición. España: Editorial Paraninfo.
- Lin C. & Snyder L. (2008). *Principles of Parallel Programming*. First edition. Addison Wesley.
- Gallí R. (2015). Principios y algoritmos de concurrencia.