



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**  
Escuela Nacional de Estudios Superiores Unidad Juriquilla



**ENES**  
JURIQUILLA

Plan de Estudios de la  
Licenciatura en Tecnología  
Sistema Escolarizado: Modalidad Presencial

Programa de estudios de la asignatura

Programación Paralela

<b>Clave</b>	<b>Semestre</b> 6	<b>Créditos</b> 8	<b>Duración</b>	16 semanas		
			<b>Eje de formación</b>	Profundización		
			<b>Área de profundización</b>	Ciencias de la Computación y Matemáticas		
			<b>Etapas de formación</b>	Intermedia		
<b>Modalidad</b>	Curso (X) Taller (X) Lab ( ) Sem ( )		<b>Tipo</b>	T ( ) P ( ) T/P (X)		
<b>Carácter</b>	Obligatorio ( ) Optativo ( )		<b>Horas</b>			
	Obligatorio E (X) Optativo E ( )					
			<b>Semana</b>		<b>Semestre</b>	
			<b>Teóricas</b>	2	<b>Teóricas</b>	32
			<b>Prácticas</b>	4	<b>Prácticas</b>	64
			<b>Total</b>	6	<b>Total</b>	96
<b>Seriación</b>						
Ninguna (X)						
Obligatoria ( )						
<b>Asignatura antecedente</b>						
<b>Asignatura subsecuente</b>						
<b>Indicativa ( )</b>						
<b>Asignatura antecedente</b>						
<b>Asignatura subsecuente</b>						

**Objetivos generales:**

Al finalizar el curso-taller el alumnado será capaz de construir algoritmos computacionales que hacen uso de programación paralela, empleando para ello sistemas de cómputo de alto



rendimiento. Asimismo, el alumnado será capaz de realizar una evaluación del rendimiento de algoritmos paralelos empleando métricas cuantitativas y análisis de complejidad.

**Objetivos específicos:**

1. El alumnado aplicará el modelo de programación paralela para la resolución de problemas científicos y tecnológicos.
2. El alumnado analizará el funcionamiento de las arquitecturas de alto rendimiento.
3. El alumnado aplicará el modelo de programación paralela con OpenMP para la resolución de problemas complejos.
4. El alumnado utilizará el modelo de programación paralela con MPI para la resolución de problemas complejos.
5. El alumnado aplicará el modelo de programación sobre arquitecturas de GPU para la resolución de problemas complejos.
6. El alumnado analizará el rendimiento y complejidad computacional de algoritmos paralelos.

**Índice temático**

	Tema	Horas Semestre	
		Teóricas	Prácticas
1	Introducción a la programación paralela	2	6
2	Arquitecturas de alto rendimiento	2	6
3	Programación con OpenMP	8	16
4	Programación con MPI	8	16
5	Programación sobre arquitecturas GPU	8	14
6	Implementación y evaluación de algoritmos paralelos	4	6
<b>Subtotal</b>		32	64
<b>Total</b>		96	

**Contenido Temático**

Tema	Subtemas
1	<p><b>Introducción a la programación paralela</b></p> <p>1.1 Definiciones y conceptos introductorios</p> <p>1.2 Componentes y clasificación de arquitecturas</p> <p>1.2.1 Antecedentes</p> <p>1.2.2 Taxonomía de Flynn</p> <p>1.3 Escalabilidad</p> <p>1.3.1 Ley de Moore</p> <p>1.3.2 Ley de Amdahl</p> <p>1.3.3 Ley de Gustafson</p> <p>1.4 Tipos de paralelismo</p> <p>1.5 Sistemas de cómputo de alto rendimiento en la actualidad</p>
2	<p><b>Arquitecturas de alto rendimiento</b></p> <p>2.1 Topología de red y modelos de comunicación</p> <p>2.2 Interfaces de programación paralela</p> <p>2.3 Arquitecturas multicore</p> <p>2.4 Arquitecturas manycore</p>

	2.5 Condiciones de Bernstein		
3	<b>Programación con OpenMP</b> 3.1 Modelo de programación paralela con OpenMP 3.2 Directivas de OpenMP 3.3 Funciones auxiliares y variables de entorno 3.4 Métodos básicos de paralelización: Ciclos y secciones paralelas 3.5 Sincronización 3.6 Bloqueos		
4	<b>Programación con MPI</b> 4.1 Modelo de programación paralela de MPI 4.2 Paso de mensajes estándar 4.3 Nombres de grupo y procesos nulos 4.4 Medición de tiempo 4.5 Métodos de paso de mensajes 4.6 Comunicación colectiva		
5	<b>Programación sobre arquitecturas GPU</b> 5.1 Introducción 5.2 Modelo de ejecución paralela 5.3 Jerarquía de memoria 5.4 Transferencia asíncrona de datos 5.5 Cooperación y sincronización 5.6 Aplicaciones		
6	<b>Implementación y evaluación de algoritmos paralelos</b> 6.1 Análisis de rendimiento 6.2 Análisis de complejidad de algoritmos paralelos		
<b>Estrategias didácticas</b>			
<b>Evaluación del aprendizaje</b>			
Exposición	(X)	Exámenes parciales	(X)
Trabajo en equipo	(X)	Examen final	(X)
Lecturas	(X)	Trabajos y tareas	(X)
Trabajo de investigación	(X)	Presentación de tema	( )
Prácticas (taller o laboratorio)	( )	Participación en clases	(X)
Prácticas de campo	( )	Asistencia	(X)
Aprendizaje por proyectos	(X)	Rúbricas	( )
Aprendizaje basado en problemas	(X)	Portafolios	( )
Casos de enseñanza	(X)	Listas de cotejo	( )
Otras (especificar)	( )	Otras (especificar)	( )
<b>Código de conducta</b>			
La conducta del profesorado y alumnado del curso será acorde con los principios y valores especificados en el Código de Ética de la Universidad Nacional Autónoma de México aprobado el 1 de julio del 2015 por el Consejo Universitario, en especial en lo referente a la integridad y honestidad académica. "La integridad y la honestidad académica implican: Citar las fuentes de ideas, textos, imágenes, gráficos u obras artísticas que se empleen en el trabajo universitario, y no sustraer o tomar la información generada por otros o por sí mismo sin señalar la cita			

correspondiente u obtener su consentimiento y acuerdo. No falsificar, alterar, manipular, fabricar, inventar o fingir la autenticidad de datos, resultados, imágenes o información en los trabajos académicos, proyectos de investigación, exámenes, ensayos, informes, reportes, tesis, audiencias, procedimientos de orden disciplinario o en cualquier documento inherente a la vida académica universitaria” (Gaceta UNAM, 30 de julio 2015).

**Perfil Profesiográfico**

Título o Grado	Deberá contar con licenciatura o posgrado en ciencias o ingeniería de la computación o bien alguna otra afín del área de las Ciencias Físico Matemáticas y las Ingenierías.
Experiencia docente	Con experiencia docente en licenciatura y/o en posgrado, preferentemente de tres años impartiendo la asignatura u otra relacionada en el nivel superior.
Otra característica	El curso-taller deberá ser impartido por académicas/os de tiempo completo de la UNAM, o equivalente, que estén activos en investigación sobre temas directamente relacionados con el curso-taller.

**Bibliografía básica**

- Kirk, D. B. & Hwu, W. W. (2016). *Programming Massively Parallel Processors: A Hands-on Approach*. Elsevier Science
- Kurgalin, S. & Borzunov, S. (2019). *A practical approach to High-Performance Computing*. Springer Nature.
- Lorenzon, A. F. Schneider A. C. & Filho, B. (2019). *Parallel Computing Hits the Power Wall: Principles, Challenges, and a Survey of Solutions*. Springer Nature.

**Bibliografía complementaria**

- Hwu, W. W. (2011). *GPU Computing Gems Emerald Edition*. Elsevier Science.
- Stearling, T. Anderson, M. & Brodowicz, M. (2017). *High Performance Computing: Modern Systems and Practices*. Morgan Kaufmann



CONSEJO ACADÉMICO DEL ÁREA DE LAS  
CIENCIAS FÍSICO MATEMÁTICAS  
Y DE LAS INGENIERÍAS