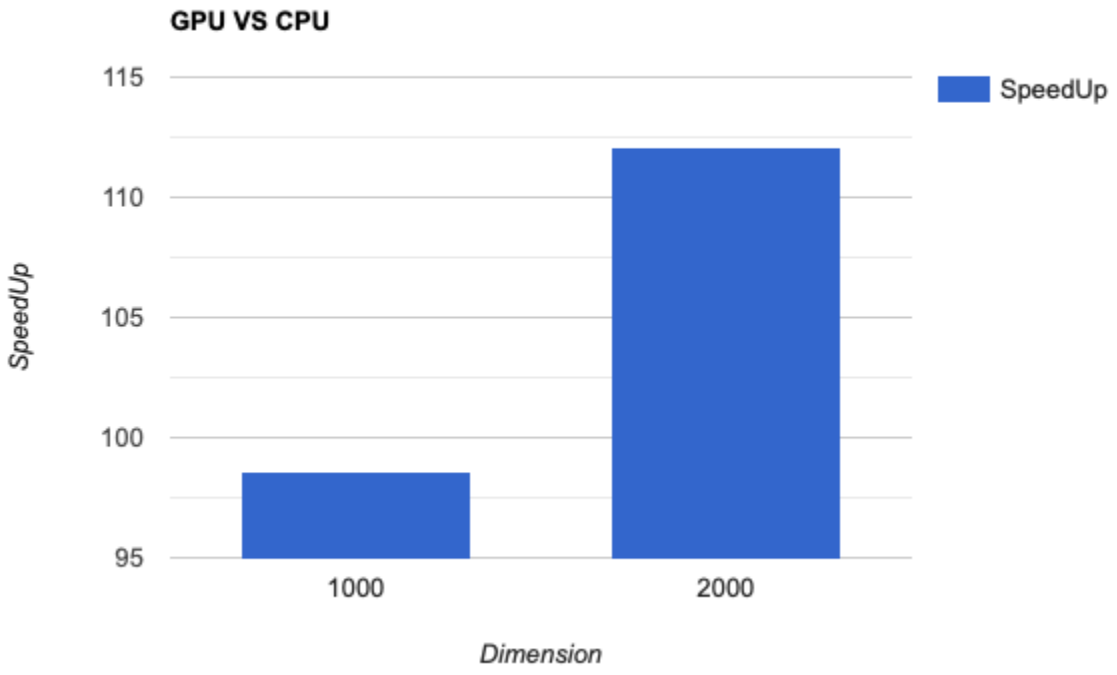
2.

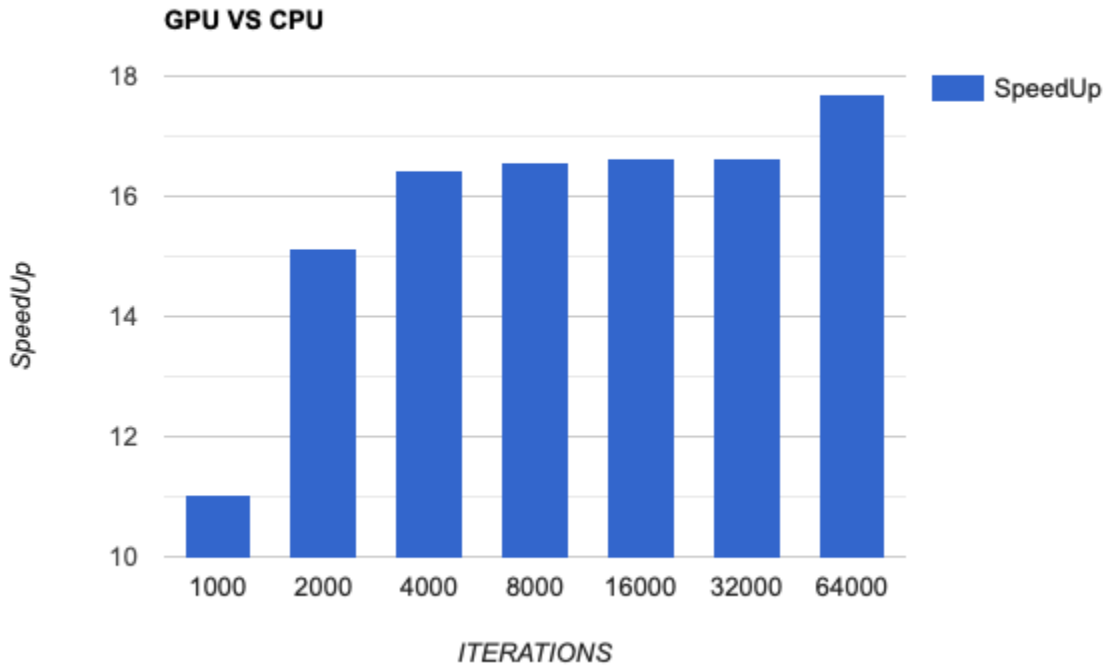
DIMENSION	CPU TIME(S)	GPU TIME(S)	SPEEDUP
1000	962.63	9.76	98.6
2000	3809.34	33.98	112.1
4000	>1.5 HRS	205.85	
8000	>1.5 HRS	3281.38	
16000	>1.5 HRS	10320.36	



3.

ITERATIONS	CPU TIME(S)	GPU TIME(S)	SPEEDUP
1000	10.91	0.99	11.02
2000	23.33	1.54	15.14
4000	43.03	2.62	16.42
8000	80.24	4.84	16.57

16000	153.08	9.2	16.63
32000	300.68	18.07	16.63
64000	590.40	33.35	17.7



4.

a.There is no data for the CPU time for the dimensions more than 2000. However, observing the speedup until 2000 it is clear that GPU usage is best for DIM=2000, and the pattern seems to indicate that the speedup will increase for higher dimensions. The GPU usage for DIM=2000 is better because the ratio of cost for the transfer of memory from and to the host and device is lower (cudaMemcpy) .This means that it would likely be more beneficial for a higher dimension , as the speedup caused by GPU operations would more effectively offset the memory transfer cost. It also implies that at a certain stage the cost of GPU operations would be much larger than memory transfer for it to matter and the speedup would stagnate.

b.The speedup is at the lowest for DIM=1000 and ITER=1000. For the dimension graph, the memory transfer cost slows down the GPU as indicated above. For the iteration graph, we know that there are cudaMemcpy transfers before and after running the kernel, for lower iterations the ratio of the memory transfer time to GPU kernel time is much higher and results in a lower speedup.

c.The speedup is at the highest for DIM=2000 and ITER=64000. For the dimension graph and the iteration graph, the ratio of cost of all the calls to cudaMemcpy reduces compared to the other GPU operations which results in more speedup.

d.The problem size has more effect than the number of iterations. This is because the increase in time taken due to number of iterations is directly proportional to the increase in number of iterations as the GPU runs the same kernel sequentially.However, when the problem size is increased, the GPU has more opportunity to parallelize operations for the elements in the heating matrix, this would make it more effective and increase speedup significantly.