REPORT
LAB3-HISTOGRAMS

Nithin Krishnan(NKE1490)
Matthew Lenk(MLG7376)

We were not able to get the correct result for about 60 of the 1000 iterations. We believe it is due to the different method of accessing the input array as it is a Multi-level array and not a 2D array. We tried to tackle this by padding INPUT_WIDTH to a multiple of 128, but it did not make any change.

Optimizations to the Program:
Initially, we tried to implement a naive implementation of the histogram, without any optimization techniques, this was slower than the CPU implementation.

Then, we implemented a an algorithm using tiling, This gave us about 2x better performance than the CPU implementation. We improved on this by changing the block size and thread size during kernel call to try and use the resources of the GPU better.

Finally, we made use of the atomicAdd operation and #pragma unroll to further reduce the execution time to 0.761 seconds(14x better than CPU).

Future optimizations we looked to make was have shared local memory constantly incremented instead of global to reduce the latency to access global memory. But we could not move forward with only a few iterations not being the same.

We have also printed the values of gold_bins and kernel bins just to show that only about 60 of the iterations differ in value.