

Lab 1

4.

	1	2	4	8	16	32	64
2	0.814	0.818					
4	0.828	0.821	0.884				
8	0.801	0.816	0.845	0.876			
16	0.82	0.818	0.834	0.916	1.025		
32	0.822	0.804	0.858	0.914	1.099	1.596	
64	0.834	0.827	0.865	0.918	1.049	1.587	3.304
128	0.828	0.821	0.844	0.918	1.026	1.574	3.268
256	0.829	0.853	0.91	0.936	1.092	1.594	3.341
512	0.901	0.904	0.94	1.014	1.145	1.695	3.221
1024	1.423	1.209	1.155	1.241	1.426	1.719	3.44
2048	3.349	2.868	2.439	2.536	2.52	2.859	5.851
4096	10.184	8.513	8.01	6.909	7.105	7.855	13.618

5.

	1	2	4	8	16	32	64
2	1	0.995					
4	1	1.009	0.937				
8	1	0.982	0.948	0.914			
16	1	1.002	0.983	0.895	0.800		
32	1	1.022	0.958	0.899	0.748	0.515	
64	1	1.008	0.964	0.908	0.795	0.526	0.252
128	1	1.009	0.981	0.902	0.807	0.526	0.253
256	1	0.972	0.911	0.886	0.759	0.520	0.248
512	1	0.997	0.959	0.889	0.787	0.532	0.280
1024	1	1.177	1.232	1.147	0.998	0.828	0.414
2048	1	1.168	1.373	1.321	1.329	1.171	0.572
4096	1	1.196	1.271	1.474	1.433	1.296	0.748

6. Number of cores: 32  
Number of threads: 64

7.

a. I don't get speedup when both the number of cores is above 2 and the number of unknowns is above 512. I also get very slight or no speedup when the number of processes is 2 for 512 unknowns and below. I also don't get speedup when the number of unknowns is 1024 for 16 and 32 processors.

b. I don't get speedup in the case above because the number of processes is too many to be efficient for the number of unknowns I have. The cost of communication outweighs the

benefits of parallel processes running, because there is not too much computation to be done. Thus, the tasks can be better handled by a fewer number of cores. Communication between cores is the most expensive kind of task.

c. I get speedup whenever the number of unknowns is greater than 2048 and number of cores is less than 64, and when the number of unknowns is greater than 1024 when the number of cores is 2, 4 or 8.

d. In these cases, the cost of communication is not outweighed by the benefits of parallel processes running. When there is a lot of computation to be done, multiple cores really have an advantage because computation happens very quickly. However, the less communication to be used between cores the better. Notice how for 2048 and 4096 unknowns, after you add more than 16 cores, speedup starts going down. This is because there is too much communication happening between cores to be becoming more efficient, but speedup is still observable up to 32 cores because there is so much computation to do.