

CS 484 : Parallel Programming -- Machine Problem 0

Nathan Nard

University of Illinois Urbana-Champaign
20190127

1. Benchmarking Part 1

Time per element access increases only slightly as array size increases up to 4194304. These small increases are likely attributed to longer arrays requiring more time to find elements in the contiguous data. But we see a doubling in access time when the array size goes up from 4194304 to 8388608, a much larger increase compared to previous increases. This is due to the CPU's cache size of 30720 KB. The size of the two arrays are $4194304 * 8 / 1000 = 33554$ KB and $8388608 * 8 / 1000 = 67109$ KB. While both will require flushing the cache a number of times as array elements are accessed in sequential order, the latter array size will need to do this a greater number of times as its data size is more than twice that of the CPU cache size. Element access time will double again for a size of 16777216, but then the last few seem to plateau/taper off at about 1.5 seconds per access each. Likely because the array size has become close to the number of element accesses (referred to as "repetitions" in the cpp file), which is about 100 million, which means they will each have flushed the cache a similar number of times causing a negligible difference in element access times.

2. Benchmarking Part 2

	Optimization		Performance Increase?
	0	3	
test1	0.0019593238830566	0.0000000000000000	Yes
test2	0.0028539419174194	0.0008825778961182	Yes
	0.0022427797317505	0.0009930133819580	Yes
test3	0.0236785411834717	0.0016731023788452	Yes
	0.0032806873321533	0.0004936933517456	Yes
test4	0.0232986450195313	0.0054608821868896	Yes
test5	6.0628969669342041	1.1163949966430664	Yes
	8.0769040584564209	2.1645071506500244	Yes
	6.0959568023681641	1.1310789585113525	Yes