

Nivel inicial

Introducción a la programación paralela para HPC



Computación
de Alto Desempeño

SADOSKY
CAPACITACIONES



Fundamentación

La Computación de Alto Desempeño (HPC, por sus siglas en inglés) es una tecnología esencial para resolver problemas computacionales de gran escala y complejidad.

Actualmente, la demanda de profesionales con conocimientos en HPC es impulsada por la creciente necesidad de procesar volúmenes masivos de datos y ejecutar simulaciones de alta complejidad en sectores como la investigación científica, la ingeniería, la meteorología y la bioinformática. Desde la modelización climática y la dinámica de fluidos hasta el plegamiento de proteínas y el entrenamiento de modelos de inteligencia artificial a gran escala, las aplicaciones de HPC ocupan un rol cada vez más estratégico en el desarrollo de soluciones avanzadas en ciencia, industria y tecnologías de la información.

Sin embargo, el diseño, la configuración y el uso eficiente de sistemas HPC requieren conocimientos especializados en programación de arquitecturas paralelas, gestión de recursos y optimización de código. Operar estos entornos sin una formación adecuada puede derivar en el uso ineficiente de recursos computacionales costosos.

Para responder a esta necesidad, desde la Fundación Sadosky ofrecemos dos capacitaciones iniciales en HPC en torno a la programación paralela y la administración de clústeres.

Introducción a la programación paralela para HPC es un curso dirigido a profesionales que tengan conocimientos básicos de programación y deseen iniciarse en el mundo de la computación de alto rendimiento.

Objetivo de la capacitación

Comprender los principios fundamentales de la computación paralela y su importancia en entornos de Alto Desempeño (HPC).

Contenidos

Módulo 1: Computación paralela para resolver problemas enormes

Definiciones, usos y conceptos básicos del paralelismo
Paralelismo vs. concurrencia



Arquitecturas para programación paralela (CPU multicore, GPU, clústeres)

Módulo 2: Programación *multithread* con OpenMP.

Directivas básicas y estructura de un programa paralelo con OpenMP
Sincronización y comunicación
Directivas Avanzadas (*shared*, *private* y *reduce*)
Distribución de trabajo (*work-sharing*)

Módulo 3: Programación en clústeres con MPI

Introducción al cómputo distribuido
Fundamentos de MPI
Comunicación punto a punto
Comunicación colectiva

Módulo 4: Programación en GPU

Diferencias conceptuales entre CPU y GPU
Modelo de programación CUDA
OpenCL: alternativa multiplataforma

Organización y cronograma de cursada

La modalidad de la capacitación es virtual y autoasistida. Esto significa que los/as participantes pueden realizar el recorrido en los tiempos que les sean de su preferencia, dentro de los plazos estipulados.

La propuesta se organiza en cuatro módulos de contenidos, videos tutoriales y actividades, además de las secciones de inicio y de cierre. Cada módulo cuenta con actividades de corrección automática que tienen como propósito acompañar el recorrido y darles la posibilidad de repasar y ejercitar los conocimientos adquiridos. El recorrido por estos recursos habilita el acceso a los módulos sucesivos hasta llegar a la evaluación final integradora.

Los/as participantes tendrán a disposición el acceso a los materiales del curso durante un lapso de dos meses desde su matriculación en la propuesta. Pasado ese tiempo, se dará de baja el acceso.



Evaluación y cierre de la cursada

Al finalizar el recorrido se habilitará una instancia de evaluación final para obtener el certificado de cursada correspondiente. Esta evaluación final consta de una serie de preguntas de tipo opción múltiple. Cada participante cuenta con 3 (tres) intentos de una hora para completar la evaluación. Se considera aprobada la cursada si el/la participante responde correctamente el 70% de las preguntas.

En resumen, para acceder al certificado, el/la participante debe:

- Completar la encuesta inicial.
- Recorrer los contenidos y actividades de cada módulo.
- Completar y aprobar la evaluación final.
- Completar la encuesta final.

Bibliografía

- Apptainer Community (2024). Apptainer user guide.
- Hager, G., & Wellein, G. (2010). Introduction to high performance computing for scientists and engineers. CRC Press.
- Hildebrand, D., & Honeyman, P. (2005). Scaling NFSv4 with parallel file systems. Proceedings of the 5th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid'05), 2, 1039–1046.
- Li, T.-L. (2011). A practical guide to building high-performance computing clusters.
- Nemeth, E., Snyder, G., Hein, T. R., Whaley, B., & Mackin, D. (2017). UNIX and Linux system administration handbook (5.^a ed.). Addison-Wesley Professional.
- Ramesh, R. (2024). Scalable systems and software architectures for high-performance computing on cloud platforms. arXiv.
- SchedMD. (s.f.). Slurm workload manager documentation.
- Schmuck, F., & Haskin, R. (2002, 28–30 de enero). GPFS: A shared-disk file system for large computing clusters [Sesión de conferencia]. 1st USENIX Conference on File and Storage Technologies (FAST '02), Monterey, CA, Estados Unidos.
- Sterling, T., Anderson, M., & Brodowicz, M. (2024). High performance computing: Modern systems and practices. Morgan Kaufmann.
- Texas Advanced Computing Center. (s.f.). Lmod: A new environment module system.



Créditos

Equipo de producción de Sadosky Capacitaciones para este curso:

Coordinación de área: Débora Cingolani

Coordinación autoral: Cintia Callamullo León | Ricardo Hugo Medel

Contenidos: Verónica Gil Costa

Diseño gráfico y producción audiovisual: Fabio Viale | Facundo Manini

Didactización, diseño instruccional y maquetación: María Martínez

Corrección: Florencia Acher