Lab 2: Heterogeneous Acceleration

Goals

Predict the expected performance improvement for offloading the color conversion portion of the serial application to a GPU and analyze the actual results compared to your predictions.

Part I: Estimating the impact

1. Read up online about the GPU in the lab machines so you understand their capabilities, speeds, and design.
2. Analyze the RGB to YCbCr code to determine what you need to do to offload this calculation to the GPU, and estimate the performance impact of making that change. You should address the following:
    a. GPU initialization
    b. Data movement (bandwidth)
    c. Kernel execution (divergence, parallelism, coalescing, bandwidth)

Predicted Performance Impact

Part	Predicted Performance Impact	Why? (How did you come up with this impact?)
Initialization		
Data Movement		
Kernel Execution		
Total Impact (speedup)		

Part 2: Measuring the impact

Implement each of the parts above and measure the net performance improvement and individual performance impact. Use the GPU profiler to analyze your application performance. (You can even include screenshots from this in your report if it helps clarify what is happening.)

Please make sure you've completed part 1 before you do this part.

Actual Performance Impact

Part	Predicted Performance Impact	Actual Performance Impact
Initialization		
Data Movement		
Kernel Execution		
Total Impact (speedup)		

In addition to the above, you should investigate the parallel scaling by looking at the speedup as a function of the number of GPU cores used. To do this change the global dimensions of your kernel to only have as many work-items/threads as you want to execute in parallel. This will require changing how much work each work-item does so it still does the complete calculation. Also keep in mind how many actual cores there are on the device when you interpret your results! Note that you will have to have a large enough image for this to make sense.

Actual Parallel Scaling

Cores	16	32	64	128	256	512	1024	(Image Size)
Execution Time								
Speedup								1.0

Analysis

You have now tried to predict the impact of offloading the color conversion to the GPU as well as measured the results. The point of this lab is twofold: 1) analyze code to predict the performance benefit of an optimization and 2) get better performance by optimizing. Your lab report should address these two issues.

Note: The most important thing is to show that you understand what is going on. If your estimates were far off the actual performance then you should analyze the real results and use the profiler to figure out what is really happening. (As well as talk to the TA.) Your report should explain what is really going on and what you missed when you did your first estimates. You will not lose any points for having plausible, but incorrect, initial estimates if you plausibly identify what was wrong about them from the actual implementation. (Indeed, that's the whole point!)

You should produce a 2-page, concise report with the following 4 sections:

- **1. Performance Estimates**: Your estimates of the performance impact of the GPU offload and how you came up with them.
- **2. Performance Achieved**:
  - Your measured performance results from implementing the optimizations.
  - The parallel scaling of your implementation.
- **3. Discussion**:
  - A discussion of the measured results and how they differ from your predictions.
  - A discussion of what you learned about these optimizations from implementing them and measuring the results.
  - Comment on any unexpected or odd results.
  - Comments on the difficulty of the GPU offload.
- **4. Lab comments**: Any feedback on the lab itself.

The comparative results should be presented in a graphical form to make it easy to see trends and analysis.