

# PROGRAMACIÓN MULTINÚCLEO



## Actividad 4 de programación



Entregar a más tardar: Lunes 10/Octubre, 5:30pm



Dr. Moisés Alencastre Miranda

malencastre@itesm.mx



# Actividad 4 de programación

- Programar de 2 formas lo siguiente (en C para el CPU y en CUDA en C con un número específico de bloques y con cierto número de threads por bloque):
  - Realizar el preprocesamiento de una imagen con al menos con una resolución de 20MP. Paralelizar correctamente la convolución en CUDA (encontrar la mejor opción de desempeño con bloques/threads 1D/2D/3D)
  - Programar un detector de orillas en base al cálculo de convolución (suma de Euler) usando alguno de los operadores/máscaras como: Sobel, Canny, Prewitt, Gaussiano Laplaciano, Kirsch, Differential Edge Detector, Roberts o algún otro (cada equipo uno diferente) que utilice una matriz de  $N \times N$  donde  $N \geq 3$ .
  - Integrar el cargador de imagen según sea PNG, JPG, GIF, etc sea con OpenGL o OpenCV como se vió en los ejemplos.
  - Desplegar, la imagen inicial y la imagen con las orillas detectadas.
  - Incluir también en cada caso como parámetros del ejecutable:

# Actividad 4 de programación

- Número de bloques de CUDA a considerar.
- Número de threads por cada bloque de CUDA.
- No olviden tomar el tiempo de ejecución de las operaciones en cada uno de los programas para poder compararlos. Para esto, además de entregar los 2 códigos fuente, incluir un archivo de texto que incluya 2 tiempos de ejecución: medir el tiempo total de todo el programa en cada uno de los casos, además de para ciertos valores de bloques/threads. Incluir las características de la computadora donde se hizo (velocidad, mem y # de cores del CPU, velocidad, mem y # de cores del GPU).
- No olviden documentar, indentar, optimizar y automatizar el código. No olviden incluir nombre y matrícula.