

Github link:

<https://github.com/shubh-ranade/nyu-hpc-homework-2>

Please find my comments and explanations for questions 1 and 3 in respective cpp files.

I ran my code on Intel Core i7-6500U processor with base frequency of 2.5GHz and max memory bandwidth of 34.1 GB/s

Q4 Jacobi/Gauss-Seidel:

Jacobi -

Number of threads	gridsize	max_iter	time in seconds
-------------------	----------	----------	-----------------

-----	-----	-----	-----
-------	-------	-------	-------

1 (serial)	100	100	0.002860
------------	-----	-----	----------

2	100	100	0.002523
---	-----	-----	----------

4	100	100	0.002058
---	-----	-----	----------

8	100	100	0.077466
---	-----	-----	----------

1 (serial)	100	1000	0.041616
------------	-----	------	----------

2	100	1000	0.104065
---	-----	------	----------

4	100	1000	0.029419
---	-----	------	----------

8	100	1000	0.866219
---	-----	------	----------

1 (serial)	100000	100	45.86752
------------	--------	-----	----------

2	100000	100	32.17725
---	--------	-----	----------

4	100000	100	32.05655
---	--------	-----	----------

8	100000	100	32.05655
---	--------	-----	----------

1 (serial)	100000	1000	415.3316
------------	--------	------	----------

2	100000	1000	340.8675
---	--------	------	----------

4	100000	1000	311.0212
---	--------	------	----------

Gauss-Seidel -

| Number of threads | gridsize | max_iter | time in seconds |

|-----|-----|-----|-----|

| 1 (serial) | 100 | 100 | 0.006701 |

| 2 | 100 | 100 | 0.003898 |

| 4 | 100 | 100 | 0.015327 |

| 8 | 100 | 100 | 0.147629 |

| 1 (serial) | 100 | 1000 | 0.039415 |

| 2 | 100 | 1000 | 0.033383 |

| 4 | 100 | 1000 | 1.375098 |

| 8 | 100 | 1000 | 1.191786 |

| 1 (serial) | 100000 | 100 | 53.76188 |

| 2 | 100000 | 100 | 40.45902 |

| 4 | 100000 | 100 | 41.60815 |

| 8 | 100000 | 100 | 46.96951 |

| 1 (serial) | 100000 | 1000 | 549.3978 |

| 2 | 100000 | 1000 | 0.104065 |

| 4 | 100000 | 1000 | 413.1400 |

| 8 | 100000 | 1000 | 419.3472 |

Q2:

Given loop ordering in MMult0 is the most efficient for column major matrices.

I found that without omp, block size of 64 works best.

Results for block size:

With omp, the algorithm run slower.

Serial execution with efficient loop ordering:

Baseline MMult0 timing

Dimension	Time	Gflop/s	GB/s	Error
64	0.225779	8.858907	141.742513	0.000000e+00
128	0.210227	9.516779	152.268469	0.000000e+00
192	0.205692	9.772472	156.359545	0.000000e+00
256	0.204245	9.857137	157.714184	0.000000e+00
320	0.245430	8.277792	132.444672	0.000000e+00
384	0.253860	8.029748	128.475963	0.000000e+00
448	0.250672	8.608727	137.739633	0.000000e+00
512	0.252803	8.494682	135.914912	0.000000e+00
576	0.283901	8.077587	129.241385	0.000000e+00
640	0.333009	6.297580	100.761276	0.000000e+00
704	0.406337	5.152086	82.433370	0.000000e+00
768	0.665379	4.084754	65.356071	0.000000e+00
832	0.520665	4.424573	70.793163	0.000000e+00
896	0.686855	4.189082	67.025317	0.000000e+00
960	0.840201	4.212021	67.392339	0.000000e+00
1024	0.510102	4.209912	67.358591	0.000000e+00
1088	0.653243	3.943136	63.090170	0.000000e+00

1152	0.848562	3.603329	57.653262	0.000000e+00
1216	0.973034	3.695753	59.132041	0.000000e+00

Block sizes of 16 and 64 produce comparable results!!

I report both timings.

With blocking and block size 16

Dimension	Time	Gflop/s	GB/s	Error
16	0.243286	8.220800	131.532799	0.000000e+00
64	0.255968	7.814100	125.025597	0.000000e+00
112	0.439131	4.555854	72.893658	0.000000e+00
160	0.231969	8.652183	138.434930	0.000000e+00
208	0.389270	5.178296	82.852735	0.000000e+00
256	0.271166	7.424486	118.791775	0.000000e+00
304	0.236340	8.558880	136.942079	0.000000e+00
352	0.278564	7.202128	115.234047	0.000000e+00
400	0.276483	7.407330	118.517275	0.000000e+00
448	0.240918	8.957266	143.316263	0.000000e+00
496	0.306421	7.168019	114.688305	0.000000e+00
544	0.281662	8.001962	128.031389	0.000000e+00
592	0.248620	8.345059	133.520943	0.000000e+00
640	0.252082	8.319315	133.109036	0.000000e+00
688	0.346087	7.527829	120.445269	0.000000e+00
736	0.299270	7.993218	127.891487	0.000000e+00
784	0.356875	8.101827	129.629227	0.000000e+00
832	0.264579	8.707115	139.313837	0.000000e+00
880	0.339383	8.031888	128.510215	0.000000e+00

928	0.438822	7.284770	116.556315	0.000000e+00
976	0.508645	7.311297	116.980747	0.000000e+00
1024	0.234628	9.152701	146.443220	0.000000e+00
1072	0.291664	8.447556	135.160895	0.000000e+00
1120	0.335311	8.379843	134.077486	0.000000e+00
1168	0.381281	8.358220	133.731525	0.000000e+00
1216	0.423237	8.496640	135.946233	0.000000e+00

With blocking and block size 64

Dimension	Time	Gflop/s	GB/s	Error
64	0.210975	9.480562	151.688992	0.000000e+00
128	0.245398	8.152819	130.445102	0.000000e+00
192	0.226100	8.890423	142.246768	0.000000e+00
256	0.187727	10.724423	171.590770	0.000000e+00
320	0.215161	9.442310	151.076966	0.000000e+00
384	0.238026	8.563918	137.022690	0.000000e+00
448	0.228248	9.454490	151.271842	0.000000e+00
512	0.237028	9.060027	144.960428	0.000000e+00
576	0.253779	9.036364	144.581818	0.000000e+00
640	0.241155	8.696275	139.140404	0.000000e+00
704	0.256137	8.173300	130.772793	0.000000e+00
768	0.511364	5.315019	85.040309	0.000000e+00
832	0.286312	8.046204	128.739260	0.000000e+00
896	0.383271	7.507193	120.115095	0.000000e+00
960	0.453617	7.801605	124.825688	0.000000e+00
1024	0.244063	8.798909	140.782539	0.000000e+00
1088	0.370166	6.958574	111.337190	0.000000e+00

1152	0.363345	8.415269	134.644305	0.000000e+00
1216	0.519839	6.917705	110.683278	0.000000e+00
1280	0.523786	8.007670	128.122725	0.000000e+00

Timing with 4 threads:

With block size 16:

Dimension	Time	Gflop/s	GB/s	Error
16	2.563034	0.780326	12.485222	0.000000e+00
64	0.523985	3.817204	61.075262	0.000000e+00
112	2.114940	0.945945	15.135128	0.000000e+00
160	1.473726	1.361882	21.790105	0.000000e+00
208	1.515147	1.330403	21.286455	0.000000e+00
256	0.949600	2.120121	33.921937	0.000000e+00
304	0.530435	3.813474	61.015578	0.000000e+00
352	0.601020	3.338080	53.409273	0.000000e+00
400	0.538651	3.802088	60.833407	0.000000e+00
448	0.721312	2.991729	47.867670	0.000000e+00
496	0.575486	3.816655	61.066483	0.000000e+00
544	0.663614	3.396322	54.341153	0.000000e+00
592	0.574649	3.610459	57.767350	0.000000e+00
640	0.961972	2.180056	34.880889	0.000000e+00
688	1.144700	2.275954	36.415269	0.000000e+00
736	0.923805	2.589431	41.430896	0.000000e+00
784	1.474635	1.960717	31.371471	0.000000e+00
832	1.203600	1.914026	30.624413	0.000000e+00
880	2.378911	1.145856	18.333689	0.000000e+00

928	2.706830	1.180981	18.895696	0.000000e+00
976	2.828737	1.314670	21.034727	0.000000e+00
1024	5.353856	0.401110	6.417755	0.000000e+00
1072	2.359332	1.044300	16.708803	0.000000e+00
1120	3.741904	0.750916	12.014659	0.000000e+00
1168	3.292929	0.967779	15.484465	0.000000e+00
1216	4.441931	0.809578	12.953253	0.000000e+00

With block size 64:

Dimension	Time	Gflop/s	GB/s	Error
64	2.753878	0.726306	11.620901	0.000000e+00
128	1.220267	1.639546	26.232731	0.000000e+00
192	0.926488	2.169614	34.713819	0.000000e+00
256	1.308074	1.539107	24.625709	0.000000e+00
320	0.786438	2.583312	41.332997	0.000000e+00
384	0.977456	2.085446	33.367137	0.000000e+00
448	0.814002	2.651061	42.416969	0.000000e+00
512	1.114835	1.926280	30.820484	0.000000e+00
576	0.858575	2.670978	42.735648	0.000000e+00
640	1.346402	1.557597	24.921559	0.000000e+00
704	1.209308	1.731141	27.698252	0.000000e+00
768	1.653604	1.643628	26.298048	0.000000e+00
832	1.258940	1.829889	29.278228	0.000000e+00
896	1.722556	1.670362	26.725799	0.000000e+00
960	2.296716	1.540871	24.653941	0.000000e+00
1024	6.453081	0.332784	5.324548	0.000000e+00
1088	2.923975	0.880933	14.094933	0.000000e+00

1152 4.906581 0.623173 9.970764 0.000000e+00

1216 4.552474 0.789920 12.638724 0.000000e+00