

**Curso Especial**

## Cómputo Científico de Alto Rendimiento

Datos básicos

Semestre	Horas de teoría	Horas de práctica	Horas trabajo adicional estudiante	Créditos
Ambos	3	2	5	10

<b>Objetivos</b>	<p>Al finalizar el curso el estudiante será capaz de entender y aplicar elementos del cómputo científico de alto rendimiento para la resolución eficiente de problemas numéricos en física, química, biología, fisico-química, biofísica, matemáticas e ingeniería. El estudiante será capaz de desarrollar, usar, y adecuar software científico de alto rendimiento en la resolución numérica de estos problemas. También desarrollará la capacidad de implementar códigos de computadora haciendo uso de técnicas numéricas y de paralelización en unidades de procesamiento central (CPUs) y unidades de procesamiento gráfico (GPUs). Esta materia es autocontenida y no requiere seriación.</p>
<b>Temario</b>	<ol style="list-style-type: none"><li>1. Elementos básicos de programación en lenguajes de alto nivel: Fortran 2000, Fortran 95, C, C++, Java, Python.</li><li>2. Desarrollo y uso de software científico y librerías numéricas para la resolución eficiente de problemas numéricos en física, química, biología, fisico-química, biofísica y matemáticas: software hecho en casa, Octave, Python, Boost C++ Libraries, y librerías matemáticas como BLAS, ATLAS, LAPACK, NTL, GSL, GMP, MPFR, y FFTW.</li><li>3. Optimización numérica multivariable utilizando técnicas de descenso de gradiente, algoritmos genéticos, e integración numérica de Monte Carlo.</li><li>4. Cálculo de propiedades estáticas y dinámicas de sistemas de muchas partículas haciendo uso de diversas técnicas de simulación: Monte Carlo, Monte Carlo cinético, dinámica molecular, dinámica browniana, o dinámica disipativa de partículas.</li><li>5. Teoría, desarrollo, e implementación de algoritmos numéricos paralelizados en OPENMP y MPI en unidades de procesamiento central (CPUs), así como en unidades de procesamiento gráfico (GPUs).</li></ol>

**Universidad Autónoma de San Luis Potosí**  
**Posgrado en Ciencias Interdisciplinarias**  
**Facultad de Ciencias**

Métodos y prácticas	Métodos	Clases presenciales de maestro y estudiantes. Exposición del maestro con apoyo de recursos computacionales y audiovisuales. Lectura de libros, artículos, y desarrollo de proyectos por parte de los estudiantes. Diseño e implementación de código en Fortran 2000, Fortran 95, C, C++, Java, o Python. Desarrollo, uso y adecuación de software científico y librerías numéricas. Uso de unidades de procesamiento central (CPUs) y unidades de procesamiento gráfico (GPUs).
	Prácticas	Programación en Fortran 2000, Fortran 95, C, C++, Java, o Python, haciendo uso de OPENMP, MPI, Cuda C u OPENCL para el desarrollo e implementación de técnicas de computo paralelo.
Mecanismos y procedimientos de evaluación	Tareas y proyectos	El curso será evaluado mediante 10% de exposiciones, 30% de tareas, y 60% correspondiente al desarrollo de proyectos que involucren la implementación de código de computadora.
Bibliografía básica de referencia	<ol style="list-style-type: none"> <li>1. Handbook of Open Source Tools, S. Koranne, Springer, 2011.</li> <li>2. Numerical Optimization, J. Nocedal y S. Wright, Springer, 2006.</li> <li>3. Evolutionary Algorithms for Solving Multi-Objective Problems, C. Coello Coello, D. A. Van Veldhuizen, y G. B. Lamont, Springer, 2002.</li> <li>4. Parallel Computing: Architectures, algorithms, and applications; C. Bischof, IOS Press, 2008.</li> <li>5. Cuda by Example: An introduction to General-Purpose GPU Programming, Pearson Education, Inc. 2011.</li> <li>6. Numerical Simulation in Molecular Dynamics: Numerics, Algorithms, Parallelization, Applications : Griebel, S. Knapek, G. Y Zumbusch, Springer, 2007.</li> <li>7. Statistical Mechanics: Theory and Molecular Simulation, M. Tuckerman, Oxford University Press, 2010.</li> </ol>	

**Universidad Autónoma de San Luis Potosí**  
**Posgrado en Ciencias Interdisciplinarias**  
**Facultad de Ciencias**

---

Elaboración y Fecha	Guillermo Iván Guerrero García, 3 de julio de 2015.
---------------------	---