Machine: Core i5, 2.80Ghz, 2 Physical core, 4 Hyperthread Core , GPU GTX 970 1664 cores . All time in milliseconds (ms)

		THREADS_PER_BLOCK = 64		THREADS_PER_BLOCK = 128 = NUM_OF_CORES_PER_SM, TOTAL_BLOCKS=13=NUM_OF_SMs, Persistent threads=1664=Number of cores		
#input = 1000	**Serial**	**G1: GPU Base**	**G2: GPU Persistant Memory (10000 requests, Batch size 1000)**	**G3: GPU Persistant Memory + Persistent Kernel (1000 requests, 10 iterations)**	**JCUDA GPU Time (1000 requests, 10 iterations)**	**JCUDA CPU Time(1000 requests, 10 iterations)**
Run 1	23.67	3.04	2.16	3.475	3.092	3.188
Run 2	22.9	3.26	2.088	3.387	3.268	3.079
Run 3	26.4	3.09	2.2255	3.824444444	3.176	3.18
Run 4	23.19	3.38	2.172	3.708888889	3.1	3.24
Run 5	21.42	3.43	2.251	3.636666667	3.128	3.13
Mean	**23.516**	**3.24**	**2.1793**	**3.6064**	**3.1528**	**3.1634**
Speed Up Over serial		**7.3**	**10.8**	**6.5**		

G2 performs better than G1 and G3. G2 uses a persistent memory buffer to store the input values i.e. memory is reused for the computation of next batch. But a separate kernel call is made for each batch in G2. In case of G3, persistent memory and persistent kernels are used. G3 performs worst because of need of synchronization between CPU and GPU (signalling persistent threads). G1 is base case without persistent memory and persistent threads. Also, persistent kernel (G3) has one disadvantage that it keeps the GPU occupied for all time (leads to wastage of computation cycles and power).

#input = 10000	**Serial**	**GPU Base**	**GPU Persistant Memory (100000 requests, Batch size 10000)**	**GPU Persistant Memory + Persistent Kernel (10000 requests, 10 iterations)**	**JCUDA GPU Time (10000 requests, 10 iterations)**	**JCUDA CPU Time(1000 requests, 10 iterations)**
Run 1	192.24	12.42	11.9898	19.09333333	20.156	23.231
Run 2	196.77	14.45	12.68	18.21222222	22.752	22.996
Run 3	196.18	13.22	13.0359	19.94222222	22.733	22.834
Run 4	194.45	14.44	12.411	20.38555556	22.409	23.049
Run 5	197.61	13.63	12.848	21.12777778	22.055	22.063
Mean	**195.45**	**13.632**	**12.59294**	**19.75222222**	**22.021**	**22.8346**
Speed Up Over serial		**14.3**	**15.5**	**9.9**		

#input = 50000	**Serial**	**GPU Base**	**GPU Persistant Memory (500000 requests, Batch size 50000)**	**GPU Persistant Memory + Persistent Kernel (50000 requests, 10 iterations)**
Run 1	973.47	58.77	54.12111111	85.26555556
Run 2	967.96	61.52	55.035	84.54333333
Run 3	970.62	64.03	52.25333333	84.31125
Run 4	968.26	55.6	54.893	76.26666667
Run 5	969	64	54.909	76.26888889
Mean	**969.862**	**60.784**	**54.24228889**	**81.33113889**
Speed Up Over serial		**16**	**17.9**	**11.9**

Summary

	Serial	**G1**	**G2**	**G3**
1000	**23.516**	**3.24**	**2.1793**	**3.6064**
10000	**195.45**	**13.632**	**12.59294**	**19.75222222**
50000	**969.862**	**60.784**	**54.24228889**	**81.33113889**
Speed Up 1000		**7.3**	**10.8**	**6.5**
Speed Up 10000		**14.3**	**15.5**	**9.9**
Speed Up 50000		**16**	**17.9**	**11.9**