Assignment 1: Linear Transformations

Objective

The first assignment will help you understand the following basics

1. The mapping between multi-dimensional kernels and a linear array in memory
2. Using CUDA math functions

Problem Statement:

Given one-dimensional arrays you have to implement three kernel routines:

- **process_kernel1(float *input1, float *input2 float *output, int datasize)**:
  This function takes as arguments two 1-D arrays named **input1** and **input2** of total size **datasize** , processes them, and writes to an array named output. Your objective would be to compute for each data point of the two arrays the following operation

  output[i]=sin(input1[i])+cos(input2[i])

  You will have to use a 3 Dimensional Grid of dimensions <<<4,2,2>>> with 2-Dimensional blocks with <32,32,1>. Accordingly, ascertain the number of elements of the input arrays and output array.

- **process_kernel2(float *input, float *output, int datasize)**:
  This function takes as argument one 1-D array named input of total size datasize , processes, and writes to an array named output. Your input array to this function will be the output array obtained as a result of process_kernel1. The following operation needs to be performed here

  output[i]=log(input[i]))

  You will have to use a 2 Dimensional Grid of dimensions <<<2,?,1>>> with 3-Dimensional blocks with <8,8,?>. Accordingly, fill in the blanks.

- **process_kernel3(float *input, float *output, int datasize)**:
  This function takes as argument one 1-D array named input of total size datasize , processes, and writes to an array named output. Your input array to this function will be the output array obtained as a result of process_kernel2. The following operation needs to be performed here

  output[i]=sqrt(input[i]))

  You will have to use a 1 Dimensional Grid of dimensions <<< ?,1,1 >>> with 2-Dimensional blocks with <128,?,1>. Accordingly, fill in the blanks.