

Presentación de la asignatura

Computación de alto rendimiento

Características

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de prácticas

- ❖ **Carácter:** Obligatoria
- ❖ **Curso:** Segundo (segundo cuatrimestre)
- ❖ **Objetivos formativos:** Computación de altas prestaciones. Hardware y middleware en plataformas de altas prestaciones. Uso de aceleradores GPUs. Virtualización. Clusters y arquitecturas Warehouse Scale Computers.
- ❖ **Carga docente:** 6 créditos ECTS
 - ❖ 60 horas presenciales
 - ❖ 90 horas no presenciales
- ❖ **Programación:** 2 h/sem de teoría + 2 h/sem de prácticas
- ❖ **Área de conocimiento:** Arquitectura y Tecnología de Computadores
- ❖ **Profesor:** Ricardo Moreno
- ❖ **Tutorías:**

Introducción a la asignatura

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

¿Qué es la Computación de Alto Rendimiento (CAR)?

- Uso de sistemas avanzados para procesar grandes volúmenes de datos rápidamente.
- Aplicaciones en inteligencia artificial, climatología, bioinformática y ciberseguridad.

Relación con la vida cotidiana:

- Streaming de contenido.
- Asistentes virtuales.
- Sistemas de recomendación personalizados.

Impacto en diversas industrias:

- Salud, finanzas, manufactura y entretenimiento.

Beneficios para el estudiante:

- Habilidades clave para el mercado laboral.
- Acceso a herramientas tecnológicas avanzadas.

Introducción a la asignatura

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

🔧 **Optimización de procesos:**

- 🔧 Aceleración de tareas complejas mediante procesamiento paralelo.
- 🔧 Reducción de tiempos en aplicaciones críticas.

🔧 **Análisis de grandes volúmenes de datos:**

- 🔧 Manejo de datos en tiempo real para diversos sectores.
- 🔧 Aplicaciones en salud y finanzas.

Competencias de la asignatura

Características

Introducción

Competencias

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

🏠 **Diseño de soluciones escalables:**

- 🏠 Desarrollo de soluciones para centros de datos.

🏠 **Optimización de algoritmos:**

- 🏠 Técnicas de paralelización para reducción de tiempos.

🏠 **Evaluación de rendimiento:**

- 🏠 Comparación entre diferentes arquitecturas de hardware, como CPUs, GPUs y FPGAs

🏠 **Implementación de soluciones en entornos reales:**

- 🏠 Uso de herramientas como OpenMP, MPI y Docker.

Calendario (Parte 1)

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

🔧 **Semana 1: Introducción a la asignatura**

- 🔧 Presentación de la asignatura, competencias y metodología.
- 🔧 Introducción a la Computación de Alto Rendimiento: el uso de la IA en aplicaciones cotidianas, la necesidad del alto rendimiento.

🔧 **Semana 2: Conceptos fundamentales de paralelismo**

- 🔧 Tipos de paralelismo: datos, tareas e instrucciones.
- 🔧 Comparación con sistemas secuenciales.

🔧 **Semana 3: Arquitecturas multiprocesador y multicomputador**

- 🔧 Introducción a sistemas multiprocesador.
- 🔧 Aplicaciones prácticas de arquitecturas multiprocesador.

🔧 **Semana 4: Programación paralela con OpenMP (Parte 1)**

- 🔧 Introducción y configuración del entorno de OpenMP.
- 🔧 Directivas de paralelización básica: bucles y estructuras condicionales.

Calendario (Parte 2)

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

- 🔧 **Semana 5: Programación paralela con OpenMP (Parte 2)**
 - 🔧 Optimización y sincronización de tareas.
 - 🔧 Balanceo de carga y mejora del rendimiento.
- 🔧 **Semana 6: Introducción a la comunicación entre procesos con MPI**
 - 🔧 Introducción a Message Passing Interface (MPI).
 - 🔧 Comunicación punto a punto vs. colectiva.
- 🔧 **Semana 7: Implementación avanzada de MPI**
 - 🔧 Implementación de modelos distribuidos en aplicaciones prácticas.
 - 🔧 Patrones de comunicación eficiente en MPI.
- 🔧 **Semana 8: Control de teoría 1 y repaso**
 - 🔧 Evaluación teórica de los conceptos vistos.
 - 🔧 Resolución de ejercicios y optimización de código en prácticas anteriores.

Calendario (Parte 3)

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

- 🔧 **Semana 9: Middleware para computación distribuida**
 - 🔧 Introducción a middleware y su uso en computación de alto rendimiento.
- 🔧 **Semana 10: Uso de GPUs y FPGAs**
 - 🔧 Integración de hardware acelerador en entornos de alto rendimiento.
 - 🔧 Casos de estudio.
- 🔧 **Semana 11: Evaluación del rendimiento en sistemas paralelos**
 - 🔧 Factores que afectan la escalabilidad.
 - 🔧 Métricas clave para la optimización.
- 🔧 **Semana 12: Optimización de código para alto rendimiento**
 - 🔧 Técnicas de optimización avanzada.
 - 🔧 Mejora del rendimiento en sistemas distribuidos.

Calendario (Parte 4)

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

- 🔧 **Semana 13: Retos actuales en computación de alto rendimiento**
 - 🔧 Desafíos actuales y futuras tendencias.
- 🔧 **Semana 14: Innovaciones tecnológicas en CAR**
 - 🔧 Computación cuántica, edge computing y sostenibilidad.
- 🔧 **Semana 15: Clústeres y Warehouse Scale Computers**
 - 🔧 Introducción a la gestión de clústeres y simulación de infraestructuras.

Metodología de la enseñanza

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

- Clases teóricas y prácticas con resolución de casos reales.
- Proyectos aplicados a problemas del mundo real.
- Evaluaciones parciales y participativas para asegurar el aprendizaje progresivo.
- Uso de herramientas digitales como Moodle y Teams.
- **Recomendación:** Si dispones de portátil, tráelo a clase para realizar búsquedas en internet y mejorar tu comprensión.
- No utilices auriculares o el móvil durante las clases

Software y Herramientas Utilizadas

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

🏠 **Sistemas Operativos y Lenguajes:**

🏠 Sistemas Operativos: Windows y Ubuntu.

🏠 Lenguajes de Programación: Python, C/C++, Bash scripting.

🏠 **Herramientas de Computación:**

🏠 Programación paralela: OpenMP, MPI.

🏠 Virtualización y contenedores: Docker, VirtualBox.

🏠 Computación cuántica: IBM Q Experience.

🏠 Colaboración: Microsoft Teams, Word.

Evaluación de las Prácticas (I)

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

🔧 Estructura de la calificación:

🔧 Bloque 1 (B1): Prácticas e informes prácticos (50%)

- Resolución de problemas prácticos.
- Presentación de informes detallados.

$$\text{Fórmula: } \mathbf{B1} = 0.5 \times \text{NP}$$

🔧 Bloque 2 (B2): Controles de teoría (25%)

- 🔧 Evaluaciones escritas.
- 🔧 Pruebas sobre conceptos clave.

$$\text{Fórmula: } \mathbf{B2} = 0.25 \times \text{NC1} + 0.25 \times \text{NC2}$$

🔧 La nota final (NF) será:

$$\mathbf{NF} = \mathbf{B1} + \mathbf{B2}, \text{ si ambas } \geq 4.$$

- 🔧 Si no se alcanza el **mínimo**: $\text{NF} = \mathbf{min}(\text{B1} + \text{B2}, 4.5).$

Actitudes Esperadas

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

Actitudes Esperadas

- Responsabilidad y compromiso con el aprendizaje.
- Desarrollar **pensamiento crítico**.
- Actitud proactiva en la resolución de problemas.
- Trabajo en casa
- Aprendizaje autónomo.**
- Uso responsable de IA.**

Reflexión final

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

- ¿Utilizas versiones gratuitas o de pago de herramientas de IA?
- ¿Has probado diferentes bots especializados para tareas concretas?
- ¿Qué aplicaciones de IA conoces en la actualidad?
- ¿En qué sectores crees que la IA tiene mayor impacto?
- ¿Qué factores consideras que influyen en el rendimiento de las aplicaciones de IA?

Reflexión final

Características

Introducción

Objetivos

Calendario

Metodología

Herramientas

Evaluación

Actitudes

Reflexión final

Bibliografía

Presentación de
prácticas

- Materiales en “UACloud”
- J.L. Hennessy y D. A. Patterson, 2003. Computer architecture: a quantitative approach, Morgan Kaufmann, 3ª edición.
- W. Stallings, 2006. Organización y Arquitectura de Computadores, Prentice Hall, 7ª edición.
- Qemu. Generic and open-source machine emulator and virtualizer.
<http://www.qemu.org/>
- Proteus. Co-simulador de varios procesadores y SPICE.
<http://www.labcenter.co.uk/>