

# Homework #3: Concurrent Data Structures 개발/디버깅

## Part 1 : fine-grained lock BST



2013011800 구장희



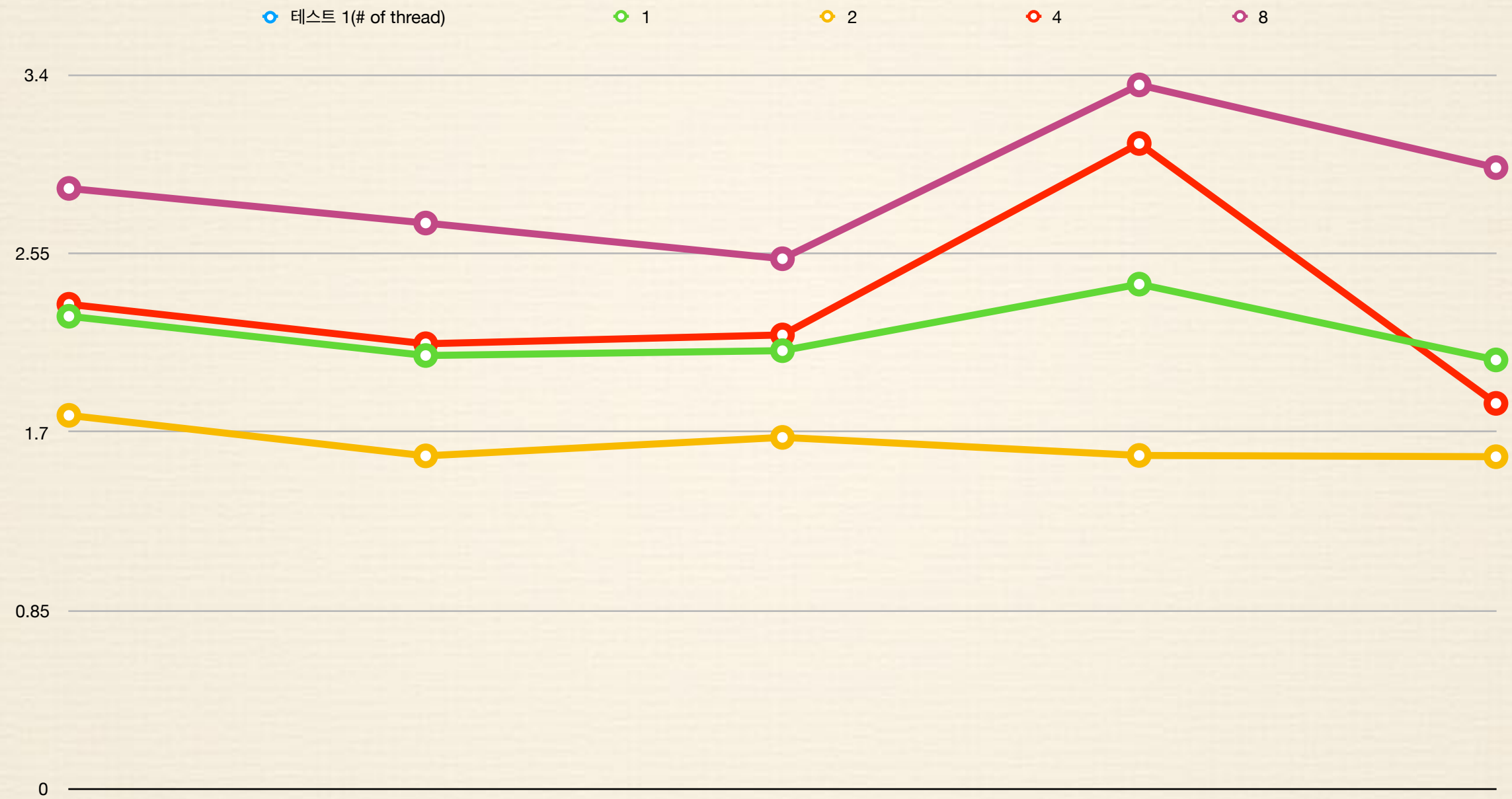
# 실행 결과의 표( 5번씩 테스트)

표 1

	1회	2회	3회	4회	5회
테스트 1(# of thread)					
1	2.251	2.064	2.087	2.404	2.043
2	1.779	1.586	1.674	1.588	1.582
4	2.308	2.12	2.162	3.074	1.836
8	2.86	2.695	2.525	3.353	2.959
테스트 2(ReentrantLock)(# of thread_insert_search)					
1_1_1	4.455	4.61	4.662	4.599	4.56
1_1_4	4.033	3.902	3.954	3.98	2.703
1_1_9	2.475	2.42	2.29	2.296	2.31
2_1_1	4.222	3.324	3.094	3.181	3.343
2_1_4	4.242	3.568	3.003	2.522	2.788
2_1_9	2.416	2.977	2.62	2.229	2.269
4_1_1	5.448	3.876	4.194	3.987	4.079
4_1_4	3.828	2.803	3.147	2.765	2.771
4_1_9	2.39	2.322	2.895	2.606	2.861
8_1_1	4.626	3.89	4.046	4.335	3.896
8_1_4	2.822	3.193	2.772	3.144	3.052
8_1_9	2.376	3.088	2.731	2.432	2.465
테스트 2(ReentrantReadWriteLock)(# of thread_insert_search)					
1_1_1	10.115	7.726	9.89	9.727	9.142
1_1_4	9.793	6.895	8.65	6.68	7.472
1_1_9	6.246	5.766	5.428	5.795	4.63
2_1_1	9.502	7.196	7.13	7.806	7.582
2_1_4	9.577	7.263	7.561	7.144	6.071
2_1_9	6.482	4.731	5.026	4.571	4.665
4_1_1	9.176	6.983	9.51	7.916	8.167
4_1_4	9.895	6.141	6.616	5.849	5.618
4_1_9	6.404	6.561	4.983	5.195	4.927
8_1_1	9.523	7.668	7.952	10.298	8.536
8_1_4	8.907	6.739	7.294	6.412	6.424
8_1_9	6.402	6.591	5.681	6.138	4.958

a. 100만개의 랜덤한 숫자를 BST에 insert할 때 thread 1개, 2개, 4개, 8개로 나누어서 insert할 경우 실행시간이 어떻게 되는지 그래프로 그리고 설명하세요.

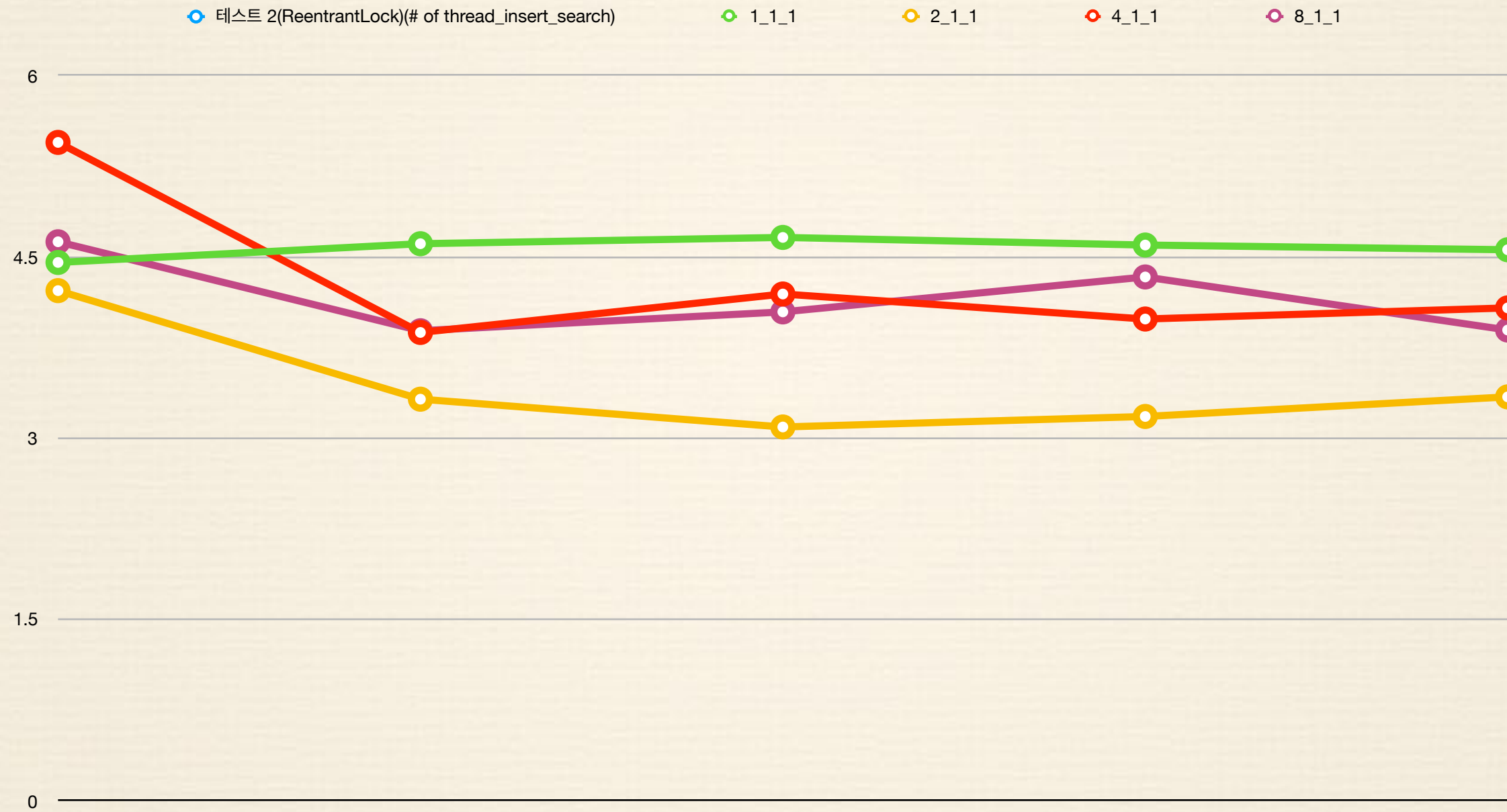
\* # of thread = 1 , 2 , 4 , 8



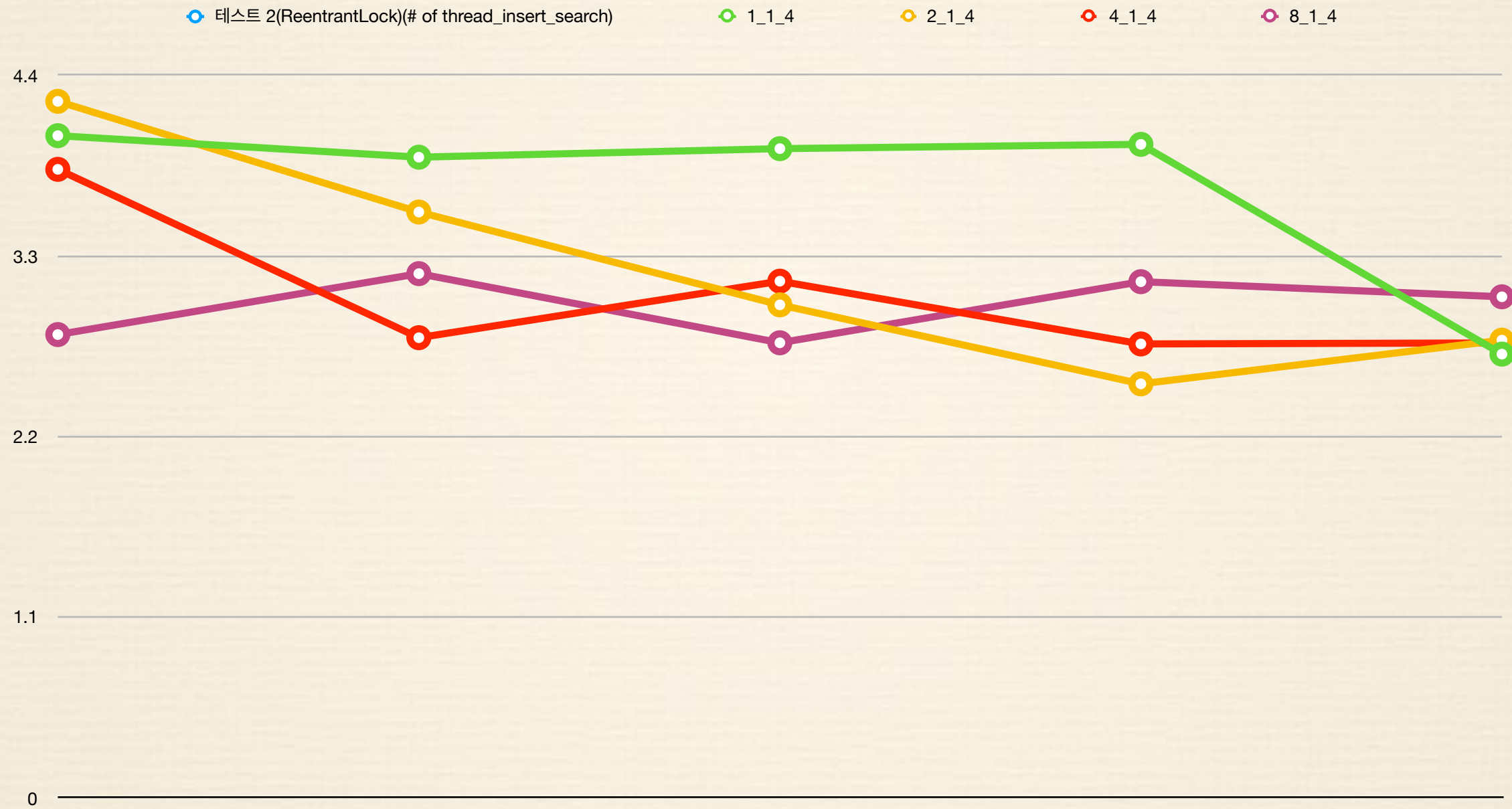


b. 100만개의 랜덤한 숫자를 BST에 insert한 후 추가로 100만개의 insert/search operation을 thread 1,2,4,8개로 실행할 때 실행시간에 대해서 그래프로 그리고 설명하세요. Insert 와 search 비율은 1:1, 1:4, 1:9로 해서 실험하세요 (search 가 더 많게).

\* # of thread = 1,2,4,8 , insert:search = 1:1

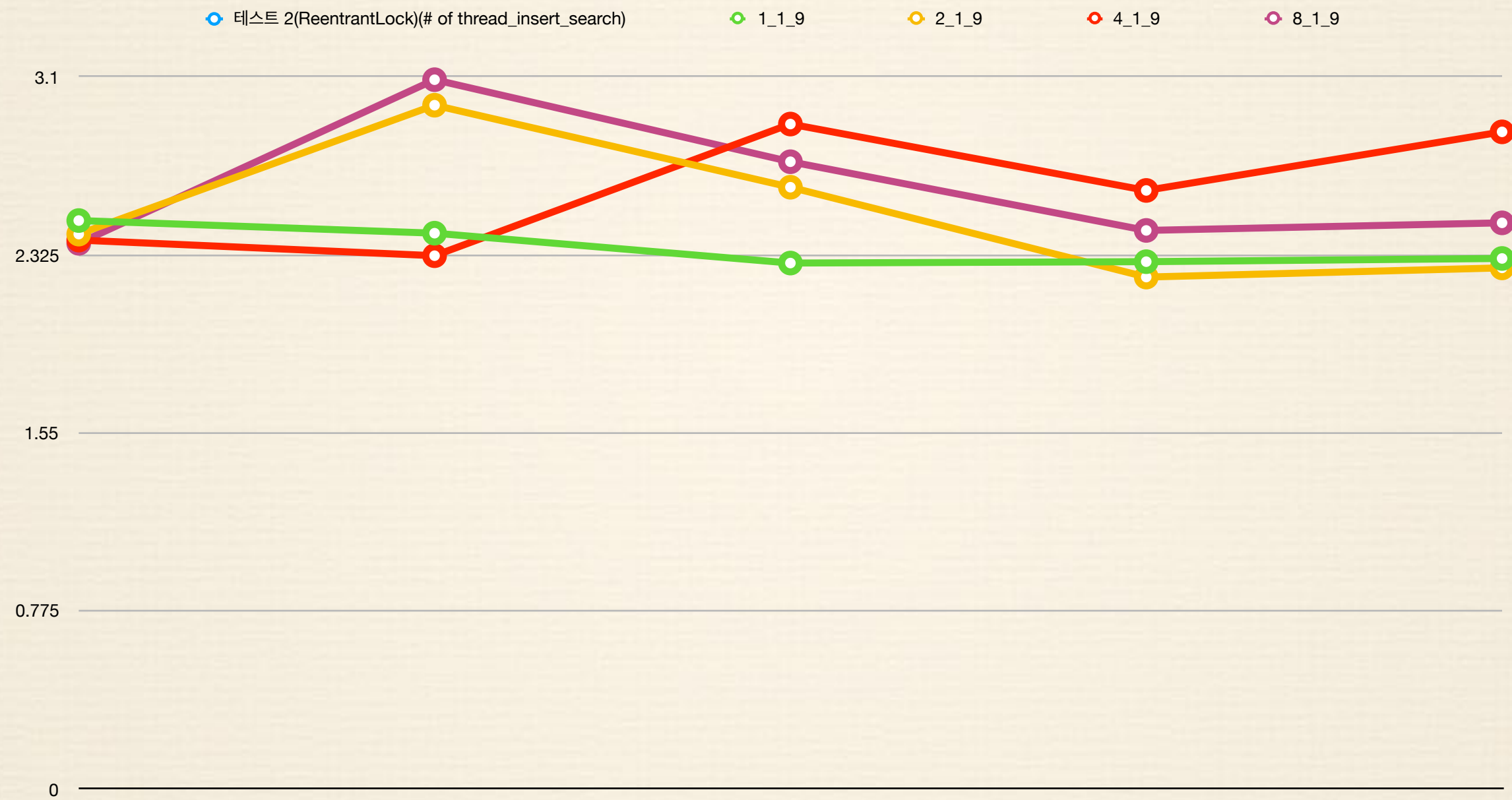


\* # of thread = 1,2,4,8, insert:search = 1:4





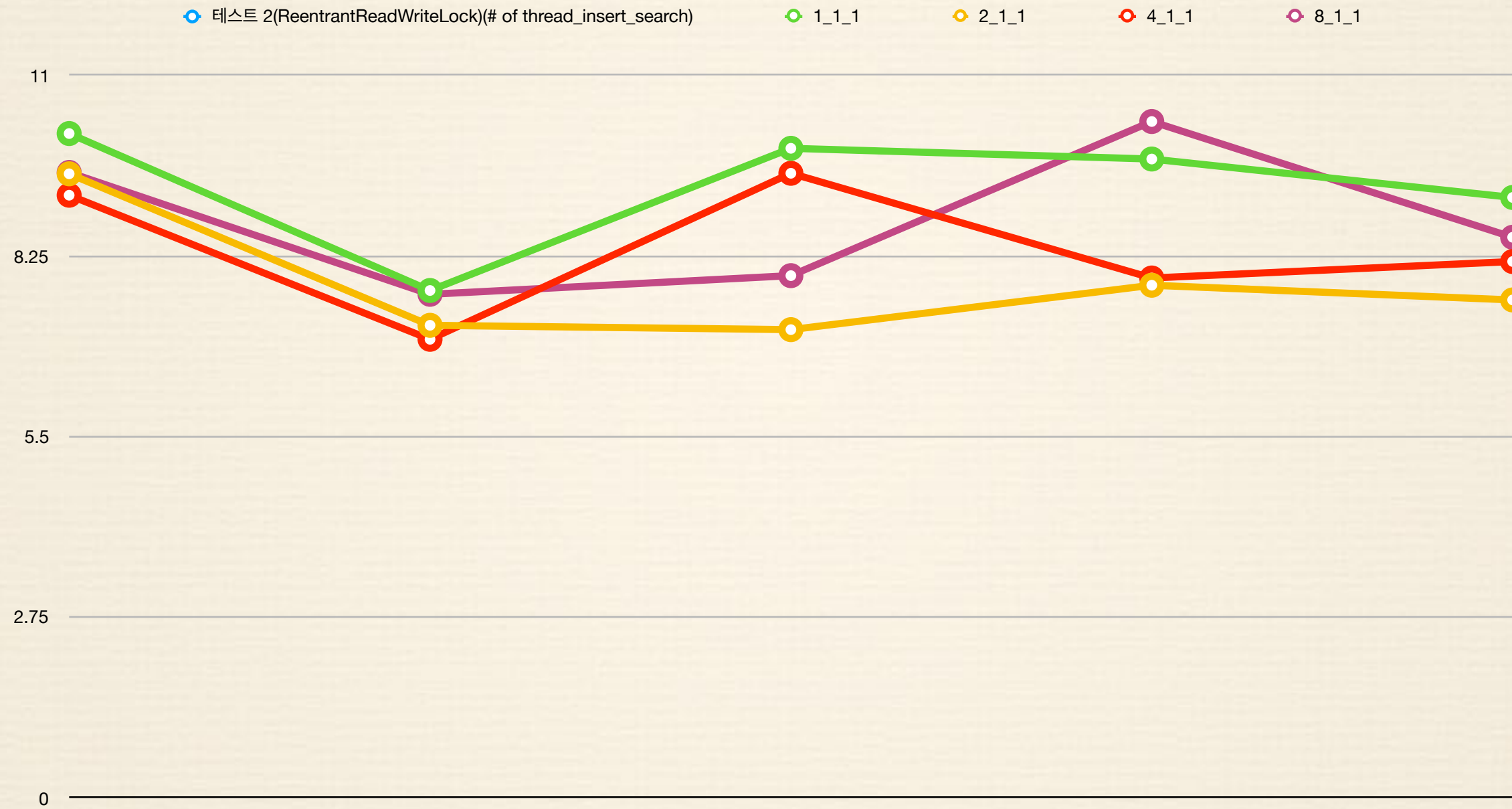
\* # of thread = 1,2,4,8 , insert:search = 1:9





c. 구현에 사용한 lock을 ReadWriteLock으로 바꿔서 b의 실험을 반복하고 성능이 어떻게 달라지는지 설명하세요.

\* # of thread = 1,2,4,8 , insert:search = 1:1





\* # of thread = 1,2,3,8, insert:search = 1:4



\* # of thread = 1,2,4,8 , insert:search = 1:9

