Lab2 computer architecture

Cache blocking and Matrix Multiplication

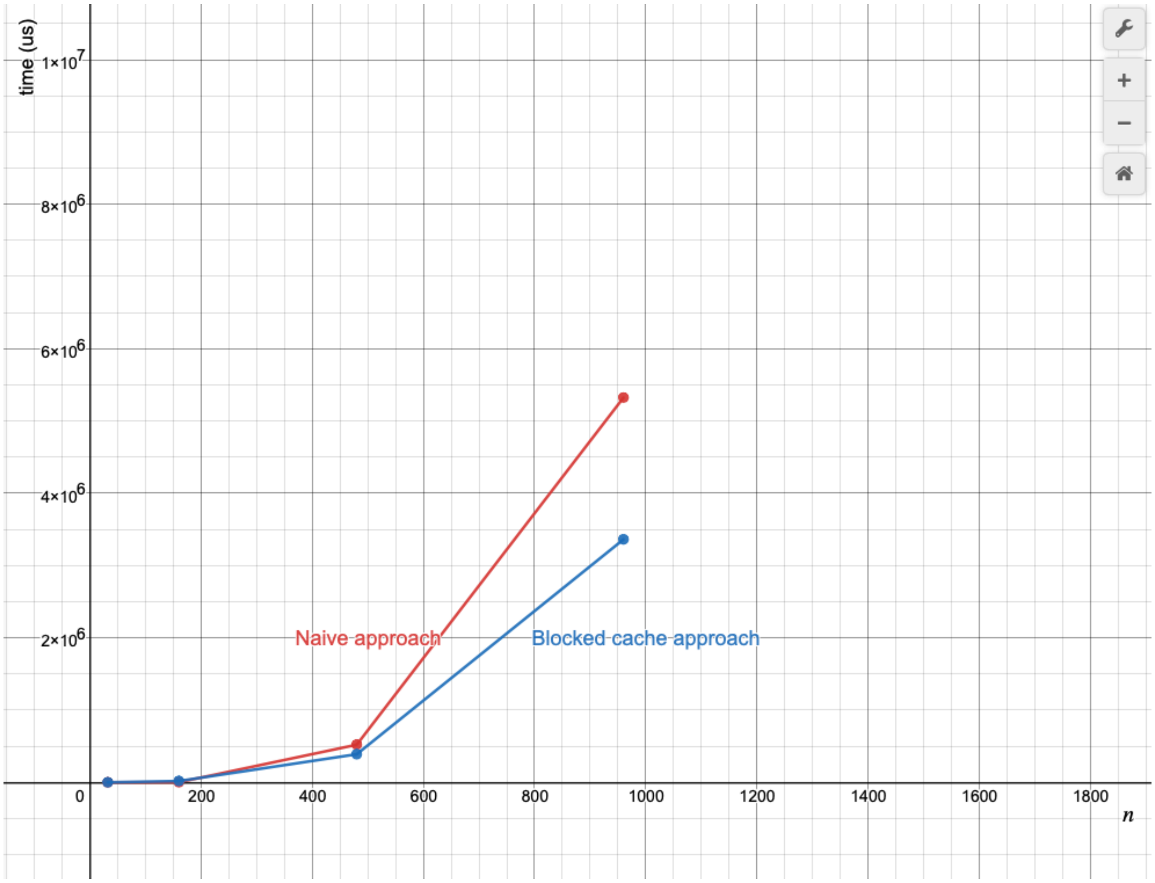
Name: **Salma Ragab Gad**

ID: **32**

We will run the program for sizes 32x32, 160x160, 480x480 and 960x960.

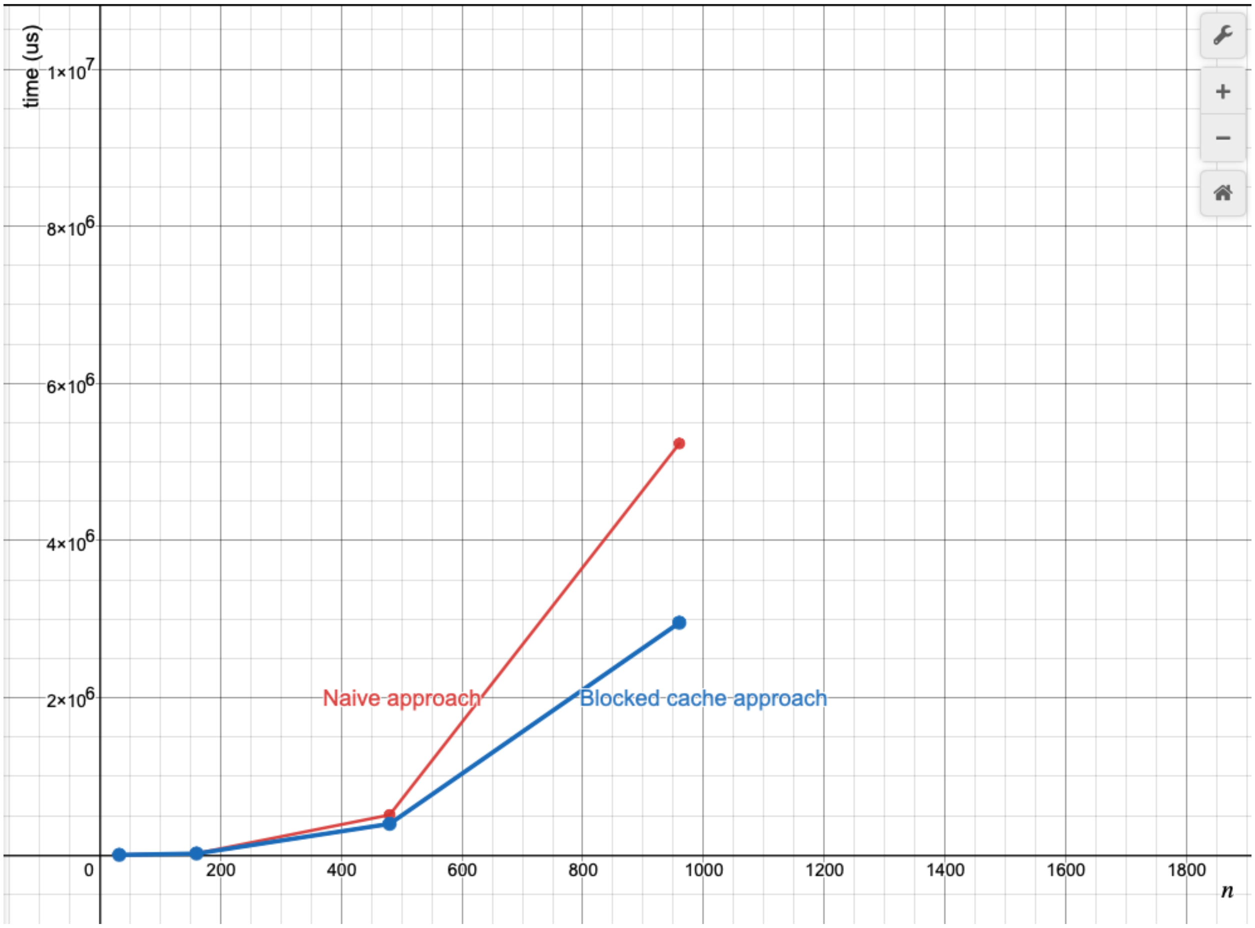
For block size = 16:

Matrices sizes	Time for naïve approach	Time for blocked cache approach
32x32	117 us	115 us
160x160	19081 us	18323 us
480x480	520620 us	389163 us
960x960	5323738 us	3361515 us



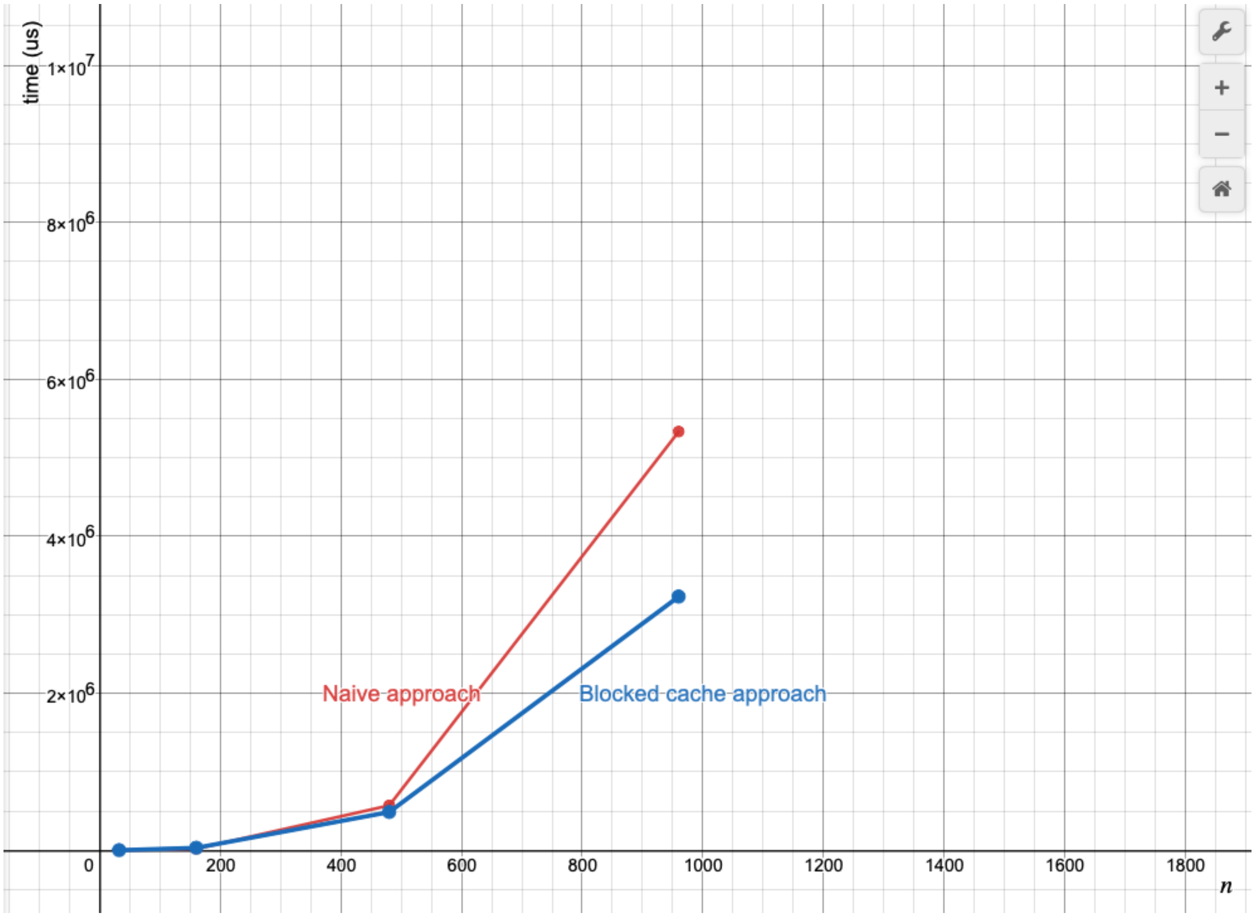
For block size = 32:

Matrices sizes	Time for naïve approach	Time for blocked cache approach
32x32	117 us	112 us
160x160	18737 us	17088 us
480x480	509383 us	394937 us
960x960	5233954 us	2953943 us



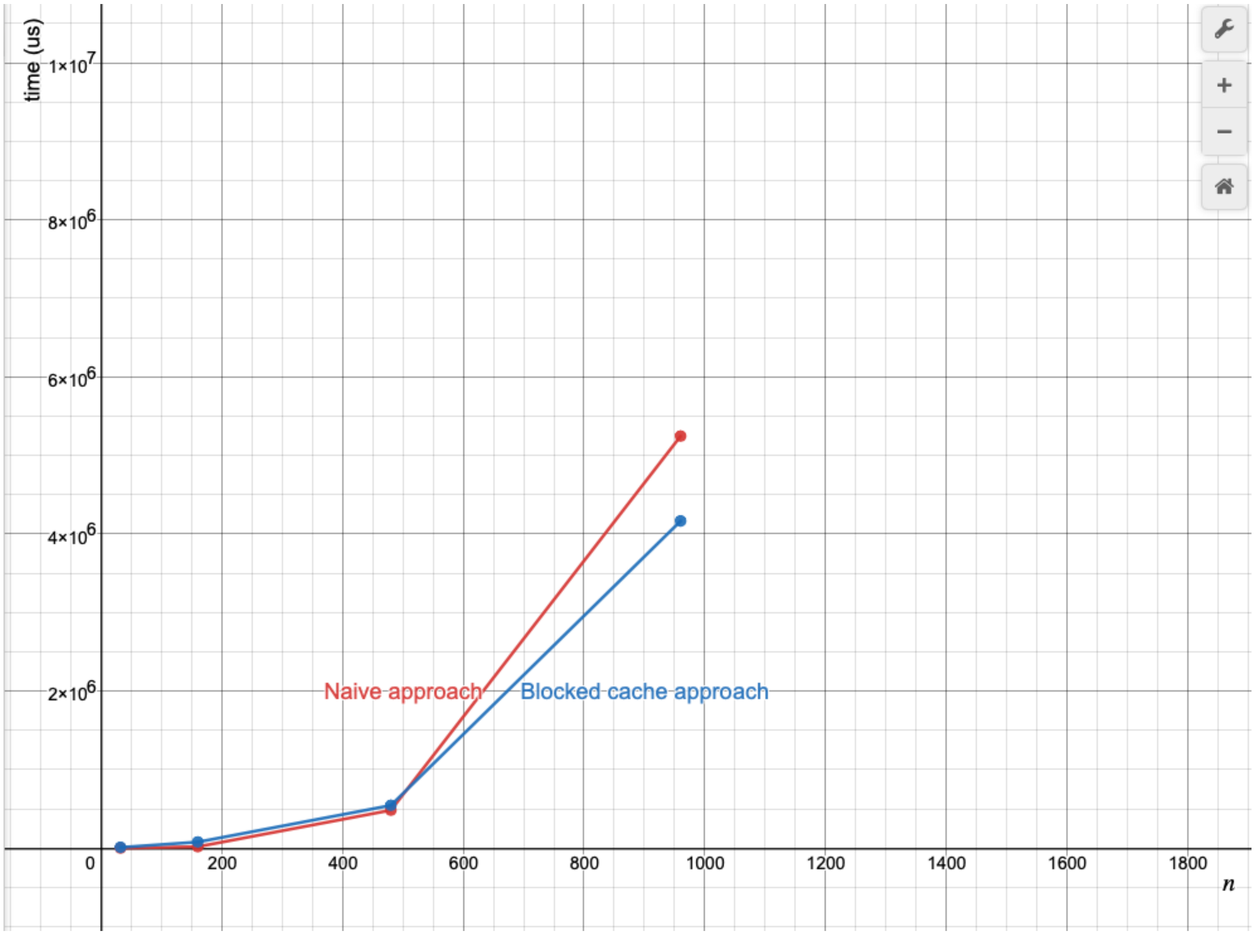
For block size = 64:

Matrices sizes	Time for naïve approach	Time for blocked cache approach
32x32	118 us	952 us
160x160	19330 us	31211 us
480x480	570865 us	486973 us
960x960	5332245 us	3230532 us



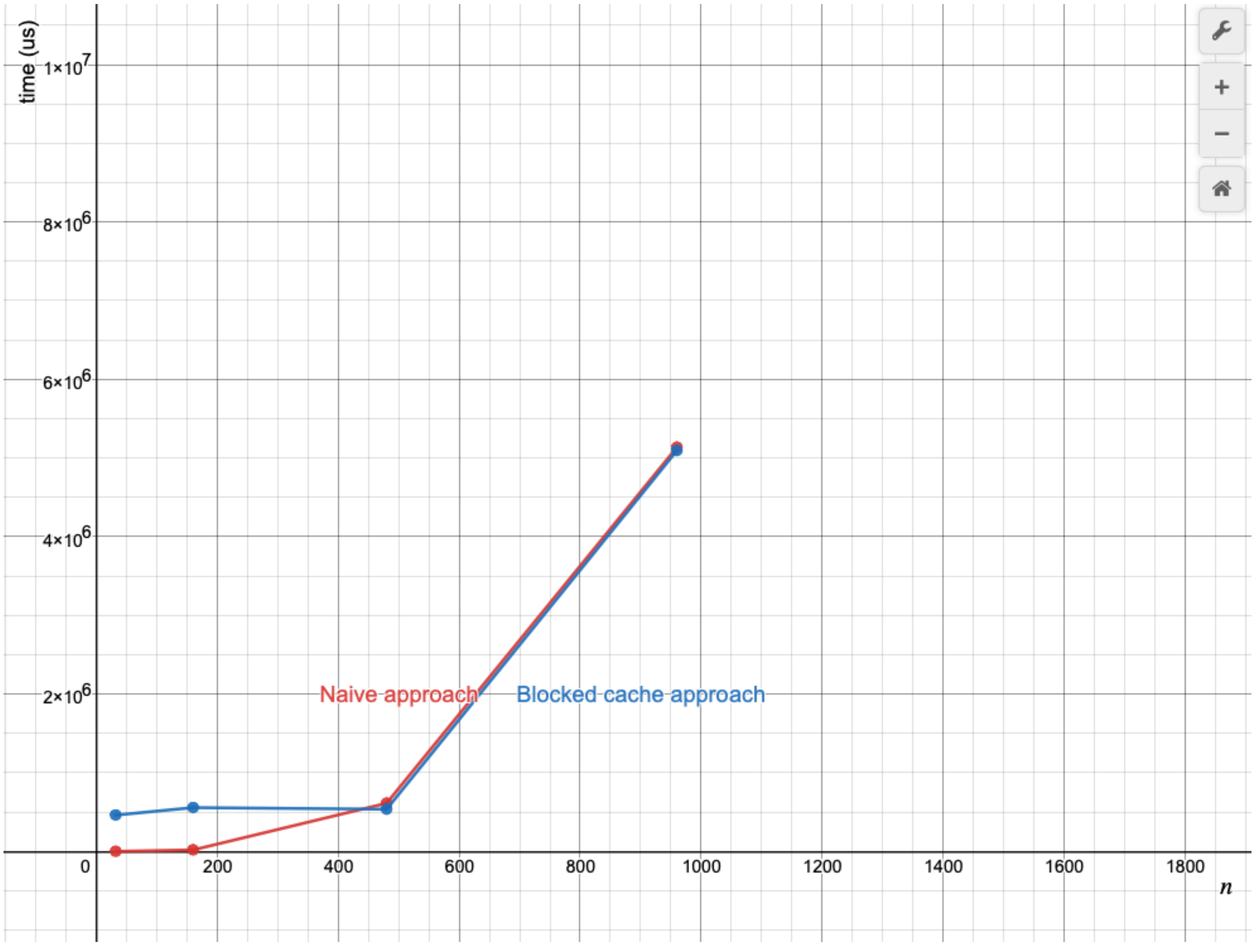
For block size = 128:

Matrices sizes	Time for naïve approach	Time for blocked cache approach
32x32	153 us	9269 us
160x160	17988 us	77609 us
480x480	482534 us	544324 us
960x960	5245495 us	4164211 us



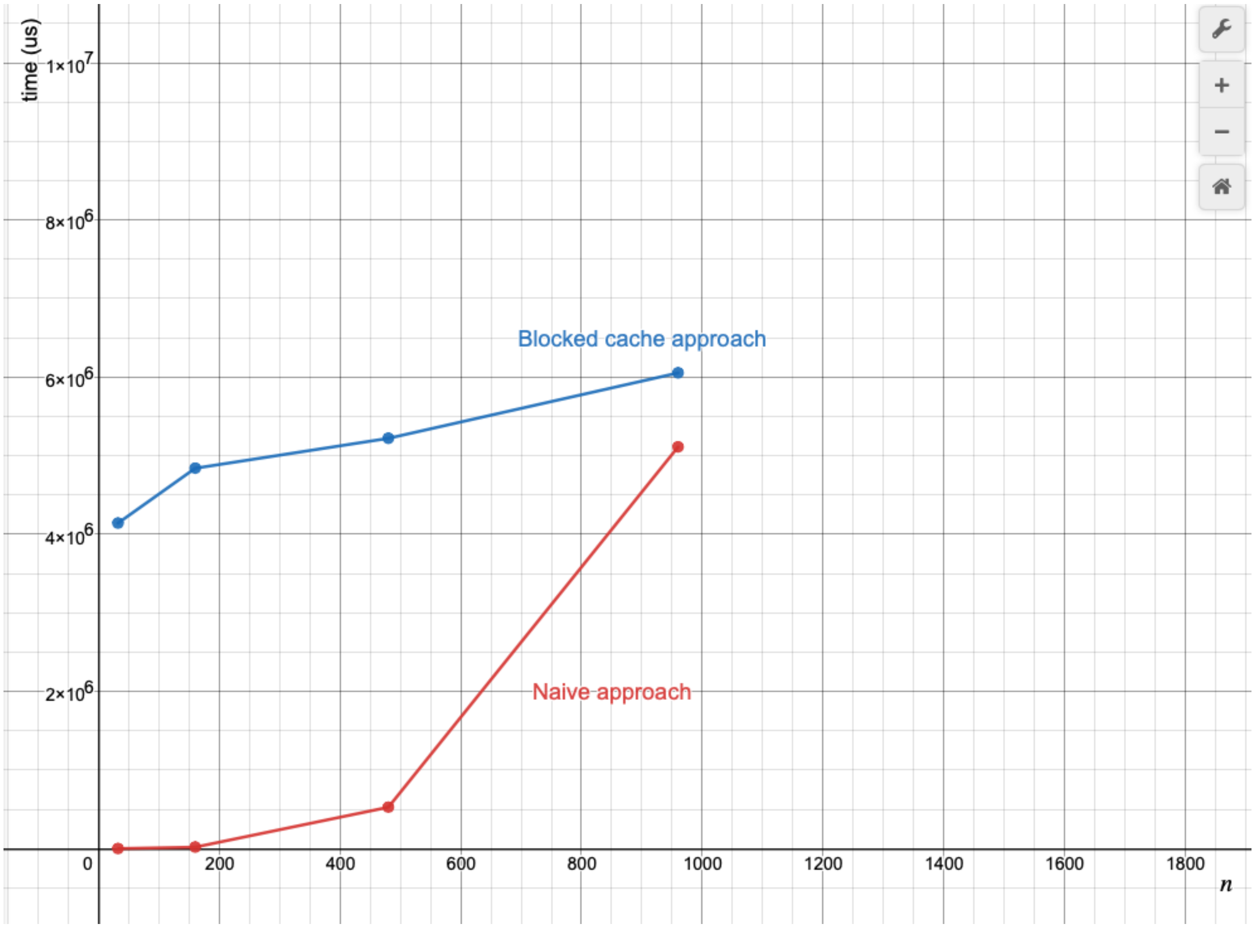
For block size = 500:

Matrices sizes	Time for naïve approach	Time for blocked cache approach
32x32	117 us	461441 us
160x160	18784 us	556594 us
480x480	612335 us	536689 us
960x960	5136053 us	5094925 us



For block size = 1000:

Matrices sizes	Time for naïve approach	Time for blocked cache approach
32x32	161 us	4141681 us
160x160	19054 us	4840426 us
480x480	526905 us	5220921 us
960x960	5111100 us	6053408 us



Conclusion:

- The blocked cache approach takes less time than the naïve approach but when the block size is too large, the naïve approach takes less time (like in block size = 1000).
- We should choose a block size close to the cache size to reduce cache misses and take advantages of the algorithm.

Code Link:

https://drive.google.com/drive/folders/1SVP4PAZwnkMHBgYv_hBH_VP_rEE2QfLS?usp=sharing