

### PRÁCTICA HOLA MUNDO EN GPU

FECHA: 23/01/2022

GRUPO: 4CDV1

EQUIPO: NetPower

Integrantes:

Alcibar Zubillaga Julián
De Luna Ocampo Yanina

### OBJETIVOS

**Parte 1:** Conectarse al servidor remoto

**Parte 2:** Revisar y correr el programa Hola.cu

### ESCENARIO

Para poder correr un programa en paralelo es necesario que se cuente con hardware especializado. Ya que en nuestros hogares y en la mayoría de los laboratorios de cómputo de ESCOM no se cuenta con el software para realizarlo, se hará uso de un servidor externo que contiene el hardware apropiado.

En esta práctica deberás conectarte a ese servidor externo y revisar y correr el programa “Hola.cu”. Este programa muestra una cadena corriendo tanto del CPU como del GPU.

**Nota:** Se utilizará el programa GNS3 para realizar la simulación de esta práctica. Es necesario que este programa esté instalado y corriendo en el equipo en donde se realizará la práctica.

### RECURSOS NECESARIOS PARA REALIZAR LA PRÁCTICA

- 1 computadora personal
- Acceso a internet
- 1 programa cliente SSH\* (Putty: <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>)

### INTRODUCCIÓN

Recordemos que los servidores remotos, que es en donde ejecutamos el programa de clase, proporcionan acceso a objetos y datos compartidos de una organización. Un nivel de acceso de usuario que depende del grupo de seguridad que el administrador asigne en nombre de usuario, que el usuario utiliza para acceder a este servidor. Si trabajamos con un servidor remoto, debe conocer la ubicación del host de administración en el que se ejecuta un servicio de administración de TM1, y debe poder acceder al host de administración desde un sistema [1].

En la práctica, se usa un servidor remoto. Pudimos aprender diversos comandos nuevos para poder acceder a carpetas, copiar y pegar un archivo, salirnos de ellas, entre otros, lo podemos visualizar de mejor forma en la captura de pantalla número 2, están los comandos utilizados en ésta. Asimismo, aprendimos a ejecutar un programa el cual solo se podía editar mediante las flechas del teclado, agregando nuestro nombre y entendiendo que en la línea de comando de helloFromGPU la función la corro desde el GPU, reservo memoria, así como desde el CPU.

A continuación se muestran algunas imágenes que explican parte del proceso que se llevó a cabo en esta práctica.

```
bcruz@labdatos01: ~/hpc_4AV1/estudiantes/DelunaV
login as: bcruz
bcruz@148.204.125.119's password:
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-100-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Wed 23 Feb 2022 12:03:53 PM PST

System load:  0.08      Temperature:   35.0 C
Usage of /:   12.2% of 915.39GB   Processes:    463
Memory usage: 1%      Users logged in: 4
Swap usage:   0%          IPv4 address for emp6s0: 148.204.125.119

 * Super-optimized for small spaces - read how we shrank the memory
   footprint of MicroK8s to make it the smallest full K8s around.

https://ubuntu.com/blog/microk8s-memory-optimisation

3 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

15 updates could not be installed automatically. For more details,
see /var/log/unattended-upgrades/unattended-upgrades.log

Last login: Wed Feb 23 12:02:54 2022 from 148.204.59.59
bcruz@labdatalogin:~$ ssh -X -C 148
-C
bcruz@labdatalogin:~$ ssh -X -C 148.204.125.21
bcruz@148.204.125.21's password:
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-96-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Wed Feb 23 12:04:34 PM PST 2022

System load:  1.12
Usage of /:   24.3% of 1.79TB
Memory usage: 4%
Swap usage:   0%
Temperature:  53.0 C
Processes:    893
Users logged in: 2
IPv4 address for eno1: 148.204.125.21
IPv4 address for eno2: 192.168.1.157
IPv6 address for eno2: 2806:1000:8003:5715:aelf:6bff:fefb:6595
IPv6 address for eno2: 2806:1000:8003:2e2a::1
IPv6 address for eno2: 2806:1000:8003:2e2a:aelf:6bff:fefb:6595
IPv6 address for eno2: fd78:b46a:c5c:9000:aelf:6bff:fefb:6595

 * Super-optimized for small spaces - read how we shrank the memory
   footprint of MicroK8s to make it the smallest full K8s around.

https://ubuntu.com/blog/microk8s-memory-optimisation

32 updates can be applied immediately.
To see these additional updates run: apt list --upgradable
```

Captura de pantalla 1, en donde muestra los comandos para acceder al servidor remoto.

```

*** System restart required ***
Last login: Wed Feb 23 12:04:16 2022 from 148.204.125.119
(base) bcruz@labdatos01:~$ ls
CienciaDatos  anaconda3  cuda  hpc_4AV1
(base) bcruz@labdatos01:~$ cd hpc_4AV1
(base) bcruz@labdatos01:~/hpc_4AV1$ cd estudiantes
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes$ ls
AlcibarJ  DeLunaY  RamirezK  RoblesC  SainzJ  SalinasJ
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes$ cd DeLunaY
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/DeLunaY$ cd..
cd.: command not found
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/DeLunaY$ cd ..
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes$ ls
AlcibarJ  DeLunaY  RamirezK  RoblesC  SainzJ  SalinasJ
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes$ cd ..
(base) bcruz@labdatos01:~/hpc_4AV1$ ls
estudiantes  unidad01  unidad02  unidad03
(base) bcruz@labdatos01:~/hpc_4AV1$ cd unidad01
(base) bcruz@labdatos01:~/hpc_4AV1/unidad01$ ls
01_hola.cu
(base) bcruz@labdatos01:~/hpc_4AV1/unidad01$ cp ~/hpc_4AV1/unidad01/01_hola.cu ~/hpc_4AV1/estudiantes/DeLunaY/
(base) bcruz@labdatos01:~/hpc_4AV1/unidad01$ ls
01_hola.cu
(base) bcruz@labdatos01:~/hpc_4AV1/unidad01$ cd ..
(base) bcruz@labdatos01:~/hpc_4AV1$ ls
estudiantes  unidad01  unidad02  unidad03
(base) bcruz@labdatos01:~/hpc_4AV1$ cd estudiantes
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes$ ls
AlcibarJ  DeLunaY  RamirezK  RoblesC  SainzJ  SalinasJ  benji
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes$ cd DeLunaY
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/DeLunaY$ ls
01_hola.cu
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/DeLunaY$

```

Captura de pantalla 2, se puede visualizar los diversos comandos que se utilizaron para ingresar a las carpetas, salirnos de ellas, copiar y pegar archivos, entre muchos más.

```

GNU nano 4.8                                01_hola.cu
/*
Programa para introducir a la programación en CUDA
Cada hilo gereado imprime el mensaje

Compilar
nvcc 01_hola.cu -o 01_hola.x

*/

#include <stdio.h>
#include <cuda_runtime.h>

__global__ void helloFromGPU()
{
    printf("Hola mundo desde el GPU (dispositivo)\n");
}

int main(int argc, char **argv)
{
    ^G Get Help    ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify    ^C Cur Pos
    ^X Exit        ^R Read File    ^\ Replace    ^U Paste Text  ^T To Spell    ^_ Go To Line

```

Captura de pantalla 3, se muestra la impresión esperada del programa.

### PARTE 1: CONECTARSE AL SERVIDOR REMOTO

UTILIZA UN CLIENTE SSH PARA CONECTARTE AL SERVIDOR LAMBDA.

(SE RECOMIENDA UTILIZAR UN PROGRAMA COMO PUTTY PARA CONECTARSE DE FORMA REMOTA; TAMBIÉN, ES POSIBLE UTILIZAR LA LÍNEA DE COMANDOS DE WINDOWS O LA TERMINAL DE LINUX).

El ingreso al servidor Lambda se hace en dos pasos:

1. Primero hay que ingresar al servidor de login, con el siguiente comando:

```
ssh -X -C bcruz@148.204.125.119
```

2. Desde este servidor se ingresa al servidor Lambda2, con el siguiente comando:

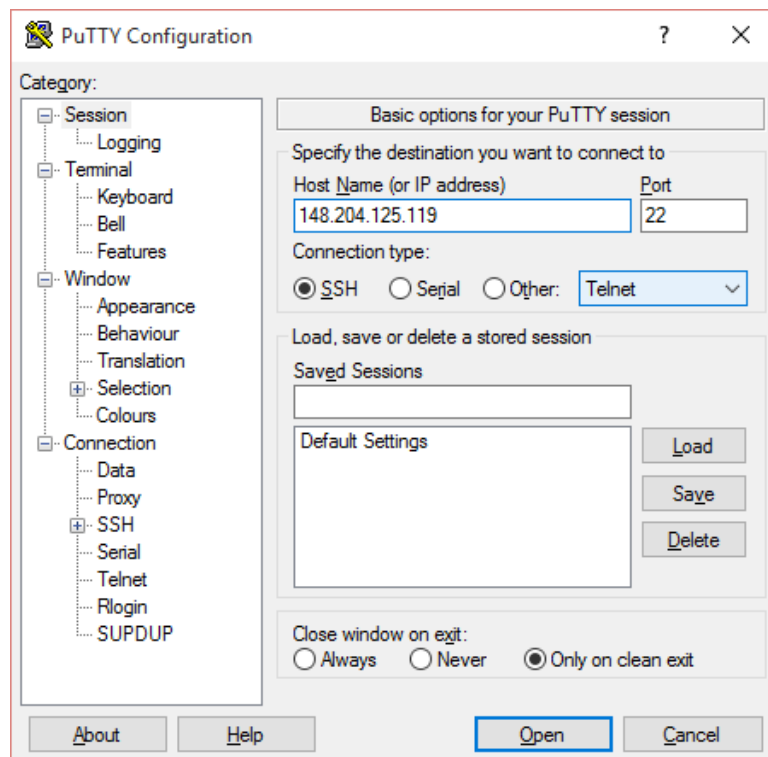
```
ssh -X -C bcruz@148.204.125.21
```

3. El usuario y contraseña para los dos servidores es:

**Usuario:** bcruz

**Contraseña:** xwjf47

SI USASTE UN PROGRAMA CLIENTE SSH COMO PUTTY, USA LOS SIGUIENTES VALORES. DEJA LOS VALORES POR DEFECTO AL REALIZAR LA CONEXIÓN:



INCLUYE AQUÍ LA CAPTURA DE PANTALLA CON LA CONEXIÓN AL SERVIDOR LAMBDA REALIZADA CORRECTAMENTE

#### A) CAPTURAS ALCIBAR ZUBILLAGA JULIÁN



```
C:\Users\alumno>ssh -X -C bcruz@148.204.125.119
"ssh" no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.

C:\Users\alumno>
"-" no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.

C:\Users\alumno>
C:\Users\alumno>bcruz@labdatos01: ~
* Management:  https://landscape.canonical.com
* Support:     https://ubuntu.com/advantage

System information as of Wed 23 Feb 2022 11:54:33 AM PST

System load:  0.01   Temperature:   34.0 C
Usage of /:   12.2% of 915.9GB   Processes:    443
Memory usage: 1%      Users logged in: 1
Swap usage:   0%        Disk space for sigd@0: 148.204.125.119

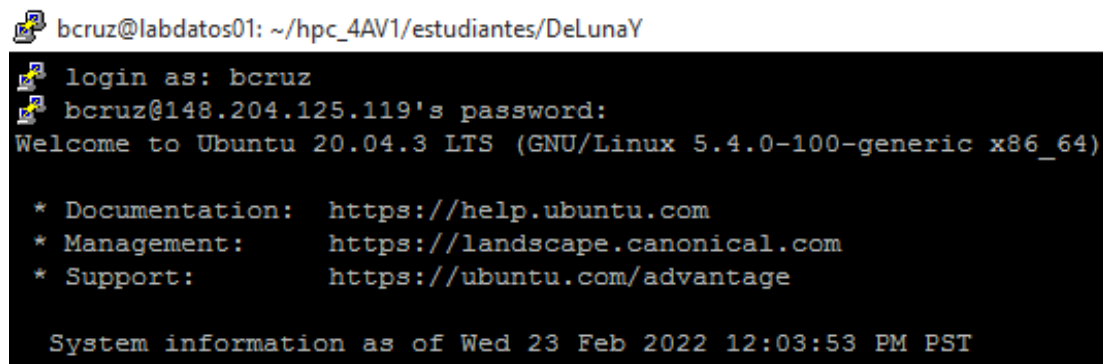
* Super-optimized for small spaces - read how we shrink the memory
  footprint of MicroK8s to make it the smallest full K8s around.
  https://ubuntu.com/blog/microk8s-memory-optimization

3 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

13 updates could not be installed automatically. For more details,
see /var/log/unattended-upgrades/unattended-upgrades.log

Last login: Wed Feb 23 11:51:45 2022 from 148.204.59.59
bcruz@labdatos01:~$
```

### B) CAPTURAS DE LUNA OCAMPO YANINA



```
bcruz@labdatos01: ~/hpc_4AV1/estudiantes/DeLunaY
login as: bcruz
bcruz@148.204.125.119's password:
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-100-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

System information as of Wed 23 Feb 2022 12:03:53 PM PST
```

Captura de pantalla con la conexión al servidor Lambda realizada correctamente

## PARTE 2: REVISAR EL PROGRAMA HOLA GPU Y CORRERLO

Dentro del servidor Lambda. Acceder a la carpeta “hpc\_4AV1” (esa será la carpeta que se usará durante el curso, se recomienda no acceder a las otras carpetas).

Dentro de esa carpeta hay una carpeta llamada “estudiantes” y otras carpetas con las unidades:

```
(base) bcruz@labdatos01:~/hpc_4AV1$ ls -la
total 24
drwxrwxr-x 6 bcruz bcruz 4096 Feb 17 13:49 .
drwxr-xr-x 15 bcruz bcruz 4096 Feb 17 12:29 ..
drwxrwxr-x 8 bcruz bcruz 4096 Feb 17 13:43 estudiantes
drwxrwxr-x 2 bcruz bcruz 4096 Feb 17 12:31 unidad01
drwxrwxr-x 2 bcruz bcruz 4096 Feb 17 13:49 unidad02
drwxrwxr-x 2 bcruz bcruz 4096 Feb 17 13:49 unidad03
(base) bcruz@labdatos01:~/hpc_4AV1$
```

Dentro de la carpeta “estudiantes” encontrarás tu carpeta de trabajo (busca la que corresponde a tu apellido). Entra a tu carpeta y crea una subcarpeta llamada “u01”, el comando es “mkdir u01”. Dentro de esa carpeta trabajaremos los programas de la unidad 1.

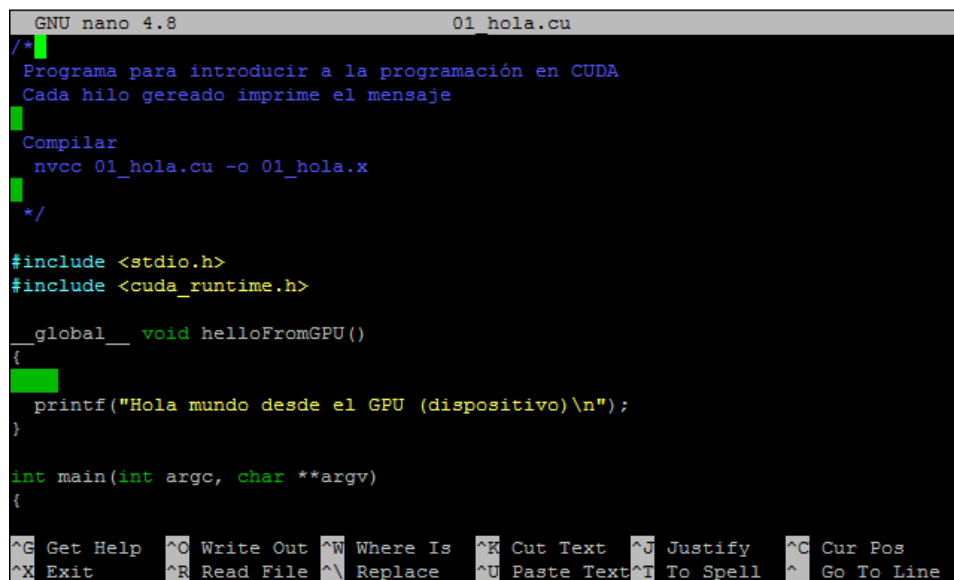
Copiaremos el primer programa a nuestra carpeta de trabajo, ejecuta el siguiente comando:

```
cp ~/hpc_4AV1/unidad01/01_hola.cu ~/hpc_4AV1/estudiantes/<tu_carpeta_de_trabajo>/u01
```

Reemplaza <tu\_carpeta\_de\_trabajo> por el nombre de tu carpeta de trabajo. Este comando copiará el programa “01\_hola.cu” a la subcarpeta u01 de tu carpeta de trabajo. Accede a esa subcarpeta y abre el programa, usa el comando:

```
nano 01_hola.cu
```

Se abrirá el editor “nano” y si todo salió bien mostrará una ventana con la siguiente información:



```
GNU nano 4.8                                01_hola.cu
/*
Programa para introducir a la programación en CUDA
Cada hilo gereado imprime el mensaje

Compilar
nvcc 01_hola.cu -o 01_hola.x

*/

#include <stdio.h>
#include <cuda_runtime.h>

__global__ void helloFromGPU()
{
    printf("Hola mundo desde el GPU (dispositivo)\n");
}

int main(int argc, char **argv)
{
    ^G Get Help    ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify    ^C Cur Pos
    ^X Exit        ^R Read File   ^\ Replace     ^U Paste Text  ^T To Spell   ^_ Go To Line
```

Revisa el programa y analízalo. Es un programa simple en C. Encontrarás una función llamada: `__global__ void helloFromGPU()`. Todo lo que venga dentro de esa función se correrá en el GPU.

Dentro de la función *main* encontrarás una simple instrucción para imprimir una cadena: `printf("Hola mundo desde el CPU (host)!\n");`. Aquí se encuentra la programación tradicional y correrá en el CPU. La instrucción: `helloFromGPU<<<2, 3>>>()`; indica que la función correspondiente se ejecutará en el GPU. Por el momento no le pondremos mucha atención a los parámetros.

Modifica ambas instrucciones `printf` para que impriman “Hola mundo <tu nombre> desde el CPU” y “Hola mundo <tu nombre> desde el GPU (Dispositivo)” cambiando <tu nombre> por tu nombre completo.

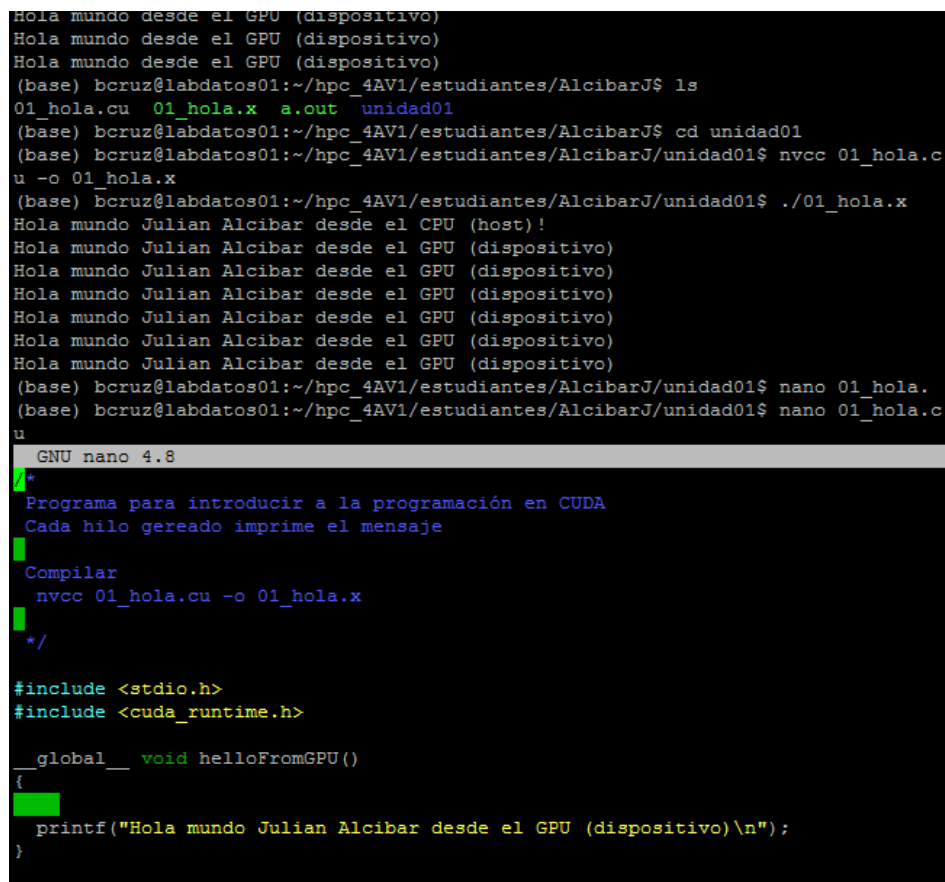
Finalmente, compila y ejecuta el programa, usa los siguientes comandos:

```
nvcc 01_hola.cu -o 01_hola.x
```

```
./01_hola.x
```

INCLUYE AQUÍ LA CAPTURA DE PANTALLA CON EL RESULTADO DEL PROGRAMA EJECUTÁNDOSE

### A) CAPTURAS ALCIBAR ZUBILLAGA JULIÁN



```
Hola mundo desde el GPU (dispositivo)
Hola mundo desde el GPU (dispositivo)
Hola mundo desde el GPU (dispositivo)
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/AlcibarJ$ ls
01_hola.cu  01_hola.x  a.out  unidad01
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/AlcibarJ$ cd unidad01
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/AlcibarJ/unidad01$ nvcc 01_hola.cu -o 01_hola.x
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/AlcibarJ/unidad01$ ./01_hola.x
Hola mundo Julian Alcibar desde el CPU (host)!
Hola mundo Julian Alcibar desde el GPU (dispositivo)
Hola mundo Julian Alcibar desde el GPU (dispositivo)
Hola mundo Julian Alcibar desde el GPU (dispositivo)
Hola mundo Julian Alcibar desde el GPU (dispositivo)
Hola mundo Julian Alcibar desde el GPU (dispositivo)
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/AlcibarJ/unidad01$ nano 01_hola.cu
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/AlcibarJ/unidad01$ nano 01_hola.cu
GNU nano 4.8
/*
Programa para introducir a la programación en CUDA
Cada hilo gereado imprime el mensaje

Compilar
nvcc 01_hola.cu -o 01_hola.x
*/

#include <stdio.h>
#include <cuda_runtime.h>

__global__ void helloFromGPU()
{
    printf("Hola mundo Julian Alcibar desde el GPU (dispositivo)\n");
}
```

## B) CAPTURAS DE LUNA OCAMPO YANINA

bacruz@labdatos01: ~/hpc\_4AV1/estudiantes/DeLunaY

```
GNU nano 4.8
/*
Programa para introducir a la programación en CUDA
Cada hilo gereado imprime el mensaje
Compilar
nvcc 01_hola.cu -o 01_hola.x
*/

#include <stdio.h>
#include <cuda_runtime.h>

nvcc 01_hola.cu -o 01_hola.x
__global__ void helloFromGPU()
{
    printf("Hola mundo Yanina De Luna Ocampo desde el GPU (dispositivo)\n");
}

int main(int argc, char **argv)
{
    printf("Hola mundo Yanina De Luna Ocampo desde el CPU (host)!\n");

    helloFromGPU<<<2, 3>>>>();
    cudaDeviceReset();
    return 0;
}
```

Captura de pantalla con la ejecución del programa

## PARTE 3: CUESTIONARIO

EN EL CASO DE ESTE PROGRAMA EN PARTICULAR, ¿QUÉ SE EJECUTÓ EN PARALELO?

En el código hay una línea que dice “helloFromGPU<<<2, 3>>>>()”,” ahí es en donde está el cómputo paralelo, la función la corro desde el GPU, reservo memoria, asimismo desde el CPU. Lo que significan los números frente a esta función es:

Corro en 3 núcleos y corro la función 6 veces.



Uso 2, 3 de cada uno de ellos.

1 núcleo: 2 3

2 núcleo: 2 3

3 núcleo: 2 3

Son 6 hilos diferentes, si vamos cambiando los número, se van alterando el número de núcleos finales. Podemos ver el ejemplo de que de arriba, teniendo: <<<2, 3>>>

```
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/DeLunaY$ nvcc 01_hola.cu -o 01_hola.x
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/DeLunaY$ ./01_hola.x
Hola mundo Yanina De Luna Ocampo desde el CPU (host)!
Hola mundo Yanina De Luna Ocampo desde el GPU (dispositivo)
Hola mundo Yanina De Luna Ocampo desde el GPU (dispositivo)
Hola mundo Yanina De Luna Ocampo desde el GPU (dispositivo)
Hola mundo Yanina De Luna Ocampo desde el GPU (dispositivo)
Hola mundo Yanina De Luna Ocampo desde el GPU (dispositivo)
Hola mundo Yanina De Luna Ocampo desde el GPU (dispositivo)
```

Ahora lo vemos editando el código y el número de hilos finales, teniendo: <<<4, 1>>>

```
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/DeLunaY$ nano 01_hola.cu
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/DeLunaY$ nvcc 01_hola.cu -o 01_hola.x
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/DeLunaY$ ./01_hola.x
Hola mundo Yanina De Luna Ocampo desde el CPU (host)!
Hola mundo Yanina De Luna Ocampo desde el GPU (dispositivo)
Hola mundo Yanina De Luna Ocampo desde el GPU (dispositivo)
Hola mundo Yanina De Luna Ocampo desde el GPU (dispositivo)
Hola mundo Yanina De Luna Ocampo desde el GPU (dispositivo)
(base) bcruz@labdatos01:~/hpc_4AV1/estudiantes/DeLunaY$ █
```

### CONCLUSIONES

Un proceso paralelo es aquel que se realiza al mismo tiempo que otro, siendo ejecutados ambos de modo simultáneo. Cuando hablamos de procesos paralelos en un ordenador, nos referimos a aquellos procesos que se ejecutan y/o procesan a la vez, anteponiéndose a los procesos lineales o secuenciales, que serán ejecutados de uno en uno.

El proceso paralelo sólo es posible si disponemos de varias unidades de proceso, generalmente núcleos o procesadores. Esto se debe a que los procesadores son secuenciales, esto es, ejecutan las órdenes una por una, y por tanto no pueden ejecutar dos a la vez. Para solucionar esto se han inventado diversas fórmulas, como añadirles coprocesadores, derivar las operaciones a otros equipos conectados entre sí, disponer de múltiples núcleo, sin embargo hay que recalcar que la programación en paralelo necesita una gran arquitectura, por ejemplo el servidor al cual nosotros nos conectamos tenía un GPU Titan RTX, la cual cuesta alrededor de 500 mil pesos y no nada mas eso, sino que su procesador tambien tenia una gran capacidad, por lo que tenerlo conlleva un gran costo que tal vez podamos evitar haciendo otro tipo de arquitectura.

### BIBLIOGRAFÍA

1. *Servidores locales y remotos*. (2014). © Copyright IBM Corp. 2014. [https://www.ibm.com/docs/es/cognos-tm1/10.2.2?topic=SS9RXT\\_10.2.2/com.ibm.swg.ba.cognos.tm1\\_ug.10.2.2.doc/c\\_localandremotetm1servers\\_n90186.html](https://www.ibm.com/docs/es/cognos-tm1/10.2.2?topic=SS9RXT_10.2.2/com.ibm.swg.ba.cognos.tm1_ug.10.2.2.doc/c_localandremotetm1servers_n90186.html)
2. M. (2021, 14 septiembre). *Servidores remotos - SQL Server*. Microsoft Docs. <https://docs.microsoft.com/es-es/sql/database-engine/configure-windows/remote-servers?view=sql-server-ver15>

### CONSIDERACIONES FINALES

Descarga el documento antes de llenarlo.

Este documento se debe llenar en equipo, aunque la práctica la deben hacer TODOS los integrantes de este.

Después de llenar el documento, guárdalo como PDF y envíalo a través de la plataforma TEAMS, en la pestaña de tareas correspondiente. Solamente lo tiene que subir uno de los integrantes. Pero deben incluir TODOS los nombres de los integrantes del equipo en la primera página.

Queda estrictamente prohibido cualquier tipo de plagio a otros equipos o grupos. En caso de que ocurra, se anulará la práctica y se descontarán dos puntos a los equipos involucrados.