|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Programa académico:** | | Matemática | | | | | | | |
| **Nombre de la asignatura:** | | | Computación Científica II | | | | **Semestre:** | | V |
| **Código asignatura:** |  | | | **Intensidad horaria:**  **(solo presencial)** | 3 | **Créditos académicos:** | | 3 | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Pre-requisitos:** | | | | **Área curricular:** | | | | |
|  | | | | Ciencias Básicas | | | | |
| **Horas por semana:** | | **Presenciales:** | 3 | | **Trabajo Independiente:** | 3 | **Total:** | 6 |
| **Justificación:** | La aparición de computadores en paralelo de bajo costo como los multiprocesadores de escritorio y estaciones de trabajo ha hecho que tales métodos paralelos sean de aplicación general, al igual que los estándares de software para la programación paralela portátil. Este curso toma en cuenta estos nuevos desarrollos y cubre los problemas más tradicionales que abordan la computación de alto rendimiento. Siempre que sea posible, se emplea una vista independiente de la arquitectura de las plataformas y se diseñan algoritmos de manera abstracta, con el fin de dar solución a problemas computacionalmente complejos e intensivos en datos en varias ramas de la ciencia. | | | | | | | |
| **Objetivo general:** | | Reconocer diferentes modelos de programación en paralelo, herramientas de software libre y arquitecturas computacionales relevantes para resolver problemas en ciencia de manera eficiente. | | | | | | |

| **Competencias** | | **Indicador de logro en la asignatura** |
| --- | --- | --- |
| **1. Transversales** | Pensamiento Investigativo | Identificación o formulación de preguntas de investigación |
| Solución de Problemas | Producción de modelos que expliquen la relación entre variables, tanto de tipo cuantitativo como cualitativo |
| Gestión de la información | Búsqueda de fuentes |
| **2.**  **Específicas del programa** | Aplicativas | Validación de modelos. |
| Teóricas | Capacidad de relacionar acontecimientos y maneras de pensar del pasado y del presente para reconocer la dimensión científica y cultural de las matemáticas. |

|  |  |
| --- | --- |
| **Módulo I:** | Introducción a la Computación en Paralelo |

| **Indicadores de logro esperados en el módulo** | **Estrategias de evaluación** |
| --- | --- |
| Del ámbito declarativo:  Conoce algunas herramientas computacionales para realizar cálculos numéricos y figuras. | Resolución de Talleres  Quices |
| Del ámbito procedimental:  Reconoce diferentes tipos de modelos matemáticos. | Resolución de Talleres  Quices  Parcial |
| Del ámbito condicional:  Escribe códigos propios de manera eficiente y organizada. | Resolución de Talleres  Autoevaluación |

| **Semana** | **Intensidad horaria** | **Contenido** | **Estrategias didácticas y de evaluación** | **Trabajo independiente, recursos didácticos y bibliografía** |
| --- | --- | --- | --- | --- |
| 1 | 3 | * Introducción y presentación del curso * Repaso sobre los diferentes tipos de lenguajes de programación | Didácticas: Clase magistral  Evaluación: Quiz | Aula virtual: Lecturas  Laboratorios:  Bibliografía: 1, 5, 6  Otros recursos:  Trabajo independiente: |
| 2 | 3 | * Análisis del desempeño * Optimización de códigos en serie | Didácticas: Clase magistral  Evaluación: Quiz y Taller | Aula virtual: Códigos  Laboratorios:  Bibliografía: 3, 5, 6  Otros recursos:  Trabajo independiente: |
| 3 | 3 | * Rendimiento en Python * Librerías especializadas de optimización: Numexpr, Theano, Numba | Didácticas: Clase magistral  Evaluación: Quiz y Taller | Aula virtual: Notebooks  Laboratorios:  Bibliografía: 2  Otros recursos:  Trabajo independiente: |
| 4 | 3 | * Pruebas de código * Depuración de código | Didácticas: Clase magistral  Evaluación: Quiz y Taller | Aula virtual: Códigos notebook  Laboratorios:  Bibliografía: 1,2, 5  Otros recursos:  Trabajo independiente: |
| 5 | 3 | * Arquitecturas de paralización * Memoria compartida * Memoria distribuida * Escalabilidad | Didácticas: Clase magistral  Evaluación: Quiz y Taller | Aula virtual: Códigos  Laboratorios:  Bibliografía: 2,4,6  Otros recursos:  Trabajo independiente: Proyecto individual |
| 6 | 3 | * Primer examen | Evaluación: Parcial | En clase |

|  |  |
| --- | --- |
| **Módulo II:** | Programación en paralelo |

| **Indicadores de logro esperados en el módulo** | **Estrategias de evaluación** |
| --- | --- |
| Del ámbito declarativo:  Tiene la suficiente capacidad para deducir esquemas numéricos y plantear algoritmos de solución de sistemas simbólicos | Resolución de Talleres  Quices |
| Del ámbito procedimental:  Tiene la capacidad de resolver un problema científico e implementar una solución analítica de manera asistida por el computador. | Resolución de Talleres  Quices  Parcial |
| Del ámbito condicional:  Escribe códigos propios de manera eficiente y organizada. | Resolución de Talleres  Quices  Parcial |

| **Semana** | **Intensidad horaria** | **Contenido** | **Estrategias didácticas y de evaluación** | **Trabajo independiente, recursos didácticos y bibliografía** |
| --- | --- | --- | --- | --- |
| 7 | 3 | * Programación en paralelo en Python I: Threading & Multiprocessing | Didácticas: Clase magistral  Evaluación: Quiz y Taller IA  **Actividad IA:** Crea una red neuronal convolucional capaz de analizar imágenes de números. La etapa de entrenamiento se debe hacer en paralelo, aplicando Threading y Keras en Python. | Aula virtual: Notebooks  Laboratorios:  Bibliografía: 2,4,6  Otros recursos:  Trabajo independiente: Proyecto individual |
| 8 | 3 | * Programación en paralelo con OpenMP | Didácticas: Clase magistral  Evaluación: Quiz y Taller | Aula virtual: Códigos  Laboratorios:  Bibliografía: 6  Otros recursos:  Trabajo independiente: Proyecto individual |
| 9 | 3 | * Programación en Paralelo con MPI | Didácticas: Clase magistral  Evaluación: Quiz y Taller | Aula virtual: Códigos  Laboratorios:  Bibliografía: 6  Otros recursos:  Trabajo independiente: |
| 10 | 3 | * Programación en paralelo en Python II: Mpi4py | Didácticas: Clase magistral  Evaluación: Quiz y Taller | Aula virtual: Códigos y Notebooks  Laboratorios:  Bibliografía: 2,4  Otros recursos:  Trabajo independiente: |
| 11 |  | * Semana de Receso Académico |  |  |
| 12 | 3 | * Segundo Examen | Evaluación: Parcial | En clase |

|  |  |
| --- | --- |
| **Módulo III:** | Aplicaciones y ejemplos |

| **Indicadores de logro esperados en el módulo** | **Estrategias de evaluación** |
| --- | --- |
| Del ámbito declarativo:  Domina las prácticas estándar para presentar resultados en ciencia. | Resolución de Talleres  Quices |
| Del ámbito procedimental:  Tiene la capacidad de escoger una forma adecuada para presentar los resultados de una simulación | Resolución de Talleres  Quices |
| Del ámbito condicional:  Presenta resultados a través de figuras de manera clara y organizada. | Resolución de Talleres  Quices |

| **Semana** | **Intensidad horaria** | **Contenido** | **Estrategias didácticas y de evaluación** | **Trabajo independiente, recursos didácticos y bibliografía** |
| --- | --- | --- | --- | --- |
| 13 | 3 | * Uso de C/C++ con Python (ctypes, cython) | Didácticas: Clase magistral  Evaluación: Quiz y Taller | Aula virtual:  Laboratorios:  Bibliografía: 2  Otros recursos:  Trabajo independiente: Proyecto individual |
| 14 | 3 | * Métodos de Fourier | Didácticas: Clase magistral  Evaluación: Quiz y Taller | Aula virtual:  Laboratorios:  Bibliografía: 3  Otros recursos:  Trabajo independiente: Proyecto individual |
| 15 | 3 | * Operaciones algebraicas * Solución de Sistemas Lineales | Didácticas: Clase magistral  Evaluación: Quiz y Taller | Aula virtual:  Laboratorios:  Bibliografía: 3  Otros recursos:  Trabajo independiente: Proyecto individual |
| 16 | 3 | * Presentaciones Finales | Evaluación: Parcial | En clase |

|  |
| --- |
| **Bibliografía complementaria y fuentes especializadas en internet** |
| [1] Sterling, T., Anderson, M., & Brodowicz, M. (2017). *High Performance Computing: Modern Systems and Practices*. Morgan Kaufmann.  [2] Johansson, R. (2014). Introduction to scientific computing in Python.  [3] Hager, G., & Wellein, G. (2010). *Introduction to high performance computing for scientists and engineers*. CRC Press.  [4] Petersen, W. P., & Arbenz, P. (2004). *Introduction to parallel computing* (Vol. 9). Oxford University Press.  [5] Ananth, G., Anshul, G., George, K., & Vipin, K. (2003). Introduction to parallel computing. *Harlow, UK: Pearson*.  [6] Pacheco, P. (1997). *Parallel programming with MPI*. Morgan Kaufmann. |

|  |  |
| --- | --- |
| **Parcelación de evaluaciones** | |
| **Corte del 40%** | **Corte del 60%** |
| Talleres (16%) | Talleres (20%) |
| Quices (10%) | Quices (10%) |
| Parcial 1 (10%) Semana 6 | Parcial 2 (10%) Semana 12 |
| Base de Datos (4%) | Proyecto Final (14%) Semana 16 |
|  | Base de Datos (6%) |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fuente de información** | **Responsabilidad de la información** | **Fecha diligenciamiento (DD/MM/AAAA)** | **Periodicidad actualización** | **Factor asociado** | **Indicador S.I.I.I.** |
|  |  | 08/08/2023 | Semestral |  |  |