Exam Assignments 6

Instruction Set Characteristics

SSE

- 128-Bit vector length
- Uses Registers `xmm0`-`xmm15`
- Latest release 2009

AVX(2)

- 256-Bit vector length
- Uses Registers `ymm0`-`ymm15`
- Latest release 2013

AVX-512

- 512-Bit vector length
- Uses Registers `zmm0`-`zmm31`
- Latest release 2017
- Currently only used by high-performance CPUs
- Completely backwards-compatible with AVX(2) and SSE

Memory Aliasing

Memory aliasing is a situation in which a memory location can be accessed in multiple ways, for example by two different pointers. If the compiler can't be sure that no aliasing is happening, it can only perform optimizations that are safe in this situation, thus losing out on some performance.

The programmer can help the compiler by making sure no memory aliasing is happening and assuring the compiler by using the `restrict` keyword.

Advantages of Stride-1 Memory Access

The smallest amount of memory we can access individually is a cache line. Thus, if we access data that is spread out over memory, we will need to load more cache lines (wasting

memory bandwidth along the way) than if we could simply access data next to eachother.

Also, vectorized instructions are much faster if the data don't need to be gathered up from multiple places. This leads to the compiler possibly not automatically vectorizing the code if we do not access memory with stride 1.

When to use Structure of Arrays

Using a structure of arrays, the member elements of all struct instances are next to each other in memory. This boosts the performance of operations that work with one member of many struct instances.

If we are going to be performing many of these types of operations, a structure of arrays would be better than an array of structs. Those are better for operations on multiple members of a single struct instance.