Programación con GPUs

Paralelización de algoritmos con CUDA

Segundo Semestre

Fundamentación

Los estudiantes de las carreras de Informática y afines de la Facultad de Ingeniería de la Universidad de Palermo, deben adquirir conocimientos básicos de programación, como así también de ciencias de la computación a través de las materias que componen el tronco común de la carrera como es el caso de "Introducción a la Programación", "Estructura de Datos y Algoritmos", "Algoritmos 1" y "Algoritmos 2". Sin embargo, esas materias hacen foco en la programación secuencial, es decir, la ejecución de un solo hilo de ejecución basado en CPUs ya que ya que estas han sido desarrolladas para minimizar la latencia y maximizar el rendimiento de estas aplicaciones secuenciales. Sin embargo hay determinadas aplicaciones y algoritmos que sólo pueden ser optimizados si se corren de manera paralela. Para ello se requieren tantos CPUs como GPUs, pero se necesitan lenguajes de programación y técnicas de programación diferentes que las que se utilizan en las CPUs.

Esta materia se enfoca en la programación paralela utilizando CUDA (Compute Unified Device Architecture), que es una plataforma que permite la ejecución simultánea de tareas o procesos utilizando múltiples núcleos de procesamiento gráfico (GPU) para realizar cálculos en paralelo permitiendo mejorar el rendimiento de sus aplicaciones. Esto beneficiará a los estudiantes permitiéndoles adquirir habilidades prácticas en la comprensión de algoritmos paralelos y la posibilidad de desarrollar aplicaciones que puedan procesar datos de manera masivamente paralela.

Objetivos

- Comprender los conceptos básicos de la programación paralela y su importancia en la computación moderna.
- Hacer análisis de los algoritmos paralelos y su aplicación en la resolución de problemas complejos.
- Aprender a utilizar CUDA para desarrollar aplicaciones paralelas en C.
- Investigar y aplicar las herramientas y bibliotecas disponibles para la programación paralela en CUDA.
- Desarrollar habilidades prácticas en la implementación de algoritmos paralelos utilizando CUDA.
- Explorar casos de uso y aplicaciones de la programación paralela en diferentes dominios.
- Evaluar el rendimiento y la escalabilidad de las aplicaciones paralelas desarrolladas con CUDA.
- Entender cómo se deben abordar los problemas de programación paralela

0.1 Módulo 1: Introducción a la Programación Paralela

Límites del Paralelismo: Complejidad polinomial, Clases P, Clases de Nick. **Introducción a la Programación Paralela** Evolución de los microprocesadores y la escalabilidad vertical. Importancia del paralelismo masivo y la elección de CUDA C. **Comparación entre CPUs y GPUs** Diferencias en optimización y rendimiento entre CPUs y GPUs. Casos de uso y limitaciones de cada tipo de procesador. **Modelo de Programación CUDA** Descripción del modelo de programación desarrollado por NVIDIA. Ventajas económicas y de rendimiento de utilizar GPUs con CUDA. **Paralelización de Aplicaciones** Beneficios y limitaciones de la paralelización. Ley de Amdahl y su impacto en el rendimiento de programas paralelos. **Desafíos en la Programación Paralela** Complejidad en el diseño de algoritmos paralelos. Sensibilidad a los datos de entrada y límites de acceso a memoria.