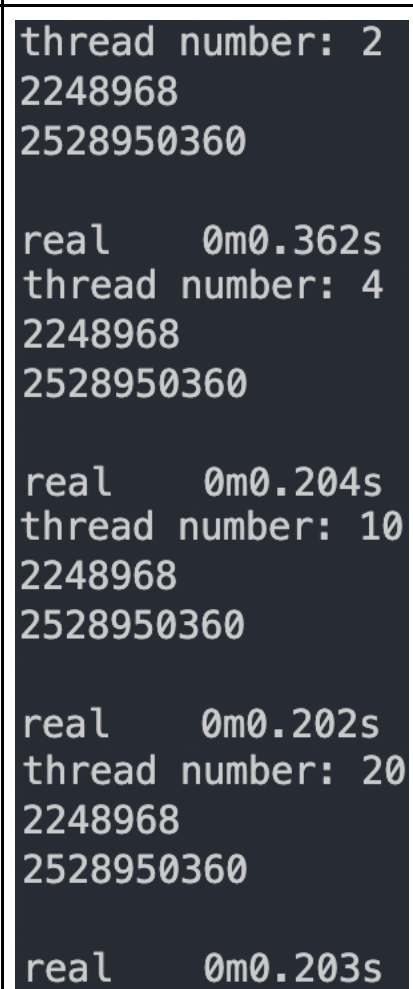


2022 NYCU OS HW2 report

陳存佩 109550171

Question	Answer
<p>Q1. (5pts)</p> <p>Briefly describe your design for the add, multiple function of matrix, the thread management.</p> <p>Also, describe the number of threads in the Multi-thread program.</p>	<p>透過迴圈建立 thread，大家都跑一樣的 add & multiple function，function 內會呼叫 global variable row，用來判斷做到哪一行（每個 thread 沒有預先設定要做哪行，看當時 row 而定），row 相關的變動跟判斷被包在 critical section 內。</p> <p>經過測試，在 thread = 10 達到最好的 speedup，因為工作站是 4 cores，基本上超過 4 都會有相近的結果。</p>
<p>Q2. (15pts)</p> <p>Try at least 3 kinds of number of threads, and compare the difference in time.(Take screenshots of the time of each case)</p> <p>Also, explain the results.</p>	 <pre>thread number: 2 2248968 2528950360 real 0m0.362s thread number: 4 2248968 2528950360 real 0m0.204s thread number: 10 2248968 2528950360 real 0m0.202s thread number: 20 2248968 2528950360 real 0m0.203s</pre>

	<p>在 thread number=2~4 時有明顯的進步，超過 4 後因為受限工作站本身硬體設備的限制，都維持在相近的數值上。</p> <p>硬體規格</p> <ul style="list-style-type: none"> • 硬體 <ul style="list-style-type: none"> ◦ 4 Cores Virtualized Intel(R) Xeon(R) Gold 6126 CPU ◦ 16 GB Memory
<p>Q3. (10pts)</p> <p>Show the best speedup between multi-thread and single-thread. (Take screenshots of the time of single-thread and multi-thread)</p> <p>Also, explain why multi-thread is faster.</p>	<pre>bash-4.4\$ time ./single < input.txt 2248968 2528950360 real 0m0.731s thread number: 10 2248968 2528950360 real 0m0.202s</pre> <p>Speedup = $0.731/0.202 = 3.6$</p> <p>因為 multi-thread 可以平行同時處理多 row 的加跟乘，所以速度會比較快。</p>