**Extra Lab-2 – CUDA lab exercise on warps**

* Launches a CUDA kernel.
* Computes and prints:
  + **Global Thread ID**
  + **Warp ID**
  + **Lane ID (thread's index within the warp)**
* Does basic addition between two arrays.

**CUDA Program**

#include <iostream>

#include <cuda\_runtime.h>

#define WARP\_SIZE 32

\_\_global\_\_ void simpleWarpInfoKernel(float \*A, float \*B, float \*C, int N) {

int globalThreadId = blockIdx.x \* blockDim.x + threadIdx.x;

if (globalThreadId < N) {

float a = A[globalThreadId];

float b = B[globalThreadId];

float result = a + b;

C[globalThreadId] = result;

int warpId = globalThreadId / WARP\_SIZE;

int laneId = threadIdx.x % WARP\_SIZE;

// Print thread and warp info

printf("GlobalTID=%d, WarpID=%d, LaneID=%d, A=%.1f, B=%.1f, Sum=%.1f\n",

globalThreadId, warpId, laneId, a, b, result);

}

}

int main() {

const int N = 64;

size\_t size = N \* sizeof(float);

float \*h\_A = new float[N];

float \*h\_B = new float[N];

float \*h\_C = new float[N];

for (int i = 0; i < N; i++) {

h\_A[i] = static\_cast<float>(i);

h\_B[i] = static\_cast<float>(i \* 2);

}

float \*d\_A, \*d\_B, \*d\_C;

cudaMalloc(&d\_A, size);

cudaMalloc(&d\_B, size);

cudaMalloc(&d\_C, size);

cudaMemcpy(d\_A, h\_A, size, cudaMemcpyHostToDevice);

cudaMemcpy(d\_B, h\_B, size, cudaMemcpyHostToDevice);

int threadsPerBlock = 32;

int blocksPerGrid = (N + threadsPerBlock - 1) / threadsPerBlock;

simpleWarpInfoKernel<<<blocksPerGrid, threadsPerBlock>>>(d\_A, d\_B, d\_C, N);

cudaDeviceSynchronize();

cudaMemcpy(h\_C, d\_C, size, cudaMemcpyDeviceToHost);

// Optional: print results summary

std::cout << "\n--- Summary ---\n";

for (int i = 0; i < N; i += WARP\_SIZE) {

std::cout << "Warp " << i / WARP\_SIZE << ": Sample Output Sum = " << h\_C[i] << std::endl;

}

delete[] h\_A;

delete[] h\_B;

delete[] h\_C;

cudaFree(d\_A);

cudaFree(d\_B);

cudaFree(d\_C);

return 0;

}

**Compilation (using nvcc)**

nvcc -o warp\_info warp\_info.cu

./warp\_info

**Sample Output**

GlobalTID=0, WarpID=0, LaneID=0, A=0.0, B=0.0, Sum=0.0

GlobalTID=1, WarpID=0, LaneID=1, A=1.0, B=2.0, Sum=3.0

...

GlobalTID=32, WarpID=1, LaneID=0, A=32.0, B=64.0, Sum=96.0

...