Homework #3: Concurrent Data Structures 개발/디버깅 Part 2: Lock-Free BST

-110 OIL

2013011800 구장회

실행 결과의 표(10번씩 테스트) 표1

	1	2	3	4	5	6	7	8	9	1
TEST1(# of thread)										
1	4.475	2.484	3.419	3.229	2.321	3.106	3.346	2.592	2.27	2.14
2	2.837	1.575	2.096	1.614	1.82	2.633	1.401	1.571	2.632	1.36
4	2.557	1.232	1.607	1.375	1.14	2.613	0.903	0.74	0.622	0.56
8	2.463	1.181	1.385	1.264	1.649	0.775	0.934	1.787	0.888	0.78
TEST2(insert : search : # of thread)										
1:1_1	7.386	5.835	6.787	6.449	5.321	5.435	7.067	5.362	5.379	5.29
1:1_2	9.222	6.269	9.32	4.775	6.742	5.849	6.421	4.627	5.196	4.57
1:1_4	6.518	4.425	4.924	5.423	3.896	4.123	5.003	3.417	3.644	3.66
1:1_8	7.406	4.533	5.228	4.899	3.542	4.245	5.192	3.602	3.904	3.0
1:4_1	8.481	5.052	4.938	4.624	4.685	6.08	4.83	5.504	4.54	7.49
1:4_2	7.274	5.223	3.346	4.165	4.486	3.476	4.0	3.668	3.262	4.0
1:4_4	5.025	3.478	3.123	2.981	4.96	4.597	3.107	3.034	3.582	3.7
1:4_8	7.012	4.711	3.007	3.994	2.905	3.239	4.26	3.328	4.242	3.1
1:9_1	5.949	4.771	4.354	5.455	5.483	4.155	5.799	4.161	4.031	4.
1:9_2	5.752	4.274	3.791	3.846	4.057	3.422	3.814	3.853	3.349	3.8
1:9_4	5.172	3.603	3.661	3.368	4.475	3.258	3.843	3.471	3.12	
1:9_8	6.094	3.092	3.386	3.349	3.465	2.835	2.675	2.991	3.471	3.60
TEST3(

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

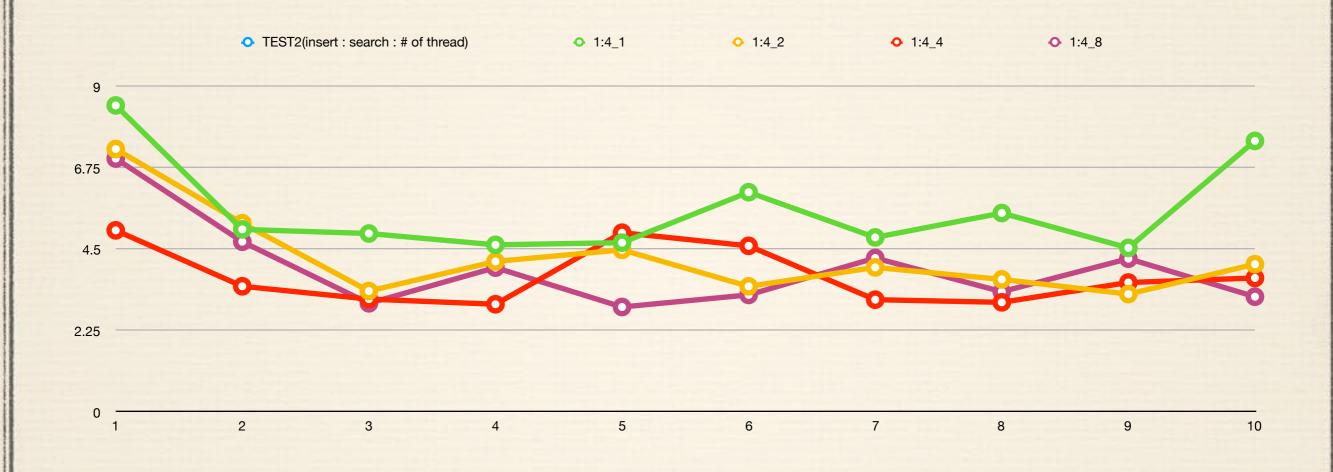


b.100만개의 랜덤한 숫자를 BST에 insert한 후 추가로 100만개의 insert/search operation을 thread 1,2,4,8개로 실행할 때 실행시간에 대해서 그래프로 그리고 설명하세요. Insert 와 search 비율은 1:1, 1:4, 1:9로 해서 실험하세요 (search 가 더 많게).

* # 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

