An-Najah National University
Faculty of Engineering and IT

جامعة النجاح الوطنية
كلية الهندسة وتكنولوجيا المعلومات

Name: Tariq Sabri

Arch 2 Hw 3,Memory

The tables here show time values from running a matrix multiplication function that is not optimized and compare it to functions with different memory optimization approaches. Here we have Interchange, Blocked and Blocked Interchange.

Interchange

Input Size	Unoptimized	Optimized	Improvement%
512	0.566365	0.441379	1.283172
1024	5.859431	3.496998	1.675560
2048	57.324988	27.388802	2.093008

Blocked

Input Size	Unoptimized	Optimized	Improvement%
512/16	5.966988	4.009286	1.488292
512/32	5.966988	4.395856	1.357412
1024/16	5.781582	3.940287	1.467300
1024/32	5.781582	4.344669	1.330730
1024/64	5.781582	3.900649	1.482210

Blocked Interchanged

Input Size	Unoptimized	Optimized	Improvement%
512/16	0.556756	0.450598	1.235594
512/32	0.556756	0.523895	1.062725
1024/16	5.783555	3.618765	1.598213
1024/32	5.783555	3.508823	1.648289
1024/64	5.783555	3.967878	1.457594

What can be observed:

- **On the interchange optimization we can see that the improvement increases as the input size increases, this can be caused due to the fact that increased sizes increase the differance in time which results in a better improvement while that's not the case with blocked and blocked interchange.**
- **the time for Blocked increases as the block size increases.**

It's worth noting that I've tried first to run a function where the 3 loops were interchanged which resulted with doubling the time for the conventional multiplication instead of decreasing the time...this and some mistakes with the other optimization functions.. you can find the outputs in a file called output_wrong.txt. Can't put it here since I want to keep the report one page long.

Issue number: AD3-3