Design Document

MMWT

**Purpose:**

Computing a product of two matrices using threads concurrently

**Constraints:**

* Matrices are at least 3 by 3
* Threading, not sequential computation
* Matrices does not have to be square, nor same dimensions
* Allocated memory for matrices will fit within the finite machine

**Design:**

To improve efficiency, threads should be used (Under the assumption, there is more than one CPU). Since threads share common data, my design would be in a *m* x *n* x *p* multiplication, there should be *m* x *n* threads, each doing *p* multiplications (that is to say, the ideal scenario is *m* x *n* CPUs so that all threads can execute simultaneous).

Following Gunnet et al. [2001]’s layered approach to matrix multiplication, the algorithm of my computation will revolve around C:= AB + C.

**Improvements:**

While overall, it worked as I had hoped – the speed isn’t as fast as ideal because the issues from the Transition Lookaside Buffer

Using *thrashing* to reduce TLB misses and make more cache friendly.