

Introducción a la Programación Paralela (CUDA)

Teórica 00 - Antes de Comenzar
Segundo Semestre, 2025

1.1 Introducción

Antes de comenzar, vas a tener que configurar todo el entorno de trabajo para poder correr los ejemplos y hacer pruebas locales en tu computadora. Para ello vas a necesitar tener instalado un cliente de <https://git-scm.com/> para poder clonar el repositorio de la materia y tener toda la información de la materia.

Primero deberás clonar el repositorio <https://github.com/felixcuello/up-materias>. Allí encontrarás todo lo más actualizado de la materia incluyendo las teóricas, prácticas, soluciones y el entorno de desarrollo CUDA. Te recomendamos acceder a la URL y revisar los directorios para saber lo que está disponible.

1.2 Entorno de desarrollo

El entorno de desarrollo CUDA para la materia se puede utilizar:

- **con Docker:** Recomendada, porque no requiere más que tener <https://www.docker.com/products/docker-desktop/> o <https://github.com/abiosoft/colima>, y poder utilizar el ‘Makefile’ provisto por la materia (o sino utilizar los comandos de docker manualmente)
- **con instalación local:** Hacer una instalación local utilizando la guía oficial de <https://docs.nvidia.com/cuda/quick-start-guide/index.html>.

1.2.1 Instalación con Docker

Docker es una herramienta que permite crear, desplegar y ejecutar aplicaciones en contenedores. Un contenedor es una unidad estandarizada que aísla una aplicación y todas sus dependencias (sistema de archivos, bibliotecas, etc.) para que pueda ejecutarse de manera consistente en cualquier entorno. Esto significa que puedes ejecutar la misma aplicación en diferentes sistemas operativos y bibliotecas instaladas en el sistema *host*.

Además, el *overhead* que genera Docker es mínimo ya que no hace una virtualización completa del sistema como puede ser una máquina virtual, sino que utiliza la arquitectura subyacente del sistema operativo. Esto permite que los contenedores sean más ligeros y rápidos, lógicamente, la desventaja es que al no virtualizar no se puede correr cualquier sistema operativo en el contenedor.

Para correr el container de docker deberás instalar alguna de las alternativas para correr docker como son: <https://www.docker.com/products/docker-desktop/> o <https://github.com/abiosoft/colima> para correr el container de docker provisto por la cátedra.

1.2.2 Makefile y comandos de docker

Hemos provisto un archivo **Makefile** para poder levantar el container de docker con facilidad, sin embargo, es posible que tu sistema no tenga el comando **make** instalado. Y nuevamente hay dos opciones aquí. Por un lado se pueden correr los comandos de **docker** por separado, o se puede usar el comando **make** para correr las opciones disponibles en el **Makefile**.

Makefile

Si tu computadora tiene instalado **make** (o si instalaste **make**) estarás en condiciones de ejecutar **make** en la terminal.

```
1 // Este header puede no estar disponible en tu sistema
2 // en ese caso no lo incluyas, e inclui cada header a mano (como siempre)
3 #include <bits/stdc++.h>
4
5 using namespace std;
6
7 int main() {
8     // solucion
9
10    return 0;
11 }
```

1.3 Instalación local

La instalación local no se explica porque ya está totalmente documentada en la documentación oficial de <https://docs.nvidia.com/cuda/cuda-quick-start-guide/index.html>.