



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
PROGRAMA DE POSGRADO EN CIENCIA E INGENIERÍA DE
LA COMPUTACIÓN



Tomo II

(Especialización en Cómputo de Alto Rendimiento)

Planes de Estudio

-) Maestría en Ciencia e Ingeniería de la Computación
-) Doctorado en Ciencia e Ingeniería de la Computación
-) Especialización en Cómputo de Alto Rendimiento

Grados que se otorgan

-) Maestro(a) en Ciencia e Ingeniería de la Computación
-) Doctor(a) en Ciencia e Ingeniería de la Computación
-) Especialista en Cómputo de Alto Rendimiento

Campos de conocimiento que comprende para maestría y doctorado

-) Teoría de la Computación
-) Inteligencia Artificial
-) Computación Científica
-) Señales, Imágenes y Ambientes Virtuales
-) Ingeniería de Software y Bases de Datos
-) Redes y Seguridad en Cómputo

Campos de conocimiento en los que se articula la especialización

-) Computación Científica
-) Ingeniería de Software y Bases de Datos
-) Redes y Seguridad en Cómputo

Entidades académicas participantes

- Facultad de Ciencias
- Facultad de Ingeniería
- Facultad de Estudios Superiores Cuautitlán
- Instituto de Ingeniería
- Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas
- Instituto de Matemáticas
- Centro de Ciencias Aplicadas y Desarrollo Tecnológico

Entidades académicas que se incorporan de manera exclusiva a la especialización

- Instituto de Geofísica (IG)
- Instituto de Astronomía (IA)
- Instituto de Física (IF)
- Dirección General de Cómputo y de Tecnologías de Información y Comunicación (DGTIC)

Fechas de aprobación u opiniones

Creación del plan de estudios de la Especialización en Cómputo de Alto Rendimiento, para su adición al Programa de Posgrado en Ciencia e Ingeniería de la Computación.

Lo anterior implicó también:

- a) Modificación de las Normas Operativas del Programa.
 - b) Incorporación de cuatro entidades académicas participantes en el Programa, exclusivamente para colaborar en la Especialización en Cómputo de Alto Rendimiento: Instituto de Geofísica, Instituto de Astronomía, Instituto de Física y Dirección General de Tecnologías de Información y Comunicación."
- Fecha de aprobación del Consejo Universitario: 1 de julio de 2015.

ÍNDICE

Tomo II

Actividades Académicas Obligatorias

Primer semestre

Fundamentos del Cómputo de Alto Rendimiento	4
Programación Avanzada.....	7

Segundo semestre



Seminario de Cómputo de Alto Rendimiento	10
--	----

Actividades Académicas Optativas

Programación Concurrente y Distribuida	13
Sistemas Paralelos	16
Sistemas en Tiempo Real	18
Diseño de <i>Software</i> Paralelo.....	20
Métricas de Alto Desempeño	23
Laboratorio de <i>Clusters</i> y <i>GRIDS</i>	25
Supercómputo	27
Sistemas Distribuidos	30
Temas Selectos de Cómputo de Alto Rendimiento	32
Modelación Matemática y Computacional de Sistemas Continuos	34
Modelación Matemática y Computacional de Sistemas Terrestres	37
Computación Científica en Paralelo	40

ACTIVIDADES ACADÉMICAS OBLIGATORIAS
PRIMER SEMESTRE

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: FUNDAMENTOS DEL CÓMPUTO DE ALTO RENDIMIENTO			
Clave:	Semestre(s): 1	Campo de Conocimiento: Cómputo de Alto Rendimiento	
Carácter: Obligatorio		Horas	Horas por semana
Tipo: Teórico-Práctica	Teoría: 2	Práctica: 1	Horas al Semestre
Modalidad: Curso		Duración del programa: Semestral	

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente:
Actividad académica subsecuente:
Objetivo general:
El alumno dominará los conceptos fundamentales del cómputo de alto rendimiento, arquitecturas de alto rendimiento, así como las estructuras y algoritmos típicos de programación, para conocer los beneficios e impacto de esta tecnología, en el desarrollo de aplicaciones. Arquitecturas MIMD programadas con lenguajes paralelos serán utilizadas como vehículo para el desarrollo de diversos casos de estudio, así como el uso de plataformas tipo <i>clusters</i> usando PVM Y MPI.
Objetivos específicos:
Al término del curso el alumno será capaz de:
<ol style="list-style-type: none"> 1. Presentar los antecedentes del cómputo en paralelo, de arquitectura y algoritmos. 2. Analizar la clasificación de arquitecturas paralelas. 3. Analizar MIMD y otras arquitecturas. 4. Realizar la programación de procesos paralelos en MIMD. 5. Realizar la programación de procesos paralelos en <i>clusters</i>. 6. Realizar la construcción de algoritmos paralelos. 7. Realizar proyectos de aplicación.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Introducción	6	0
2	Clasificación de Arquitecturas Paralelas	7	0
3	MIMD y otras Arquitecturas	7	0
4	Programación de Procesos Paralelos en MIMD	3	4
5	Programación de Procesos Paralelos en <i>Clusters</i>	3	4
6	Construcción de Algoritmos Paralelos	3	4
7	Proyectos de aplicación	3	4
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Introducción 1.1. Introducción. 1.2. Necesidad del cómputo de alto desempeño. 1.3. Complejidad computacional de un algoritmo. 1.4. Beneficios y problemática. 1.5. Retos. 1.6. Sistemas en tiempo real. 1.7. Métricas de desempeño.
2	Clasificación de Arquitecturas Paralelas 2.1. Introducción.

Especialización en Cómputo de Alto Rendimiento

	2.2. Modelos computacionales. 2.3. Arquitecturas de <i>hardware</i> . 2.4. Arquitecturas de <i>software</i> (SISD, MISD, SIMD, MIMD). 2.5. Topologías de interconexión. 2.6. Comunicación entre procesadores.
3	MIMD y otras Arquitecturas 3.1. Arquitecturas. 3.2. Lenguajes de programación. 3.3. Ambientes e interfaces. 3.4. Sistemas de desarrollo. 3.5. Paralelismo de aplicaciones y <i>threads</i> . 3.6. <i>Clusters</i> . 3.7. PVM y MPI.
4	Programación de Procesos Paralelos en MIMD 4.1. Metodología. 4.2. Programación en un procesador. 4.3. Programación en múltiples procesadores. 4.4. Configuración de programas. 4.5. Mapeo de procesos en procesadores. 4.6. Casos de estudio.
5	Programación de Procesos Paralelos en <i>Clusters</i> 5.1. Metodología. 5.2. Programación en múltiples procesadores. 5.3. Configuración de programas. 5.4. Mapeo de procesos en procesadores. 5.5. Casos de estudio.
6	Construcción de Algoritmos Paralelos 6.1. Algoritmos <i>pipelined</i> /paralelismo algorítmico. 6.2. Algoritmos particionados/paralelismo geométrico. 6.3. Algoritmos asíncronos/paralelismo relajado. 6.4. Algoritmos <i>farm</i> de procesos. 6.5. Balance de carga estático y dinámico.
7	Proyectos de aplicación.

Bibliografía básica:

- Lewis, T.G. (1992). *Introduction to Parallel Computing*. Prentice Hall International Editions.
- Krishnamurthy, E.V. (1989). *Parallel Processing Principles and Practice*. Addison Wesley.
- Thoeni, U.A. (1994). *Programming Real-Time Multi-computers for Signal Processing*. Prentice Hall.
- Webber, H.C. (1992). *Image Processing and Transputers*. IOS Press.

Bibliografía complementaria:

- Pitas, I. (1993). *Parallel Algorithms: for Digital Image Processing, Computer Vision and neural networks*. Wiley.
- Culler, D. (1999). *Parallel Computer Architecture: A Hardware/Software Approach*. Morgan Kaufmann Publishers.

Sugerencias didácticas	Mecanismos de evaluación de aprendizaje de los alumnos
Exposición oral (X)	Exámenes parciales (X)
Exposición audiovisual (X)	Examen final escrito (X)
Ejercicios dentro de clase ()	Trabajos y tareas fuera del aula (X)
Ejercicios fuera del aula ()	Exposición de seminarios por los alumnos ()
Seminarios (X)	Participación en clase ()
Lecturas obligatorias ()	Asistencia ()
Trabajo de Investigación ()	Seminario (X)
Prácticas de taller o laboratorio (X)	Otros: La evaluación de los estudiantes se realizará sobre la base de las siguientes actividades: Tareas 20%, Exámenes 50% y Proyecto final 30%.
Prácticas de campo ()	
Otras: Las sesiones teóricas estarán complementadas con lecturas de artículos relativas a los temas tratados y con tareas periódicas. Los estudiantes tendrán que presentar exposiciones periódicas en la clase. Para desarrollar el trabajo práctico se utilizarán una plataforma multiusuario de procesamiento paralelo MIMD y <i>clusters</i> de computadoras.	

Especialización en Cómputo de Alto Rendimiento



Línea de investigación:

Cómputo de Alto Rendimiento.

Perfil profesiográfico:

Académico con nivel mínimo de especialista en el área, deseablemente doctorado, en un área afin a la ciencia e ingeniería de la computación. Con experiencia docente comprobable.

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: PROGRAMACIÓN AVANZADA			
Clave:	Semestre(s): 1	Campo de Conocimiento: Cómputo de Alto Rendimiento	No. Créditos: 6
Carácter: Obligatorio		Horas	Horas por semana
Tipo: Teórica		Teoría: 2	Práctica: 1
Modalidad: Curso		Duración del programa: Semestral	

Seriación: Sin Seriación (X) Obligatoria () Indicativa () Actividad académica antecedente: Actividad académica subsecuente: Objetivo general: El alumno dominará los conceptos y técnicas avanzadas de programación, centrando la atención en el paradigma de programación orientada a objetos, concurrencia y el modelo cliente-servidor. Los conceptos se implementarán con el lenguaje de modelado unificado UML. Objetivos específicos: Al término del curso el alumno será capaz de aplicar: <ol style="list-style-type: none"> 1. Los distintos tipos de lenguajes que existen para programar. 2. Los conceptos básicos de programación orientada a objetos. 3. Los conceptos básicos de concurrencia y sincronización, así como los de distribución de objetos en red. 4. El diseño y programación de una aplicación en internet (<i>WEB</i>) utilizando la arquitectura Modelo-Vista-Controlador (MVC), el lenguaje de programación Java y acceso a base de datos.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Introducción a tipos de lenguajes de programación	3	0
2	Conceptos de programación orientada a objetos	12	0
3	Concurrencia y distribución	7	8
4	Programación de aplicaciones <i>WEB</i>	10	8
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Introducción a tipos de Lenguajes de Programación 1.1. Lenguajes compilados. 1.2. Lenguajes interpretados. 1.3. Lenguajes de <i>script</i> . 1.4. Lenguajes declarativos. 1.5. Lenguajes visuales.
2	Conceptos de Programación Orientada a Objetos 2.1. Clases. 2.2. Objetos. 2.3. Herencia. 2.4. Relaciones de uso. 2.5. Generadores. 2.6. Iteradores. 2.7. Manejo de excepciones. 2.8. Paquetes (módulos). 2.9. Interfaces.
3	Concurrencia y Distribución

Especialización en Cómputo de Alto Rendimiento

	3.1. Hilos de control (<i>threads</i>). 3.2. Sincronización. 3.3. Tipos de datos para la concurrencia (colas concurrentes). 3.4. Conexión a servidores. 3.5. Invocación a métodos remotos. 3.6. Arquitectura cliente-servidor.
4	Programación de aplicaciones <i>WEB</i> 4.1. La arquitectura MVC (<i>Model-View-Controller</i>). 4.2. <i>Servlets</i> y <i>JSP's</i> . 4.3. Conexión a base de datos.

Bibliografía básica:

- Booch, G., Maksimchuk, R.A., Engel, M.W., Young, B.J., Conallen, J. and Houston, K.A. (2007). *Object-Oriented Analysis and Design with Applications*. Addison-Wesley, 3rd edition.
- Booch, G., Rumbaugh, J. and Jacobson, I. (2005). *Unified Modeling Language User Guide*, Addison-Wesley.
- Horstmann, C.S. and Cornell, G. (2005). *Core Java 2: Vol. II Advanced Features*. Pearson Prentice Hall.
- Robinson, J.A. (2004). *Software design for engineers and scientists*, Newnes.
- Smith, M. (2000). *Java: an Object-Oriented Language*, MacGraw-Hill.
- Kernigan, B.W. and Plauger, P.J. (1978). *The elements of programming style*, McGraw-Hill, 2nd edition.

Bibliografía complementaria:

- Aho, A.V., Lam, M.S., Sethi, R. and Ullman, J.D. (2007). *Compilers, principles, techniques & tools*. Addison Wesley, 2nd edition.
- Herlihy, M. and Shavit, N. (2008). *The Art of Multiprocessor Programming*. Morgan Kaufmann.
- Kernighan, B.W. and Pike, R. (1999). *The Practice of Programming. Addison-Wesley Professional Computing Series*. Addison-Wesley.
- Martelli, A. (2006). *Python in a Nutshell*. O'Reilly Media, Inc., 2nd edition.
- McConnell, S. (2004). *Code Complete*. Microsoft-Press 2nd edition.
- Jackson, M.A. (1975). *Principles of Programming Design*. Academic Press.
- Kurniawn, B. (2002). *Java for de Web with Servlets, JSP, and EJB: A Developer's Guide to J2EE Solutions: A Developer's Guide to Scalable Solutions*. New Riders Publishing.
- Williamson A.R. and Moran, C.L. (1997). *Java Database Programming: Servlet & JDBC*. Prentice Hall.

Sugerencias didácticas		Mecanismos de evaluación de aprendizaje de los alumnos	
Exposición oral	(X)	Exámenes parciales	(X)
Exposición audiovisual	(X)	Examen final escrito	(X)
Ejercicios dentro de clase	(X)	Trabajos y tareas fuera del aula	(X)
Ejercicios fuera del aula	(X)	Exposición de seminarios por los alumnos	()
Seminarios	(X)	Participación en clase	(X)
Lecturas obligatorias	(X)	Asistencia	()
Trabajo de investigación	()	Seminario	()
Prácticas de taller o laboratorio	(X)	Otros:	
Prácticas de campo	()		
Otras			

Línea de investigación:



Cómputo de Alto Rendimiento.

Perfil profesional:

Académico con nivel mínimo de especialista en el área, deseablemente doctorado, en un área afín a la ciencia e ingeniería de la computación. Con experiencia docente comprobable.

SEGUNDO SEMESTRE

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: SEMINARIO DE CÓMPUTO DE ALTO RENDIMIENTO				
Clave:	Semestre(s): 2	Campo de Conocimiento: Cómputo de Alto Rendimiento		No. Créditos: 12
Carácter: Obligatorio	Horas		Horas por semana	Horas al Semestre
Tipo: Práctica	Teoría: 0	Práctica: 6	6	96
Modalidad: Seminario			Duración del programa: Semestral	

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente:
Actividad académica subsecuente:
Objetivo general:
El alumno desarrollará sus habilidades a través de un proyecto interdisciplinario que sea cercano a sus intereses profesionales en el cual analizará, diseñará e implementará alguna aplicación de Cómputo de Alto Rendimiento. Lo anterior con base a las herramientas metodológicas aprendidas en los cursos anteriores. El alumno será capaz de identificar las consecuencias bajo un marco de referencia ético en la perspectiva de su responsabilidad como un profesionista altamente capacitado en el uso del cómputo de alto rendimiento,
Objetivos específicos:
Al término del curso el alumno será capaz de:
<ol style="list-style-type: none"> 1. Realizar la programación de procesos paralelos en MIMD. 2. Realizar la programación de procesos paralelos en <i>clusters</i>. 3. Realizar la construcción de algoritmos paralelos. 4. Realizar proyectos de aplicación. 5. Analizar bajo la perspectiva ética de la responsabilidad de un programador el uso de sistemas complejos de cómputo.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Cómputo de Alto Rendimiento	0	96
Total de horas:		0	96
Suma total de horas:		96	

Contenido Temático	
Unidad	Tema y Subtemas
	Elaboración de un proyecto enfocado al Cómputo de Alto Rendimiento, acordado entre el alumno y su tutor principal.

Bibliografía básica:
Específica de cada proyecto.

Bibliografía complementaria:
Específica de cada proyecto.



Sugerencias didácticas	Mecanismos de evaluación de aprendizaje de los alumnos
Exposición oral (X)	Exámenes parciales ()
Exposición audiovisual ()	Examen final escrito ()
Ejercicios dentro de clase ()	Trabajos final (X)
Ejercicios fuera del aula ()	Exposición de seminarios por los alumnos (X)
Seminarios (X)	Participación en clase ()

Especialización en Cómputo de Alto Rendimiento

Lecturas obligatorias	()	Asistencia	()
Trabajo de investigación	()	Seminario	(X)
Prácticas de taller o laboratorio	(X)	Otros: La evaluación de los estudiantes se realizará sobre la base de las siguientes actividades: Tareas 20%, Exámenes 50% y Proyecto final 30%.	
Prácticas de campo	()		
Otros: Presentar un reporte escrito de la implementación diseñada a partir de un análisis exhaustivo de un problema transdisciplinario.			
Línea de investigación: Cómputo de Alto Rendimiento			
Perfil profesiográfico: Académico con nivel mínimo de especialista en el área, deseablemente doctorado, en un área afín a la ciencia e ingeniería de la computación. Con experiencia docente comprobable y capacidad para liderar proyectos de Desarrollo.			

ACTIVIDADES ACADÉMICAS OPTATIVAS

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: PROGRAMACIÓN CONCURRENTES Y DISTRIBUIDA				
Clave:	Semestre: 1 o 2	Campo de Conocimiento: Cómputo de Alto Rendimiento		No. Créditos: 6
Carácter: Optativo		Horas		Horas por semana
Tipo: Teórico-Práctica		Teoría: 2	Práctica: 1	Horas al Semestre 48
Modalidad: Curso			Duración del programa: Semestral	

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente:
Actividad académica subsecuente:
Objetivo general: El alumno describirá y analizará los principios y técnicas de programación paralela, para programar estructuras de datos concurrentes, eficientes y confiables, que permiten construir <i>software</i> para sistemas multiprocesadores.
Objetivos específicos: Al término del curso el alumno será capaz de dominar los principios y técnicas de programación paralela, para programar estructuras de datos concurrentes, eficientes y confiables, que permiten construir <i>software</i> para multiprocesadores.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Introducción	4	0
2	Exclusión mutua	4	0
3	Objetos concurrentes	4	0
4	Fundamentos de memoria compartida	2	2
5	El poder relativo de operaciones primitivas de sincronización	2	2
6	La universalidad del consenso	2	2
7	Candados de spin y contención	4	0
8	Listas ligadas	2	2
9	Colas concurrentes y el problema ABA	2	2
10	Pilas concurrentes y eliminación	2	2
11	Ordenamiento concurrente y coordinación distribuida	2	2
12	Memoria transaccional y temas selectos	2	2
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Introducción 1.1. Objetos concurrentes y sincronización. 1.2. Ejemplos de problemas concurrentes. 1.3. Programación concurrente en Java. 1.4. Arquitecturas de multiprocesadores.
2	Exclusión mutua 2.1. Soluciones para dos hilos. 2.2. Algoritmo de Lamport. 2.3. Sellos de tiempo.
3	Objetos Concurrentes 3.1. Concurrencia y correctez. 3.2. Objetos secuenciales.

Especialización en Cómputo de Alto Rendimiento

	3.3. Definiciones de consistencia (secuencial, linearizabilidad). 3.4. Definiciones de progreso. 3.5. El modelo de memoria de Java.
4	Fundamentos de Memoria Compartida 4.1. Construcciones de registros (seguros, regulares, atómicos). 4.2. Instantáneas atómicas.
5	El Poder Relativo de Operaciones Primitivas de Sincronización 5.1. Números de consenso. 5.2. Registros atómicos. 5.3. Protocolos de consenso. 5.4. Operaciones lee-modifica-escibe.
6	La Universalidad del Consenso 6.1. Objetos de poder universal. 6.2. Construcciones de objetos universales basados en consenso.
7	Candados de <i>Spin</i> y Contención 7.1. Consideraciones de procesadores reales. 7.2. Candados de <i>test & amp; set</i> . 7.3. Retirada exponencial. 7.4. Candados de colas.
8	Listas Ligadas 8.1. Conjuntos basados en listas. 8.2. Razonamiento concurrente. 8.3. Sincronización de granularidad baja y alta, optimista, floja, sin-bloqueo.
9	Colas Concurrentes y el Problema ABA 9.1. Acotadas, no acotadas, recolección de basura y el problema ABA.
10	Pilas concurrentes y eliminación.
11	Ordenamiento concurrente y coordinación distribuida.
12	Memoria transaccional y temas selectos.

Bibliografía básica:

- Maurice Herlihy, M. and Shavit, N. (2008). *The Art of Multiprocessor Programming*. Morgan Kaufmann.
- Ben-Ari, M. (2006). *Principles of Concurrent and Distributed Programming*. Addison-Wesley, 2nd edition.
- Loyola Toal, *Concurrent and Distributed Programming*. Ray University.
- Taubenfeld, G. (2006). *Synchronization Algorithms and Concurrent Programming*. Prentice Hall.

Bibliografía complementaria:

- Goetz, B., Joshua Bloch, J., Joseph Bowbeer, J., Lea, D., Holmes, D., and Peierls, T. (2006). *Java Concurrency in Practice*. Addison-Wesley Professional.
- Hennessy, J.L. and Patterson, D.A. (2007). *Computer Architecture: A Quantitative Approach (Paperback)*. Morgan Kaufmann Publishers, Fourth Edition.
- Attiya, H. and Welch, J. (2004). *Distributed Computing: Fundamentals, Simulations, and Advanced Topics*. Wiley-Interscience, 2nd edition.

Sugerencias didácticas		Mecanismos de evaluación de aprendizaje de los alumnos	
Exposición oral	(X)	Exámenes parciales	(X)
Exposición audiovisual	(X)	Examen final escrito	(X)
Ejercicios dentro de clase	(X)	Trabajos y tareas fuera del aula	(X)
Ejercicios fuera del aula	(X)	Exposición de seminarios por los alumnos	(X)
Seminarios	(X)	Participación en clase	(X)
Lecturas obligatorias	(X)	Asistencia	(X)
Trabajo de investigación	(X)	Seminario	()
Prácticas de taller o laboratorio	(X)	Otros:	
Prácticas de campo	()		
Otras:			



Línea de investigación:

Cómputo de Alto Rendimiento.

Perfil profesiográfico:

Académico con nivel mínimo de especialista en el área, deseablemente doctorado, en un área afín a la ciencia e ingeniería de la computación. Con experiencia docente comprobable.

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: SISTEMAS PARALELOS			
Clave:	Semestre(s): 1 o 2	Campo de Conocimiento: Cómputo de Alto Rendimiento	No. Créditos: 6
Carácter: Optativo	Horas		Horas por semana
Tipo: Teórico-Práctica	Teoría: 2	Práctica: 1	Horas al Semestre 48
Modalidad: Curso		Duración del programa: Semestral	

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente: Actividad académica subsecuente:
Objetivo general: El alumno desarrollará programas que se ejecutarán en más de un procesador al mismo tiempo.
Objetivos específicos: Al término del curso el alumno será capaz de: <ol style="list-style-type: none"> 1. Analizar las principales características de los Sistemas Paralelos. 2. Realizar la programación de modelos de programación para Sistemas Paralelos levemente acoplados. 3. Conocer algunos principios de Ingeniería de <i>Software</i> en Paralelo. 4. Analizar la Dependencia de Datos. 5. Realizar programación de procesos mediante la paralelización de ciclos.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Introducción a Sistemas Paralelos	8	0
2	Modelos de programación para Sistemas Paralelos levemente acoplados	6	2
3	Principios de Ingeniería de <i>Software</i> en Paralelo	6	2
4	Dependencia de Datos	4	4
5	Paralelización de Ciclos	4	4
6	Proyectos de Aplicación	4	4
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Introducción a Sistemas Paralelos 1.1. Arquitecturas de naturaleza paralela y sus características. 1.2. Sistemas con memoria distribuida y sistemas con memoria compartida. 1.3. Sistemas paralelos levemente acoplados.
2	Modelos de Programación para Sistemas Paralelos Levemente Acoplados 2.1. Procesos paralelos y datos privados. 2.2. Comunicación.
3	Principios de Ingeniería de <i>Software</i> en Paralelo 3.1. Descomposición. 3.2. Mapeo (distribución de procesadores y datos). 3.3. Ajuste. 3.4. Otros principios: Balance de la carga computacional, minimización del factor comunicación/cálculo; reducción de cuello de botella secuenciales; escalación del programa.
4	Dependencia de Datos

Especialización en Cómputo de Alto Rendimiento

	4.1. Definiciones. 4.2. Situaciones libres de dependencias. 4.3. Situaciones con dependencias.
5	Paralelización de Ciclos 5.1. Homoparalelización. 5.2. "Scheduling". 5.3. Manejo de cargas computacionales no balanceadas.
6	Proyectos de aplicación.

Bibliografía básica:



- Babb, R.G. II. (1987). *Programming Parallel Processors*. Addison-Wesley Pub. Co. Inc., Reading, MA.
- Ben-Ari, M. (1982). *Principles of Concurrent Programming*. Prentice Hall, Englewood Cliffs, New Jersey.
- Hwang, K. (1988). *Arquitectura de Computadoras y Procesamiento Paralelo*. McGraw-Hill, Madrid, México.
- Pacheco, P. (1996). *Parallel Programming with MPI*. Morgan Kaufmann Publishers.
- Rauber, T. and Rüniger, G. (2013). *Parallel Programming: for multicore and cluster systems*. Springer.

Bibliografía complementaria:

- Pitas, I. (1993). *Parallel Algorithms for Digital Image Processing, Computer Vision*. Wiley.
- Culler, D. (1999). *Parallel Computer Architecture, a Hardware/Software Approach*. Morgan Kaufmann Publishers.

Sugerencias didácticas	Mecanismos de evaluación de aprendizaje de los alumnos
Exposición oral (X)	Exámenes parciales (X)
Exposición audiovisual (X)	Examen final escrito (X)
Ejercicios dentro de clase ()	Trabajos y tareas fuera del aula (X)
Ejercicios fuera del aula ()	Exposición de seminarios por los alumnos ()
Seminarios (X)	Participación en clase ()
Lecturas obligatorias ()	Asistencia ()
Trabajo de investigación ()	Seminario (X)
Prácticas de taller o laboratorio (X)	Otros: La evaluación de los estudiantes se realizará sobre la base de las siguientes actividades: Tareas 20%, Exámenes 50% y Proyecto final 30%.
Prácticas de campo ()	
Otras: Las sesiones teóricas estarán complementadas con lecturas de artículos relativas a los temas tratados y con tareas periódicas. Los estudiantes tendrán que presentar exposiciones periódicas en la clase. Para desarrollar el trabajo práctico se utilizarán una plataforma multiusuario de procesamiento paralelo MIMD y <i>clusters</i> de computadoras.	
Línea de investigación: Cómputo de Alto Rendimiento.	
Perfil profesiográfico: Académico nivel mínimo de especialista en el área, deseablemente doctorado, en un área afín a la ciencia e ingeniería de la computación. Con experiencia docente comprobable.	

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: SISTEMAS EN TIEMPO REAL			
Clave:	Semestre: 1 o 2	Campo de Conocimiento: Cómputo de Alto Rendimiento	No. Créditos: 6
Carácter: Optativo		Horas	Horas por semana
Tipo: Teórico-Práctica		Teoría: 2	Práctica: 1
Modalidad: Curso		3	48
Duración del programa: Semestral			

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente:
Actividad académica subsecuente:
Objetivo general:
Al final del curso el alumno contará con conocimientos suficientes para analizar y diseñar sistemas en tiempo real, tomando en cuenta el desempeño y el estricto apego al manejo de tiempo del sistema de cómputo.
Objetivos específicos:
Al término del curso el alumno será capaz de:
<ol style="list-style-type: none"> 1. Dominar los conceptos básicos en los sistemas de tiempo real. 2. Formalizar la representación de sistemas de tiempo real a través de métodos manejados por eventos. 3. Manejar el tiempo real a partir del manejo de planificadores. 4. Analizar a los sistemas de tiempo real a partir del logro de la planificabilidad para cualquier escenario dentro del contexto del caso de estudio. 5. Diseñar planificadores del tipo multiprocesador. 6. Manipular los procesos tanto en un ambiente mono-procesador como multi-procesador para lograr tiempo real.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Introducción	3	0
2	Representación formal de sistemas en Tiempo Real	5	0
3	Manejo de diversos planificadores	6	4
4	Estudio de la Planificabilidad	6	4
5	Estudio de la Planificabilidad en sistemas Multiprocesador	6	4
6	Implementación	6	4
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Introducción 1.1. Manejo de tiempo real. 1.2. Diferencias entre sistemas de alto desempeño y tiempo real. 1.3. Manejo de la concurrencia para conseguir tiempo real. 1.4. Manejo de tiempo real duro/Manejo de tiempo real suave.
2	Representación Formal de Sistemas en Tiempo Real 2.1. Álgebra de procesos en tiempo real. 2.2. Máquinas de estado finito en tiempo real. 2.3. Redes de Petri en tiempo real.
3	Manejo de Diversos Planificadores 3.1. Planificadores heurísticos. 3.2. Planificadores estáticos.

Especialización en Cómputo de Alto Rendimiento

	3.3. Planificadores dinámicos. 3.4. Planificadores casi dinámicos.
4	Estudio de la Planificabilidad 4.1. Análisis de la planificación. 4.2. Prueba de planificación para distintos algoritmos.
5	Estudio de la Planificabilidad en Sistemas Multiprocesador 5.1. Análisis y Diseño de Planificadores en el contexto distribuido. 5.2. Manejo de los retardos debido a la red de comunicación. 5.3. Manejo de métricas de desempeño para ver el efecto del tiempo real en el contexto distribuido.
6	Implementación 6.1. Manejo de planificadores en un sistema operativo. 6.2. Manejo de planificadores en sistemas concurrentes. 6.3. Manejo de redes de comunicación a partir de planificadores multiproceso.

Bibliografía básica:

- Cheng, A.M.K. (2002). *Real-Time Systems: Scheduling, Analysis, and Verification*. Wiley-Interscience, New Jersey.
- Grehan, R., Moote, R. and Cyliax, I. (1998). *Real-Time Programming: A Guide to 32-Bit Embedded Development*. Addison-Wesley, New York.
- Liu, J.W.S. (2000). *Real-Time Systems*. Prentice Hall, New Jersey.
- Butazzo, G. (2006). *Real-Time Systems*. Wiley, New Jersey.

Bibliografía complementaria:

- Ben-Ari, M. (2006). *Principles of Concurrent and Distributed Programming*. Addison-Wesley, New York.

Sugerencias didácticas	Mecanismos de evaluación de aprendizaje de los alumnos
Exposición oral (X)	Exámenes parciales (X)
Exposición audiovisual (X)	Examen final escrito (X)
Ejercicios dentro de clase (X)	Trabajos y tareas fuera del aula (X)
Ejercicios fuera del aula (X)	Exposición de seminarios por los alumnos (X)
Seminarios (X)	Participación en clase (X)
Lecturas obligatorias (X)	Asistencia (X)
Trabajo de investigación (X)	Seminario (X)
Prácticas de taller o laboratorio ()	Otros:
Prácticas de campo ()	
Otras:	



Línea de investigación:

Cómputo de Alto Rendimiento.

Perfil profesional:

Académico con nivel mínimo de especialista en el área, deseablemente doctorado, en un área afín de la ciencia e ingeniería de la computación. El profesor debe tener un amplio conocimiento en el área de cómputo en específico en el manejo de redes, sistemas, algorítmica, y una comprensión de la programación de sistemas de cómputo. Con experiencia docente comprobable.

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: DISEÑO DE SOFTWARE PARALELO				
Clave:	Semestre: 1 o 2	Campo de Conocimiento: Cómputo de Alto Rendimiento		No. Créditos: 6
Carácter: Optativo		Horas		Horas por semana
Tipo: Teórico-Práctica		Teoría: 2	Práctica: 1	Horas al Semestre 48
Modalidad: Curso			Duración del programa: Semestral	

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente:
Actividad académica subsecuente:
Objetivo general: El alumno desarrollará la programación de aplicaciones multihebra, aprenderá los principales antecedentes y conceptos básicos del diseño de <i>software</i> paralelo. El alumno resolverá problemas clásicos presentes en los cursos de sistemas operativos (productores y consumidores, cinco filósofos, lectores y escritores), así como de otros problemas complementarios de programación y su expresión e implementación utilizando el lenguaje Java.
Objetivos específicos: Al término del curso el alumno será capaz de: <ol style="list-style-type: none"> 1. Dominar los conceptos básicos de diseño de <i>software</i> paralelo. 2. Conocer los principales antecedentes y conceptos básicos de la programación paralela. 3. Programar la resolución práctica de problemas clásicos presentes en los cursos de sistemas operativos (productores y consumidores, cinco filósofos, lectores y escritores). 4. Programar la resolución práctica de problemas complementarios.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Introducción	5	0
2	Programación Secuencial	4	2
3	Programación Concurrente	6	2
4	Semáforos	4	2
5	Monitores	4	2
6	Paso de Mensajes	2	4
7	Diseño de <i>Software</i> Paralelo	7	4
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Introducción 1.1. Descripción de conceptos y tópicos generales: programación secuencial y concurrente, definición de proceso, condiciones de competencia (<i>race conditions</i>), secciones críticas, exclusión mutua, semáforos, monitores, paso de mensajes y llamadas a procedimientos remotos como elementos del procesamiento paralelo y programación distribuida. 1.2. Lenguajes para programación paralela y distribuida. 1.3. Introducción a Java como un lenguaje orientado a objetos para procesamiento paralelo y distribuido.
2	Programación Secuencial 2.1. Antecedentes, descripción y características de la programación secuencial. 2.2. La estructura del procesamiento secuencial, sus ventajas y limitaciones.

Especialización en Cómputo de Alto Rendimiento

	2.3. Desarrollo de ejemplos de programación secuencial en Java.
3	<p>Programación Concurrente</p> <p>3.1. Antecedentes, descripción y características de la programación concurrente. 3.2. El concepto de proceso concurrente y hebra de control. 3.3. Procesamiento multi-hebra y condiciones de competencia entre hebras de control. 3.4. Sincronización de procesos multi-hebra mediante variables y espera activa. 3.5. Secciones críticas, exclusión mutua, seguridad y eventual entrada (<i>liveness</i>). 3.6. Introducción a la programación de hebras de control en Java. 3.7. Productores y consumidores como procesos concurrentes en Java. 3.8. Ventajas y limitaciones de la programación concurrente.</p>
4	<p>Semáforos</p> <p>4.1. Antecedentes y definición del concepto de semáforo. 4.2. Semáforos y exclusión mutua. 4.3. Expresión implícita y explícita de semáforos en Java. 4.4. Desarrollo de ejemplos clásicos de programación en Java.</p>
5	<p>Monitores</p> <p>5.1. Presentación de antecedentes y definición del concepto de monitor. 5.2. Comparación entre monitores y semáforos como elementos de sincronización de procesos concurrentes. 5.3. Variables de condición <i>wait</i> y <i>signal</i>. 5.4. Tipos de "señalización" en monitores. 5.5. Monitores en Java. 5.6. Desarrollo de ejemplos clásicos de programación en Java.</p>
	<p>Paso de Mensajes</p> <p>6.1. Origen y definición de la técnica de paso de mensajes. 6.2. El concepto de canal, envío y recepción de mensajes. 6.3. Paso de mensajes síncrono, asíncrono y condicional. 6.4. Expresión e implementación del paso de mensajes en Java. 6.5. Desarrollo de ejemplos clásicos de programación en Java. 6.6. Origen y definición de la técnica de llamadas a procedimientos remotos. 6.7. El concepto de <i>rendevous</i>. 6.8. El sistema cliente-servidor. 6.9. <i>Redevous</i> condicional. 6.10. Expresión e implementación de las llamadas a procedimientos remotos en Java (<i>Remote Method Invocation</i>). 6.11. Desarrollo de ejemplos clásicos de programación en Java.</p>
7	<p>Diseño de <i>Software</i> Paralelo</p> <p>7.1. Conceptos básicos de procesamiento paralelo y distribuido (granularidad, procesamiento síncrono y asíncrono, memoria compartida y distribuida). 7.2. Tipos de paralelismo: funcional, de datos y actividades. 7.3. Ejemplos de programación paralela y distribuida en Java.</p>

Bibliografía básica:

- Andrews, G.R. (1991). *Concurrent Programming. Principles and Practice*. The Benjamin/Cummings Publishing Company, Inc.
- Hartley, S.J. (1998). *Concurrent Programming: The Java Programming Language*. Oxford University Press.
- Blair-Chappell, S. and Stokes, A. (2012). *Parallel Programming with Intel Parallel Studio XE*. Wrox.
- McCool, M. and Reinders, J. (2012). *Structured Parallel Programming: Patterns for Efficient Computation*. Morgan Kaufman.
- Ortega Arjona, J.L. (2010). *Patterns for Parallel Software Design*. John Wiley & Sons.

Bibliografía complementaria:



Artículos:

- Dijkstra, E.W. Co-operating Sequential Processes. In *Programming Languages* (ed. Genuys), pp. 43-112, Academic Press, 1978.
- Brinch Hansen, P. Structured Multiprogramming. *Communications of the ACM*, Vol. 15, No. 17, July 1972.
- Hoare, C.A.R. Monitors: An Operating System Structuring Concept. *Communications of the ACM*, Vol. 17, No. 10, October 1974.
- Hoare, C.A.R. Communicating Sequential Processes. *Communications of the ACM*, Vol. 21, No. 8, August 1978.

Especialización en Cómputo de Alto Rendimiento

▪ Brinch Hansen, P. Distributed Processes: A Concurrent Programming Concept. <i>Communications of the ACM</i> , Vol. 21, No. 11, November 1978.			
Sugerencias didácticas		Mecanismos de evaluación de aprendizaje de los alumnos	
Exposición oral	(X)	Exámenes parciales	(X)
Exposición audiovisual	(X)	Examen final escrito	(X)
Ejercicios dentro de clase	(X)	Trabajos y tareas fuera del aula	(X)
Ejercicios fuera del aula	(X)	Exposición de seminarios por los alumnos	(X)
Seminarios	(X)	Participación en clase	(X)
Lecturas obligatorias	(X)	Asistencia	(X)
Trabajo de investigación	(X)	Seminario	(X)
Prácticas de taller o laboratorio	()	Otros:	
Prácticas de campo	()		
Otras:			
Línea de investigación:			
Cómputo de Alto Rendimiento.			
Perfil profesiográfico:			
Académico nivel mínimo de Especialista en el área, deseablemente doctorado, en un área afín de la ciencia e ingeniería de la computación. El profesor debe tener un amplio conocimiento en el área de cómputo en específico en el manejo de redes, sistemas, algorítmica, y una comprensión de la programación de sistemas de cómputo. Con experiencia docente comprobable.			

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: MÉTRICAS DE ALTO DESEMPEÑO			
Clave:	Semestre: 1 o 2	Campo de Conocimiento: Cómputo de Alto Rendimiento	
Carácter: Optativo		Horas	Horas por semana
Tipo: Teórico-Práctica		Teoría: 2	Práctica: 1
Modalidad: Curso		Duración del programa: Semestral	
		Horas al Semestre	No. Créditos: 6
		3	48

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente:
Actividad académica subsecuente:
Objetivo general:
El alumno conocerá los conceptos, métricas y herramientas para medir el rendimiento en operaciones numéricas tanto de programas como de equipos de cómputo.
Objetivos específicos:
Al término del curso el alumno será capaz de identificar los factores que favorecen el rendimiento de un programa dado en una arquitectura específica.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Conceptos básicos de evaluación de rendimiento	12	0
2	Rendimiento numérico de equipos de cómputo	6	6
3	Rendimiento numérico de programas seriales	6	4
4	Rendimiento de programas paralelos	8	6
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Conceptos Básicos de Evaluación de Rendimiento 1.1. Unidades y símbolos. 1.2. Métricas y "benchmarks". 1.3. <i>Speed Up</i> .
2	Rendimiento Numérico de Equipos de Cómputo 2.1. Factores de la arquitectura de los procesadores. 2.2. Factores de los sistemas de comunicación. 2.3. "Benchmarks" de bajo nivel y "benchmarks" industriales.
3	Rendimiento Numérico de Programas Seriales 3.1. Herramientas de medición de rendimiento y de perfiles. 3.2. Métodos básicos de optimización.
4	Rendimiento de Programas Paralelos 4.1. Ley de Amdahl. 4.2. <i>Speed Up</i> , <i>Speed Up</i> escalado y eficiencia. 4.3. Granularidad, comunicaciones y balance de carga.



Bibliografía básica:
▪ Hockney, R.W. (1996). <i>The Science of computer benchmarking</i> . SIAM.

Especialización en Cómputo de Alto Rendimiento

<ul style="list-style-type: none"> ▪ Foster, I. (1995). <i>Designing and building parallel programs</i>. Addison-Wesley. ▪ Handy, J. (1998). <i>The Cache memory book</i>. Academic Press. ▪ Blair-Chappell, S. and Andrew Stokes, A. (2012). <i>Parallel Programming with Intel Parallel Studio XE</i>. Wrox.
Bibliografía complementaria:
<ul style="list-style-type: none"> ▪ Origin™ 2000 and Onyx2® Performance Tuning and Optimization Guide. SGI Techpubs Library. http://techpubs.sgi.com ▪ Top500 Supercomputer sites. http://www.super.unam.mx ▪ Standard Performance Evaluation Corporation. http://www.spec.org ▪ High Performance Computing Challenge Benchmark. http://icl.cs.utk.edu/hpcc/

Sugerencias didácticas	Mecanismos de evaluación de aprendizaje de los alumnos
Exposición oral (X)	Exámenes parciales (X)
Exposición audiovisual (X)	Examen final escrito (X)
Ejercicios dentro de clase (X)	Trabajos y tareas fuera del aula (X)
Ejercicios fuera del aula (X)	Exposición de seminarios por los alumnos (X)
Seminarios (X)	Participación en clase (X)
Lecturas obligatorias (X)	Asistencia (X)
Trabajo de investigación (X)	Seminario (X)
Prácticas de taller o laboratorio ()	Otros:
Prácticas de campo ()	
Otras:	
Línea de investigación:	
Cómputo de Alto Rendimiento.	
Perfil profesiográfico:	
Académico nivel mínimo de Especialista en el área, deseablemente doctorado, en un área afin a la ciencia e ingeniería de la computación.	
El profesor debe tener un amplio conocimiento en el área de cómputo en específico en el manejo de redes, sistemas, algorítmica, y una comprensión de la programación de sistemas de cómputo. Con experiencia docente comprobable.	

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: LABORATORIO DE <i>CLUSTERS</i> Y <i>GRIDS</i>			
Clave:	Semestre: 1 o 2	Campo de Conocimiento: Cómputo de Alto Rendimiento	No. Créditos: 6
Carácter: Optativo	Horas		Horas por semana
Tipo: Teórico-Práctica	Teoría: 2	Práctica: 1	Horas al Semestre 48
Modalidad: Laboratorio		Duración del programa: Semestral	

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente:
Actividad académica subsecuente:
Objetivo general: El alumno conocerá las características y formas de uso de equipos de cómputo tipo <i>cluster</i> de alto rendimiento y de infraestructuras de <i>grids</i> computacionales.
Objetivos específicos: El alumno deberá desarrollar aptitudes para la instalación de diversos cúmulos de computadores, así como tener la capacidad para monitorear procesos y administrar el cúmulo de equipos en un ambiente completamente distribuido. El alumno deberá ser capaz construir soluciones en términos de arquitecturas con base a cúmulos que le permitan administrar sistemas de cómputo distribuido.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Arquitecturas de cómputo	8	0
2	<i>Clusters</i>	12	8
3	<i>Grids</i> computacionales	12	8
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Arquitecturas de Cómputo 1.1. Clasificación de <i>Flynn</i> . 1.2. Memoria compartida y memoria distribuida. 1.3. Topologías de conexión. 1.4. Evolución de las arquitecturas paralelas.
2	<i>Clusters</i> 2.1. Tecnologías típicas de <i>clusters</i> . 2.2. Diseño y administración de <i>clusters</i> . 2.3. Ambientes de compilación y ejecución. 2.4. Talleres.
3	<i>Grids</i> Computacionales 3.1. Historia y conceptos. 3.2. <i>Globus Toolkit</i> . 3.3. <i>gLite</i> . 3.4. Infraestructuras y " <i>testbeds</i> ". 3.5. Talleres.

Bibliografía básica:
▪ Pfister, G.F. (1998). <i>In search of Clusters</i> . Prentice Hall.

Especialización en Cómputo de Alto Rendimiento

- Foster, I. and Kesselman, C. (2004). *The Grid 2*. Elsevier.
- Kirk, D.B. and Hwu, W.W. (2013). *Programming Massively Parallel Processors*. 2nd edition. A Hands-on Approach, Morgan Kaufmann.
- Cook, S. (2012). *CUDA Programming: A Developer's Guide to Parallel Computing with GPUs (Applications of GPU Computing Series)*. Morgan Kaufmann.

Bibliografía complementaria:

- The Globus Alliance. <http://www.globus.org>
- The Beowulf Project. <http://www.beowulf.org>
- E-science grid facility for Europe and Latin America. <http://www.eu-eela.org>
- Pacific Rim Applications and Grid Middleware Assembly <http://pragma.sdsc.edu/>
- Open Science Grid. <http://www.opensciencegrid.org/>

Sugerencias didácticas	Mecanismos de evaluación de aprendizaje de los alumnos
Exposición oral (X)	Exámenes parciales (X)
Exposición audiovisual (X)	Examen final escrito (X)
Ejercicios dentro de clase (X)	Trabajos y tareas fuera del aula (X)
Ejercicios fuera del aula (X)	Exposición de seminarios por los alumnos (X)
Seminarios (X)	Participación en clase (X)
Lecturas obligatorias (X)	Asistencia (X)
Trabajo de investigación (X)	Seminario (X)
Prácticas de taller o laboratorio ()	Otras:
Prácticas de campo ()	
Otros:	

Línea de investigación:



Cómputo de Alto Rendimiento.

Perfil profesiográfico:

Académico nivel mínimo de especialista en el área, deseablemente doctorado, en un área afín a la ciencia e ingeniería de la computación.

El profesor debe tener un amplio conocimiento en el área de cómputo en específico en el manejo de redes, sistemas, algorítmica, y una comprensión de la programación de sistemas de cómputo. Con experiencia docente comprobable.

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: SUPERCÓMPUTO			
Clave:	Semestre: 1 o 2	Campo de Conocimiento: Cómputo de Alto Rendimiento	No. Créditos: 6
Carácter: Optativo	Horas		Horas por semana
Tipo: Teórico-Práctica	Teoría: 2	Práctica: 1	Horas al Semestre 48
Modalidad: Curso		Duración del programa: Semestral	

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente: Actividad académica subsecuente:
Objetivo general: El alumno podrá identificar la arquitectura de las supercomputadoras para la ejecución de programas seriales y paralelos, con la creación de ejercicios en equipos. Además, construirá programas en lenguajes C/C++ orientados a la ciencia, la ingeniería y el Cómputo de Alto Rendimiento. Identificará la sintaxis, ventajas y desventajas de las aplicaciones usando CUDA C.
Objetivos específicos: Al término del curso el alumno será capaz de: <ol style="list-style-type: none"> 1. Manejar las herramientas necesarias para la programación y ejecución de procesos en supercomputadoras. 2. Manejar un sistema operativo de supercomputadoras. 3. Elaborar diseño de <i>software</i> en sistemas de supercómputo. 4. Conocerá la forma de implementar aplicaciones en paralelo a través de CUDA C.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Introducción al Supercómputo	4	0
2	Paradigmas de programación y Desarrollo de Aplicaciones	4	0
3	Construcciones en C/C++	1	4
4	Programación en Paralelo	4	2
5	Manejo de hilos	2	4
6	Memoria Constante y eventos	4	0
7	Factores que afectan el Alto Rendimiento	4	0
8	Técnicas avanzadas	2	2
9	Optimización y el compilador	4	0
10	Infraestructura	3	4
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Introducción al Supercómputo 1.1. Tipos de supercomputadoras. 1.2. Ventajas del sistema de colas. 1.3. Retos del cómputo de alto rendimiento. 1.4. Importancia de los sistemas de medición.
2	Paradigmas de Programación y Desarrollo de Aplicaciones 2.1. Programación orientada a objetos. 2.2. Programación genérica. 2.3. Ambiente de desarrollo. 2.4. Un primer programa.

Especialización en Cómputo de Alto Rendimiento

	2.5. Consultas y propiedades de los GPU's.
3	Construcciones en C/C++ 3.1. Clases y objetos. 3.2. Herencia y polimorfismo. 3.3. Plantillas (<i>templates</i>) y polimorfismo.
4	Programación en Paralelo 4.1. Manejo de errores.
5	Manejo de Hilos 5.1. Bloques. 5.2. Tipos de memoria. 5.3. Sincronización.
6	Memoria Constante y Eventos 6.1. Medición del rendimiento de las aplicaciones.
7	Factores que Afectan el Alto Rendimiento 7.1. Análisis y diseño. 7.2. Constructores y destructores. 7.3. Herencia y composición. 7.4. Argumentos y valor de regreso de las funciones. 7.5. <i>Inlining</i> . 7.6. Objetos temporales.
8	Técnicas Avanzadas 8.1. Herencia recursiva con plantillas. 8.2. <i>Traits</i> . 8.3. Metaprogramas. 8.4. Expresiones. 8.5. Bibliotecas numéricas orientadas a objetos y genéricas.
9	Optimización y el Compilador 9.1. Banderas y compilación. 9.2. Depuración y <i>profiling</i> .
10	Infraestructura 10.1. <i>Origin 2000</i> . 10.2. <i>Cluster</i> . 10.3. <i>Alpha Server SC 45</i> . 10.4. Instrucción <i>MAKE</i> .

Bibliografía básica:

- Severance, Ch. and Dowd, K. (1998). *High Performance Computing (RISC Architectures, Optimization & Benchmarks)*. O'Reilly Media.
- Eijkhout, V. (2012). *Introduction to High Performance Scientific Computing*. Lulu.com.
- Vetter, J.S. (2013). *Contemporary High Performance Computing: From Petascale toward Exascale*. Chapman and Hall/CRC.
- Jeannot, E. and Zilinskas, J. (Eds.). (2014). *High-Performance Computing on Complex Environments*. Wiley Series on Parallel and Distributed Computing.
- Hoffman, A.R. et al. (1990). *Supercomputers: directions in technology and applications*. The National Academies Press. ISBN: 0-309-04088-4.
- Hill, M.D., Jouppi, N.P. and Sohi, G.S. (1999). *Readings in computer architecture*. ISBN: 1-55860-539-8.
- MacKenzie, D. (1998). *Knowing machines: essays on technical change*. ISBN: 0-262-63188-1.

Bibliografía complementaria:



- *Parallel Computational Fluid Dynamics. Recent Advances and Future Directions*. (2010). Edited by Rupak Biswas, ISBN: 1-60595-022-X.
- Hacker, H. et al. (2010). Considering GPGPU for HPC Centers: Is It worth the Effort? In *Facing the Multicore-Challenge: Aspects of New Paradigms and Technologies in Parallel Computing*. By Rainer Keller, David Kramer and Jan-Philipp Weiss. ISBN: 3-642-16232-0.
- Vasiliadis, G., Antonatos, S., Polychronakis, M., Markatos, E.P. and Ioannidis, S. (September 2008, Boston, MA, USA). Gnot: High Performance Network Intrusion Detection Using Graphics Processors. *Proceedings of the 11th International Symposium on Recent Advances in Intrusion Detection (RAID)*.

Especialización en Cómputo de Alto Rendimiento

- NVIDIA CUDA Home Page: http://www.nvidia.com/object/cuda_home_new.html
- CUDA LLVM Compiler: <https://developer.nvidia.com/cuda-llvm-compiler>

Sugerencias didácticas	Mecanismos de evaluación de aprendizaje de los alumnos
Exposición oral (X)	Exámenes parciales (X)
Exposición audiovisual (X)	Examen final escrito (X)
Ejercicios dentro de clase (X)	Trabajos y tareas fuera del aula (X)
Ejercicios fuera del aula (X)	Exposición de seminarios por los alumnos (X)
Seminarios (X)	Participación en clase (X)
Lecturas obligatorias (X)	Asistencia (X)
Trabajo de investigación (X)	Seminario (X)
Prácticas de taller o laboratorio ()	Otros:
Prácticas de campo ()	
Otras:	
Línea de investigación:	
Cómputo de Alto Rendimiento.	
Perfil profesiográfico:	
Académico nivel mínimo de especialista en el área, deseablemente doctorado, en un área afín a la ciencia e ingeniería de la computación.	
El profesor debe tener un amplio conocimiento en el área de cómputo en específico en el manejo de redes, sistemas, algorítmica, y una comprensión de la programación de sistemas de cómputo. Con experiencia docente comprobable.	

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: SISTEMAS DISTRIBUIDOS			
Clave:	Semestre: 1 o 2	Campo de Conocimiento: Cómputo de Alto Rendimiento	No. Créditos: 6
Carácter: Optativo	Horas		Horas por semana
Tipo: Teórico-Práctica	Teoría: 2	Práctica: 1	Horas al Semestre 48
Modalidad: Curso		Duración del programa: Semestral	

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente: Actividad académica subsecuente:
Objetivo general: El alumno podrá desarrollar sistemas distribuidos, su arquitectura, diseño, modos de operación y programación.
Objetivos específicos: Al término del curso el alumno será capaz de: <ol style="list-style-type: none"> 1. Presentar los enfoques diversos en procesamiento distribuido y los requerimientos para un procesamiento distribuido. 2. Manejar los estándares en procesamiento distribuido y realizar un sistema operativo de sistemas distribuidos. 3. Elaborar diseño de <i>software</i> en sistemas de cómputo distribuido. 4. Dominar la topología, así como los principios de diseño para interconexión de redes. Ruteo y control de flujo. Protocolos multiplexados. 5. Implementar máquinas virtuales y proceso distribuido. Redes de computadoras de alta velocidad. Estrategias de sistemas de comunicación. La arquitectura de componentes de sistemas distribuidos.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Introducción	4	0
2	Fundamentos de procesamiento distribuido	8	0
3	Diseño de <i>software</i> en sistemas de cómputo	11	4
4	Conectividad	2	4
5	Implementación	7	8
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Introducción 1.1. Revisión de Redes de Computadoras. 1.2. Revisión de Protocolos de comunicación. 1.3. Revisión de Primitivas de Comunicación. 1.4. Revisión de la definición de un sistema distribuido y el tipo de retos con respecto a la algorítmica relacionada.
2	Fundamentos de Procesamiento Distribuido 2.1. Revisión de sistemas operativos, concurrencia y manejo de procesos. 2.2. Revisión de Algoritmos de consenso. 2.3. Revisión de algoritmos de sincronización. 2.4. Manejo de relojes. 2.5. Tolerancia a fallas.
3	Diseño de <i>Software</i> en Sistemas de Cómputo

Especialización en Cómputo de Alto Rendimiento

	3.1. Manejo de los sistemas operativos en un ambiente distribuido. 3.2. Manejo de los sistemas tipo <i>Middleware</i> . 3.3. Análisis y diseño con base en métricas de desempeño.
4	Conectividad 4.1. Manejo de diversas abstracciones como agentes, sistemas cooperativos, sistemas confederados, entre otros. 4.2. Manejo del ruteo y diagramas de flujo. 4.3. Manejo de las rutas de mínima distancia.
5	Implementación 5.1. Prácticas básicas de interconexión. 5.2. Prácticas sobre manejo de <i>Middleware</i> . 5.3. Prácticas sobre manejo de algoritmos de consenso y sincronización. 5.4. Prácticas sobre el manejo de métricas de desempeño.

Bibliografía básica:

- Weitzman, C. (1980). *Distributed micro-minicomputer systems: structure, implementation and application*. Prentice Hall.
- Thurber, K.J. *Tutorial, a pragmatic view of distributed processing systems*. IEEE Computer Society, New York.
- Ooi, B.C. *Efficient query processing in geographic information systems*. Springer, Berlin.
- Lyons, T. *Network computing system tutorial*. Prentice Hall, Englewood Cliffs, New Jersey.
- Bray, O.H. *Distributed database management systems*. Lexington Books, Lexington, MA.
- Coulouris, H. (2005). *Distributed Systems*. Prentice Hall.

Bibliografía complementaria:

- Ben-Ari. (1990). *Concurrent Programming*.
- Ben-Ari. (2006). *Principles of Concurrent and Distributed Programming*.

Sugerencias didácticas	Mecanismos de evaluación de aprendizaje de los alumnos
Exposición oral (X)	Exámenes parciales (X)
Exposición audiovisual (X)	Examen final escrito (X)
Ejercicios dentro de clase (X)	Trabajos y tareas fuera del aula (X)
Ejercicios fuera del aula (X)	Exposición de seminarios por los alumnos (X)
Seminarios (X)	Participación en clase (X)
Lecturas obligatorias (X)	Asistencia (X)
Trabajo de investigación (X)	Seminario (X)
Prácticas de taller o laboratorio ()	Otras:
Prácticas de campo ()	
Otros:	

Línea de investigación:



Seguridad en Cómputo.

Perfil profesiográfico:

Académico con nivel mínimo de especialista en el área, deseablemente doctorado, en un área afín a la ciencia e ingeniería de la computación.

El profesor debe tener un amplio conocimiento en el área de cómputo en específico en el manejo de redes, sistemas, algorítmica, y una comprensión de la programación de sistemas de cómputo. Con experiencia docente comprobable.

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	---	---

Denominación: TEMAS SELECTOS DE CÓMPUTO DE ALTO RENDIMIENTO				
Clave:	Semestre(s): 1 o 2	Campo de Conocimiento: Cómputo de Alto Rendimiento		No. Créditos: 6
Carácter: Optativo		Horas		Horas por semana
Tipo: Teórico-Práctica		Teoría: 2	Práctica: 1	3
Modalidad: Curso		Duración del programa: Semestral		

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente:
Actividad académica subsecuente:
Objetivo general:
Al concluir el curso el alumno será capaz de aplicar las técnicas y conocimientos de un tema especializado y avanzado de Cómputo de Alto Rendimiento.
Objetivos específicos:
Al concluir un curso de temas selectos el alumno:
<ol style="list-style-type: none"> 1. Presentará y analizará los fundamentos teóricos del mismo. 2. Aplicará las técnicas de algún tema avanzado y actual del Cómputo de Alto Rendimiento.

Indice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Fundamentos teóricos	16	8
2	Métodos y técnicas de aplicación	16	8
Total de horas:		32	16
Suma total de horas:		48	



Contenido Temático	
Unidad	Tema y Subtemas
1	Fundamentos teóricos. El contenido será específico de cada curso.
2	Métodos y técnicas de aplicación. El contenido será específico de cada curso.

Bibliografía básica:
<ul style="list-style-type: none"> ▪ Hoffman, A.R. <i>et al.</i> (1990). <i>Supercomputers: directions in technology and applications</i>. The National Academies Press. ISBN: 0-309-04088-4. ▪ Fosdick, L.D. and Jessup, E.R. (1996). <i>Introduction to High Performance Scientific Computing</i>. Scientific and Engineering Computation. ▪ Krause, E. and Jäger, W. (Eds.). (2003). <i>High Performance Computing in Science and Engineering '98: Transactions of the High Performance Computing Center</i>. Springer. ▪ Despotović-Zrakić, M. (2014). <i>Handbook of Research on High Performance and Cloud Computing in Scientific Research and Education</i>. Information Science Reference.
Bibliografía específica de cada curso.
Bibliografía complementaria:
<ul style="list-style-type: none"> ▪ Wolf, W. (2006). <i>High-Performance Embedded Computing: Architectures, Applications, and Methodologies</i>. Morgan Kaufmann.

Especialización en Cómputo de Alto Rendimiento

Bibliografía específica de cada curso.	
Sugerencias didácticas	Mecanismos de evaluación de aprendizaje de los alumnos
Exposición oral (X)	Exámenes parciales ()
Exposición audiovisual (X)	Examen final escrito ()
Ejercicios dentro de clase (X)	Trabajos y tareas fuera del aula ()
Ejercicios fuera del aula (X)	Exposición de seminarios por los alumnos ()
Seminarios (X)	Participación en clase ()
Lecturas obligatorias (X)	Asistencia ()
Trabajo de investigación (X)	Seminario ()
Prácticas de taller o laboratorio ()	Otros:
Prácticas de campo ()	
Otras:	
Línea de investigación:	
Cómputo de Alto Rendimiento.	
Perfil profesiográfico:	
Académico con nivel mínimo de especialista en el área, deseablemente doctorado, en un área afín a la ciencia e ingeniería de la computación. Con experiencia docente comprobable.	

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: MODELACIÓN MATEMÁTICA Y COMPUTACIONAL DE SISTEMAS CONTINUOS			
Clave:	Semestre(s): 1 o 2	Campo de Conocimiento: Cómputo de Alto Rendimiento	No. Créditos: 6
Carácter: Optativo		Horas	Horas por semana
Tipo: Teórico-Práctica		Teoría: 2	Práctica: 1
Modalidad: Curso		3	48
		Duración del programa: Semestral	

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente: Ninguna
Actividad académica subsecuente: Ninguna
Objetivo general:
El alumno estará familiarizado con el planteamiento de los modelos físicos y matemáticos más importantes en Ciencias de la Tierra.
Objetivos específicos:
El alumno será capaz de modelar sistemas continuos complejos utilizando herramientas de cómputo de alto rendimiento con base a las características del planteamiento físico particular.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Formulación axiomática de los modelos básicos	2	0
2	Mecánica de los Sistemas Continuos Clásicos	2	0
3	Mecánica de Sistemas Continuos no Clásicos	2	0
4	Transporte de solutos por un fluido libre	2	0
5	Flujo de un fluido en un medio poroso	4	4
6	Transporte de solutos por un fluido en un medio poroso	4	2
7	Sistemas multifásicos de la Geohidrología	4	4
8	Recuperación mejorada de aceite	4	4
9	Elasticidad lineal	4	0
10	Mecánica de fluidos	4	2
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Formulación Axiomática de los Modelos Básicos 1.1. Física microscópica y macroscópica. 1.2. Cinemática de los sistemas continuos. 1.3. Ecuaciones de balance de propiedades extensivas e intensivas.
2	Mecánica de los Sistemas Continuos Clásicos 2.1. Conservación de masa. 2.2. Balance de <i>momentum</i> lineal. 2.3. Balance de <i>momentum</i> angular. 2.4. Balance de energía cinética. 2.5. Balance de energía interna. 2.6. Bases para el transporte del calor.
3	Mecánica de Sistemas Continuos no Clásicos 3.1. Sistemas multifásicos. 3.2. Transporte de solutos.



Especialización en Cómputo de Alto Rendimiento

	3.3. Modelos básicos de la producción petrolera.
4	Transporte de Solutos por un Fluido Libre 4.1. Ecuación general de transporte de solutos por un fluido libre. 4.2. Procesos de transporte: Advección, difusión y generación de masa. 4.3. Transporte de solutos con interacciones químicas. 4.4. Problemas bien planteados para los modelos de transporte.
5	Flujo de un Fluido en un Medio Poroso 5.1. Ecuaciones básicas de flujo de un fluido a través de un medio poroso. 5.2. El nivel piezométrico y la Ley de Darcy. 5.3. El buen planteamiento de problemas de fluidos en medios porosos.
6	Transporte de Solutos por un Fluido en un Medio Poroso 6.1. Procesos de transporte: advección, difusión y dispersión mecánica. 6.2. Ecuaciones diferenciales que gobiernan el transporte de solutos. 6.3. Problemas de transporte bien planteados.
7	Sistemas Multifásicos de la Geohidrología 7.1. El caso de flujo saturado y el sistema agua-aire. 7.2. Transporte de múltiples especies: caso flujo saturado. 7.3. Condiciones de frontera y problemas bien planteados.
8	Recuperación Mejorada de Aceite 8.1. Formulación axiomática de los modelos de la producción petrolera. 8.2. El modelo de petróleo negro. 8.3. El modelo composicional.
9	Elasticidad Lineal 9.1. Elasticidad Lineal en sólidos. 9.2. Análisis deformación-esfuerzos. 9.3. Materiales isotrópicos. 9.4. Modelos estáticos y modelos dinámicos.
10	Mecánica de fluidos 10.1. Ecuaciones de Navier-Stokes. 10.2. Fluidos incomprensibles y comprensibles. 10.3. La teoría de aguas poco profundas.

Bibliografía básica:
<ul style="list-style-type: none"> ▪ Malvern, L.E. (1960). <i>Introduction to the Mechanics of a Continuous Medium</i>. Prentice Hall. ▪ Ames, W.F. (1977). <i>Numerical Methods for Partial Differential Equations</i>. Academic Press. ▪ Herrera, I. and Pinder, G.F. (2012). <i>Mathematical Modeling in Science and Engineering: An Axiomatic Approach</i>. Wiley. ▪ Allen, M.B. and Pinder, G.F. (1998). <i>Numerical Modeling in Science and Engineering</i>. Wiley.
Bibliografía complementaria:
<ul style="list-style-type: none"> ▪ Lapidus L. and Pinder, G.F. (1982). <i>Numerical Solution of Partial Differential Equations</i>. J. Wiley. ▪ Huyakorn, P.S. and Pinder, G.F. (1983). <i>Computational Methods in Subsurface Flow</i>. Academic Press, Inc.

Sugerencias didácticas	Mecanismos de evaluación de aprendizaje de los alumnos
Exposición oral (X)	Exámenes parciales (X)
Exposición audiovisual (X)	Examen final escrito (X)
Ejercicios dentro de clase (X)	Trabajos y tareas fuera del aula ()
Ejercicios fuera del aula (X)	Exposición de seminarios por los alumnos ()
Seminarios (X)	Participación en clase ()
Lecturas obligatorias (X)	Asistencia ()
Trabajo de investigación (X)	Seminario (X)
Prácticas de taller o laboratorio (X)	Otros:
Prácticas de campo (X)	
Otras:	
Línea de investigación: Modelación Matemática y Computacional.	
Perfil profesional: Académico con nivel mínimo de especialista en el área. Con experiencia docente comprobable.	

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: MODELACIÓN MATEMÁTICA Y COMPUTACIONAL DE SISTEMAS TERRESTRES			
Clave:	Semestre(s): 1 o 2	Campo de Conocimiento: Cómputo de Alto Rendimiento	No. Créditos: 6
Carácter: Optativo		Horas	Horas por semana
Tipo: Teórico-Práctica		Teoría: 2	Práctica: 1
Modalidad: Curso		Duración del programa: Semestral	

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente: Ninguna
Actividad académica subsecuente: Ninguna
Objetivo general:
El alumno aplicará los principales métodos numéricos utilizados para aproximar soluciones a las Ecuaciones Diferenciales Parciales (EDP), provenientes de los modelos continuos de sistemas terrestres. El alumno podrá solucionar las ecuaciones discretas resultantes. Aplicará estas metodologías a problemas de transporte, flujo, propagación de ondas elásticas y sistemas de varias fases. Así mismo, desarrollará programas en un lenguaje orientado a objetos (OO) y usando esquemas para cómputo en paralelo.
Objetivos específicos:
El alumno será capaz de modelar, a través de ecuaciones diferenciales parciales, problemas complejos en el ámbito de sistemas continuos y su representación programada en un lenguaje orientado a objetos.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Introducción	4	0
2	Herramientas básicas	4	4
3	Fundamentos de EDP	4	2
4	Método de Elemento Finito (MEF) para ecuaciones elípticas	5	4
5	Componentes de un programa OO para MEF	5	2
6	MEF para ecuaciones parabólicas	5	2
7	Ejemplos de aplicación	5	2
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Introducción
2	1.1. Notación Herramientas Básicas 2.1. Interpolación. 2.1.1. Polinomios de Lagrange. 2.1.2. Polinomios de Hermite. 2.2. Integración numérica. 2.2.1. Cuadratura Gaussiana. 2.3. Métodos de solución de sistemas lineales. 2.3.1. Factorización LU. 2.3.2. Jacobi y Gauss-Seidel. 2.3.3. Gradiente Conjugado (CGM). 2.3.4. Precondicionadores. 2.3.5. GMRES. 2.4. Programación orientada a objetos.
3	Fundamentos de EDP

Especialización en Cómputo de Alto Rendimiento

	<p>3.1. Planeación general.</p> <p>3.2. Clasificación.</p> <p>3.3. Operador adjunto, simetría, forma de divergencia.</p> <p>3.4. Condiciones de frontera.</p>
4	<p>Método de Elemento Finito (MEF) para Ecuaciones Elípticas</p> <p>4.1. Problema modelo de 1D.</p> <p>4.1.1. Espacios de Sobolev.</p> <p>4.1.2. Formulación variacional.</p> <p>4.1.3. Funciones Chapeaux.</p> <p>4.1.4. Integración numérica.</p> <p>4.1.5. Condiciones de frontera.</p> <p>4.1.6. Solución de sistemas tridiagonales.</p> <p>4.1.7. Estimaciones del error.</p> <p>4.1.8. Consistencia y estabilidad.</p> <p>4.2. Problema modelo en 2D.</p> <p>4.2.1. Métodos de discretización del espacio (Triangulación).</p> <p>4.2.2. Numeración de nodos y elementos.</p> <p>4.2.3. Espacios de elemento finito (Triángulos y Rectángulos).</p> <p>4.2.4. Elemento estándar.</p> <p>4.2.5. Construcción de los sistemas local y global.</p> <p>4.2.6. Solución de los sistemas lineales.</p>
5	<p>Componentes de un Programa OO para MEF</p> <p>5.1. Nodo, elemento, elemento estándar, operador.</p> <p>5.2. Elemento finito.</p> <p>5.3. Sistema local.</p> <p>5.4. Geometría.</p> <p>5.5. Sistema global.</p> <p>5.6. Frontera.</p> <p>5.7. Solver.</p>
6	<p>MEF para Ecuaciones Parabólicas</p> <p>6.1. Esquema explícito.</p> <p>6.2. Esquema implícito.</p> <p>6.3. Esquema de Crank-Nicolson.</p>
7	<p>Ejemplos de Aplicación</p> <p>7.1. Sistemas de una fase.</p> <p>7.2. Sistemas de varias fases.</p> <p>7.3. Medios porosos.</p>

Bibliografía básica:

- Allen, M.B., Herrera, I. and Pinder, G.F. (1988). *Numerical Modeling Science and Engineering*. John Wiley and Sons.
- Herrera I. and Pinder, G.F. (2012). *Mathematical Modeling in Science and Engineering: An Axiomatic Approach*. Wiley.
- Allen, M.B. and Pinder, G.F. (1998). *Numerical Modeling in Science and Engineering*. Wiley.
- Ciarlet, P.G. (2002). *The Finite Element Method for Elliptic Problems (Classics in Applied Mathematics)*. North Holland.

Bibliografía complementaria:

- Ames, W.F. (1977). *Numerical Methods for Partial Differential Equations*. Academic Press.

Sugerencias didácticas	Mecanismos de evaluación de aprendizaje de los alumnos
Exposición oral (X)	Exámenes parciales (X)
Exposición audiovisual (X)	Examen final escrito (X)
Ejercicios dentro de clase (X)	Trabajos y tareas fuera del aula (X)
Ejercicios fuera del aula (X)	Exposición de seminarios por los alumnos ()
Seminarios (X)	Participación en clase (X)
Lecturas obligatorias (X)	Asistencia (X)
Trabajo de investigación (X)	Seminario ()
Prácticas de taller o laboratorio (X)	Otros:
Prácticas de campo (X)	
Otras:	
Línea de investigación:	



Especialización en Cómputo de Alto Rendimiento

Geología.

Perfil profesiográfico:

Académico con nivel mínimo de especialista en el área. Con experiencia docente comprobable.

Especialización en Cómputo de Alto Rendimiento

	UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO PROGRAMA DE ESPECIALIZACIÓN EN CÓMPUTO DE ALTO RENDIMIENTO Programa de actividad académica	
---	--	---

Denominación: COMPUTACIÓN CIENTÍFICA EN PARALELO			
Clave:	Semestre(s): 1 o 2	Campo de Conocimiento: Cómputo de Alto Rendimiento	No. Créditos: 6
Carácter: Optativo		Horas	Horas por semana
Tipo: Teórico-Práctica		Teoría: 2	Práctica: 1
Modalidad: Curso		3	48
		Duración del programa: Semestral	

Seriación: Sin Seriación (X) Obligatoria () Indicativa ()
Actividad académica antecedente: Ninguna Actividad académica subsecuente: Ninguna
Objetivo general: El alumno aplicará las diferentes arquitecturas de cómputo capaces de realizar tareas en paralelo que existen en la actualidad.
Objetivos específicos: El alumno será capaz de diseñar e implementar códigos paralelos, asimismo, aplicará el cómputo en paralelo en la solución numérica de ecuaciones diferenciales parciales usando métodos de descomposición de dominio.

Índice Temático			
Unidad	Tema	Horas	
		Teóricas	Prácticas
1	Naturaleza del cómputo de alto rendimiento	4	2
2	Teoría básica	4	2
3	Métricas	4	2
4	Números de punto flotante y BLAS	5	2
5	Métodos de solución de sistemas lineales	5	2
6	Solución numérica de Ecuaciones Diferenciales Parciales	5	3
7	Herramientas para cómputo en paralelo	5	3
Total de horas:		32	16
Suma total de horas:		48	

Contenido Temático	
Unidad	Tema y Subtemas
1	Naturaleza del Cómputo de Alto Rendimiento 1.1. Necesidad del cómputo en paralelo. 1.2. Arquitecturas paralelas y vectoriales. 1.3. Nuevas tecnologías (GPUs).
2	Teoría Básica 2.1. Gráficas acíclicas dirigidas. 2.2. Complejidad. 2.3. Dependencia de datos.
3	Métricas 3.1. Aceleración. 3.2. Escalabilidad. 3.3. Eficiencia. 3.4. Costo.
4	Números de Punto Flotante y BLAS 4.1. Representación de reales en la computadora. 4.2. Operaciones básicas de álgebra lineal.

Especialización en Cómputo de Alto Rendimiento

	4.3. BLAS1, BLAS2 y BLAS3.
5	Métodos de Solución de Sistemas Lineales 5.1. Directos. 5.2. Iterativos. 5.3. Implementación en paralelo.
6	Solución numérica de Ecuaciones Diferenciales Parciales 6.1. Diferentes tipos de discretizaciones numéricas. 6.2. Métodos clásicos de descomposición de dominio. 6.3. Métodos avanzados de descomposición de dominio.
7	Herramientas para Cómputo en Paralelo 7.1. <i>OpenMP</i> . 7.2. Envío de mensajes con MPI. 7.3. CUDA.

Bibliografía básica:

- Allen, M.B., Herrera, I. and Pinder, G.F. (1988). *Numerical Modeling Science and Engineering*. John Wiley and Sons.
- Chapman, G., Jost, G. and Van der Pas, R. (2008). *Using OpenMP: Portable Shared Memory Parallel Programming*. The MIT Press.
- Chen, Z. (2007). *Reservoir Simulation Mathematical Techniques in Oil Recovery*. SIAM, Philadelphia, USA.
- Foster, I. (1995). *Designing and Building Parallel Programs*. Addison-Wesley, (<http://www-unix.mcs.anl.gov/dbpp/>).
- Gropp, W., Lusk, E. and Skjellum, A. (1999). *Using MPI*. 2nd edition. The MIT Press.
- Herrera, I. and Pinder, G.F. (2010). *Fundamental Concepts in Mathematical Modelling of Engineering and Science*. (En impresión).
- Herrera, I. Theory of Differential Equations in Discontinuous Piecewise-Defined Function. Num. Meth. Part. Diff. Eq., DOI 10.1002/num.20182, 23 (3) pp. 597-639, 2007.
- Herrera, I. New Formulation of Iterative Substructuring Methods Without Lagrange Multipliers Neumann-Neumann and FETI. Num. Meth. Part. Diff. Eq., DOI 10.1002/num.20293, 214 (3) pp. 845-878, 2008.
- Herrera I. and Yates, R. Unified Multipliers-Free Theory of Dual Primal Domain Decomposition Methods. Num. Meth. Part. Diff. Eq., DOI 10.1002/num.20359, 25:552-581, 2009.

Bibliografía complementaria:

- Herrera, I. and Yates, R. The Multipliers-free Domain Decomposition Methods. Num. Meth. Part. Diff. Eq., Online DOI 10.1002/num.20462, Vol. 26, No. 4, pp. 874-905, 2010.
- Karniadis, G.E. & Kirby, R.M. (2003). *Parallel Scientific Computing in C++ and MPI*. Cambridge University Press.
- Scott, L.R., Clark, T. and Bagheri, B. (2005). *Scientific Parallel Computing*. Princeton University Press.
- Shonkwiler, R.W. and Lefton, L. (2006). *An introduction to Parallel and Vector Scientific Computing*. Cambridge University Press.
- Snir, M., Otto, S., Huss-Lederman, S., Walker, D. and Dongarra, J. (1998). *MPI: The Complete Reference*. Volume I y II, MIT Press, (<http://www.netlib.org/utk/papers/mpi-book/mpi-book.html>).

Sugerencias didácticas		Mecanismos de evaluación de aprendizaje de los alumnos	
Exposición oral	(X)	Exámenes parciales	(X)
Exposición audiovisual	(X)	Examen final escrito	(X)
Ejercicios dentro de clase	(X)	Trabajos y tareas fuera del aula	(X)
Ejercicios fuera del aula	(X)	Exposición de seminarios por los alumnos	(X)
Seminarios	(X)	Participación en clase	(X)
Lecturas obligatorias	(X)	Asistencia	(X)
Trabajo de investigación	(X)	Seminario	(X)
Prácticas de taller o laboratorio	(X)	Otros:	
Prácticas de campo	(X)		
Otras:			
Línea de investigación:			
Perfil profesional:			
Académico con nivel mínimo de especialista en el área. Con experiencia docente comprobable.			