# 자료구조설계

제 출 일	2017.11.15.
과 제 번 호	09
분 반 학 과	01
학 과	컴퓨터공학과
학 번	201602038
이 름	이 미 진

### 1. 구현 내용 설명

#### 1-1. 코드 설명

```
package DS01_09_201602038;
import java.util.ArrayList;
import java.util.Collections;
public class TestMain {
        public static void main(String args[]) {
        long start, end;
        ArrayList<Object> arrayl = new ArrayList<>();
BinarySearchTree bst;
        QuadraticProbingHashTable qph = new QuadraticProbingHashTable();
        AVI.Tree avl:
        int i;
        int n=10000;
        for(;n<10000000); ){ // n이 10000, 100000, 1000000 일 동안 반복
                 int[] arrayi = new int[n];
                 System.out.println("n = "+n);
System.out.println("***** Insert *****");
                 for( i = 0; i < n; i++ ) {
                          arrayl.add(i);
                 Collections.shuffle(arrayl);
                                                   // 무작위 추출
                 for( i = 0; i < n; i++ ) {
                          arrayi[i]= (int) arrayl.get(i);
                 /* bst Insert start */
                 start = System.currentTimeMillis();
                 bst = new BinarySearchTree((Comparable<?>) arrayl.get(0));
                 for( i = 0; i < n; i++ ) {
                          bst.insert(arrayi[i]);
                 end = System.currentTimeMillis();
                 System.out.println("BST insert: "+(end-start)+"ms");
                 /* bst Insert end */
                 /* qph Insert start */
                 start = System.currentTimeMillis();
                 for(i = 0; i < n; i++) {
                          qph.put(arrayi[i], arrayi[i]);
                 end = System.currentTimeMillis();
                 System.out.println("QPH insert: "+(end-start)+"ms");
                 /* qph Insert end */
                 /* avl Insert start */
                 start = System.currentTimeMillis();
                 avl = new AVLTree((int) arrayl.get(0));
                 for( i = 1; i < n; i++ ) {
                          avl.grow(arrayi[i]);
                 }
```

```
end = System.currentTimeMillis();
         System.out.println("AVL insert: "+(end-start)+"ms\n");
         /* avl Insert end */
         System.out.println("***** Search *****");
         /* bst Search start */
         start = System.currentTimeMillis();
         for(i = 0; i < n; i++) {
                  bst.contains(i);
         end = System.currentTimeMillis();
         System.out.println("BST Search: "+(end-start)+"ms");
         /* bst Search end */
/* qph Search start */
start = System.currentTimeMillis();
         for( i = 0; i < n; i++ ) {
                  qph.get(i);
         end = System.currentTimeMillis();
System.out.println("QPH Search : "+(end-start)+"ms");
         /* qph Search end */
         /* avl Search start */
         start = System.currentTimeMillis();
         for( i = 0; i < n; i++ ) {
                  avl.search(i);
         end = System.currentTimeMillis();
         System.out.println("AVL Search: "+(end-start)+"ms\n");
         /* avl Search end */
         n = n*10;
}
```

각 알고리즘별 성능을 측정하기 위해 System.currentTimeMillis()로 BinarySearchTree, QuadraticProbingHashTable, AVLTree 의 삽입, 검색을 수행할 동안의 시작 시간과 종료시간을 기록한다. 그 후에 end-start 로 삽입과 검색에 걸린 시간을 출력한다. 이번 과제에서는 Collections 클래스에 있는 Collections.shuffle 메소드를 사용하였는데, List나 배열의 내용을 랜덤으로 리턴해주는 기능을 한다. n만큼 I를 증가시키면서 arrayl에 I를 add 해주고, arrayl에 있는 값들을 랜덤으로 리턴해주면서 arrayi에 arrayl에서 get한 값을 저장해준다. 알고리즘을 수행할 때에는 랜덤으로 리턴된 값이 들어가 있는 arrayi을 이용한다. for문을 통해 n이 10000,1000000,10000000 일 동안 반복하고, 모든 과정을 끝냈을 땐 n에 10을 곱해주어 수를 늘린다.

## 2. 실행 결과 화면

n = 10000 ***** Insert		***** Insert *****	are opt sterringereds, resovant	***** Insert	stMain [Java <terminated> TestMain n = 19999</terminated>
Tueble.			Insert *****	BST insert :	
		BST insert : 4ms	BST insert : 7ms		
SST insert :		QPH insert : 3ms	QPH insert : 5ms	QPH insert :	
OPH insert :		AVL insert : 5ms	AVL insert : 6ms	AVL insert :	
WL insert :	4ms			*****	AVL insert : 6ms
		***** Search *****	***** Search *****		
Search		BST Search : 4ms	BST Search : 4ms	BST Search :	
BST Search :		QPH Search : 1ms	QPH Search : 3ms	QPH Search :	
QPH Search :	1ms	AVL Search : 3ms	AVL Search : 6ms	AVL Search :	4ms QPH Search : 2ms
AVL Search :	4ms				AVL Search : 3ms
		n = 100000	n = 100000	n = 100000	
100000		***** Insert *****	***** Insert *****	***** Insert	
***** Insert	*****	BST insert : 35ms	BST insert : 32ms	BST insert :	31ms "**** Insert *****
BST insert :	28ms	QPH insert : 25ms	QPH insert : 23ms	QPH insert :	19ms BST insert : 39ms
QPH insert :	19ms	AVL insert : 45ms	AVL insert : 36ms	AVL insert :	49ms QPH insert : 28ms
AVL insert :					AVL insert : 35ms
		***** Search *****	***** Search *****	***** Search	****
Search	*****	BST Search : 18ms	BST Search : 24ms	BST Search :	20ms ***** Search *****
SST Search :		QPH Search : 7ms	QPH Search : 8ms	QPH Search :	
QPH Search :		AVL Search : 9ms		AVL Search :	
AVL Search :		CAR SERVER 1 2012	AVL Search : 10ms		AVL Search : 9ms
Ar Search :	2113	n = 1000000	1000000	n = 1000000	Ave search 1 5ms
- 1000000		""" Insert """"	n = 1000000		n = 1000000
			***** Insert *****	ALCOHOL: A	
***** Insert		BST insert : 602ms	BST insert : 593ms	many A	
BST insert :		QPH insert : 2324ms	QPH insert : 2219m		
QPH insert :		AVL insert : 681ms	AVL insert : 668ms	AVL insert :	
AVL insert :	659ms	Carrier and Control Control Control			AVL insert : 668ms
		***** Search *****	***** Search *****	***** Search	
***** Search		BST Search : 99ms	BST Search : 93ms	BST Search :	[H. (2014년
BST Search :	99ms	QPH Search : 19ms	QPH Search : 21ms	QPH Search :	
QPH Search :	19ms	AVL Search : 62ms	AVL Search : 62ms	AVL Search :	62ms QPH Search : 20ms
AVL Search :	62ms				AVL Search : 61ms
terminated > Te	etMain Hava	sterminated > TestMain ()	ava sterminated - TestMain Flag	eterminated - TestMain []:	ava <terminated> TestMain [Java A</terminated>
	SUNIAIII (Sava )				
1 = 10000		h = 10000	n = 10000	n = 10000	h = 10000
**** Insert		***** Insert *****	***** Insert *****	***** Insert *****	***** Insert *****
SST insert :	7ms	BST insert : 7ms	BST insert : 5ms	BST insert : 7ms	BST insert : 7ms
					and ansatz a comme
		QPH insert : 3ms	QPH insert : 4ms	QPH insert : 4ms	QPH insert : 4ms
			QPH insert : 4ms AVL insert : 5ms		
AVL insert :	5ms	QPH insert : 3ms AVL insert : 5ms	AVL insert : 5ms	QPH insert : 4ms AVL insert : 6ms	QPH insert : 4ms AVL insert : 5ms
AVL insert :	5ms	QPH insert : 3ms		QPH insert : 4ms	QPH insert : 4ms
AVL insert :	5ms	QPH insert : 3ms AVL insert : 5ms	AVL insert : 5ms	QPH insert : 4ms AVL insert : 6ms	QPH insert : 4ms AVL insert : 5ms
AVL insert : ***** Search BST Search :	5ms  4ms	QPH insert : 3ms AVL insert : 5ms	AVL insert : 5ms	QPH insert : 4ms AVL insert : 6ms	QPH insert : 4ms AVL insert : 5ms ***** Search ***** BST Search : 5ms
AVL insert : ***** Search BST Search : QPH Search :	5ms  4ms 1ms	QPH insert : 3ms AVL insert : 5ms Search BST Search : 5ms QPH Search : 2ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms	QPH insert : 4ms AVL insert : 6ms ***** Search ***** BST Search : 4ms QPH Search : 1ms	QPH insert : 4ms AVL insert : 5ms  ***** Search ***** BST Search : 5ms QPH Search : 3ms
AVL insert : ***** Search 3ST Search : QPH Search :	5ms  4ms 1ms	QPH insert : 3ms AVL insert : 5ms ***** Search ***** BST Search : 5ms	AVL insert : 5ms  ***** Search ***** BST Search : 4ms	QPH insert : 4ms AVL insert : 6ms ***** Search ***** BST Search : 4ms	QPH insert : 4ms AVL insert : 5ms ***** Search ***** BST Search : 5ms
WVL insert : Search SST Search : QPH Search : WVL Search :	5ms  4ms 1ms	QPH insert: 3ms AVL insert: 5ms ***** Search ***** BST Search: 5ms QPH Search: 2ms AVL Search: 4ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms	QPH insert : 4ms AVL insert : 6ms ***** Search ***** BST Search : 4ms QPH Search : 1ms AVL Search : 5ms	QPH insert : 4ms AVL insert : 5ms ***** Search ***** BST Search : 5ms QPH Search : 3ms AVL Search : 4ms
Search : Search : QPH Search : AVL Search :	5ms ***** 4ms 1ms 5ms	QPH insert : 3ms AVL insert : 5ms **** Search ***** BST Search : 5ms QPH Search : 2ms AVL Search : 4ms n = 100000	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  OPH Search : 2ms  AVL Search : 5ms  n = 100000	QPH insert : 4ms AVL insert : 6ms  ***** Search : 4ms QPH Search : 1ms AVL Search : 5ms n = 100000	QPH insert : 4ms AVL insert : 5ms  ***** Search ***** BST Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000
Search : Search : QPH Search : AVL Search : AVL Search : Tolera :	5ms ***** 4ms 1ms 5ms	QPH insert : 3ms AVL insert : 5ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  **** Insert *****	QPH insert : 4ms AVL insert : 6ms	QPH insert : 4ms AVL insert : 5ms  ***** Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert *****
AVL insert :  Search :  OPH Se	5ms  4ms 1ms 5ms	QPH insert: 3ms AVL insert: 5ms  ***** Search: 5ms QPH Search: 2ms AVL Search: 4ms n = 100000  **** Insert 8ST insert: 33ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  **** Insert ****  BST insert : 31ms	QPH insert: 4ms AVL insert: 6ms  ***** Search ***** BST Search: 1ms QPH Search: 1ms AVL Search: 5ms n = 100000  ***** Insert **** BST insert: 30ms	QPH insert : 4ms AVL insert : 5ms  ***** Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert : 39ms  BST insert : 39ms
VL insert :  """ Search :  PH Search :  VL Search :  " = 100000  """ Insert :  PH insert :	5ms 4ms 1ms 5ms 30ms 19ms	QPH insert: 3ms AVL insert: 5ms  **** Search **** ST Search: 5ms QPH Search: 2ms AVL Search: 4ms n = 100000 **** Insert: 33ms QPH insert: 33ms QPH insert: 22ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  ***** Insert ****  BST insert : 31ms  QPH insert : 24ms	QPH insert: 4ms AVL insert: 6ms	QPH insert : 4ms AVL insert : 5ms  ***** Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert ***** BST insert : 39ms QPH insert : 23ms
VVL insert :  """ Search :  PH Search :  VVL Search :  n = 100000  """ Insert :  QPH insert :	5ms 4ms 1ms 5ms 30ms 19ms	QPH insert: 3ms AVL insert: 5ms  ***** Search: 5ms QPH Search: 2ms AVL Search: 4ms n = 100000  **** Insert 8ST insert: 33ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  **** Insert ****  BST insert : 31ms	QPH insert: 4ms AVL insert: 6ms  ***** Search ***** BST Search: 1ms QPH Search: 1ms AVL Search: 5ms n = 100000  ***** Insert **** BST insert: 30ms	QPH insert : 4ms AVL insert : 5ms  ***** Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert : 39ms  BST insert : 39ms
SEARCH : Search : SPH Search : SVL Search : NVL Search : N = 100000  """ SST insert : NPH insert : NVL insert :	5ms 4ms 1ms 5ms 30ms 19ms 40ms	QPH insert: 3ms AVL insert: 5ms  ***** Search: 5ms QPH Search: 2ms AVL Search: 4ms n = 100000  **** Insert: 33ms QPH insert: 22ms AVL insert: 38ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 5ms  n = 100000  ***** Insert : 31ms  QPH insert : 24ms  AVL insert : 39ms	QPH insert : 4ms AVL insert : 6ms  ***** Search ***** BST Search : 4ms QPH Search : 1ms AVL Search : 5ms  n = 100000  ***** Insert **** BST insert : 30ms QPH insert : 20ms AVL insert : 39ms	QPH insert: 4ms AVL insert: 5ms  ***** Search: 5ms  BST Search: 3ms AVL Search: 4ms  n = 100000  ***** Insert: 39ms  QPH insert: 23ms  AVL insert: 41ms
SET INSERT:  """ Search:  """ Search:  """ Search:  """ Search:  """ Insert:  """ Insert:  """ Search:  """ Search:  """ Search	5ms 4ms 1ms 5ms 30ms 19ms 40ms	QPH insert: 3ms AVL insert: 5ms  ***** Search ****  BST Search: 5ms QPH Search: 2ms AVL Search: 4ms  n = 100000  ***** Insert  BST insert: 33ms QPH insert: 22ms AVL insert: 38ms  ****** Search	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  ***** Insert : 31ms  QPH insert : 24ms  AVL insert : 39ms  ***** Search *****	QPH insert : 4ms AVL insert : 6ms	QPH insert : 4ms AVL insert : 5ms  ***** Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert ***** BST insert : 39ms QPH insert : 23ms AVL insert : 41ms  ***** Search *****
VL insert :  """ Search :  PM Search :  PM Search :  1 = 100000  "Insert :  PM insert :  PM insert :  Search :  Sear	5ms 4ms 1ms 5ms 5ms 40ms 19ms 40ms	QPH insert: 3ms AVL insert: 5ms  Search: 5ms QPH Search: 2ms AVL Search: 4ms  n = 100000  Insert: 33ms QPH insert: 22ms AVL insert: 38ms AVL insert: 38ms  From Search: 24ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  **** Insert ****  BST insert : 31ms  QPH insert : 24ms  AVL insert : 39ms  **** Search ****  BST Search ****	QPH insert: 4ms AVL insert: 6ms  ***** Search : 4ms BST Search: 1ms AVL Search: 5ms n = 100000  **** Insert: 30ms QPH insert: 20ms AVL insert: 39ms AVL insert: 39ms AVL insert: 34ms BST Search BST Search: 24ms	QPH insert : 4ms AVL insert : 5ms  ***** Search ***** BST Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert **** BST insert : 23ms AVL insert : 41ms  **** Search **** BST Search **** BST Search : 21ms
WL insert :  Search :  WE search :  WL Search :  Insert :  ST insert :  WL insert :  Search :  SET Search :  WH Search :	5ms 4ms 1ms 5ms 30ms 19ms 40ms 20ms 7ms	QPH insert: 3ms AVL insert: 5ms	AVL insert : 5ms  ***** Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  ***** Insert : 31ms  QPH insert : 31ms  QPH insert : 39ms  ***** Search : 28ms  QPH Search : 28ms  QPH Search : 9ms	QPH insert : 4ms AVL insert : 6ms  ***** Search ****  BST Search : 4ms QPH Search : 1ms AVL Search : 5ms  n = 100000  **** Insert *** BST insert : 30ms QPH insert : 20ms AVL insert : 39ms  **** Search **** BST Search **** BST Search : 24ms QPH Search : 7ms	QPH insert : 4ms AVL insert : 5ms  ***** Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert : 39ms QPH insert : 23ms AVL insert : 41ms  **** Search : 21ms QPH Search : 21ms QPH Search : 8ms
WL insert :  Search :  WE search :  WL Search :  Insert :  ST insert :  WL insert :  Search :  SET Search :  WH Search :	5ms 4ms 1ms 5ms 30ms 19ms 40ms 20ms 7ms	QPH insert: 3ms AVL insert: 5ms  Search: 5ms QPH Search: 2ms AVL Search: 4ms  n = 100000  Insert: 33ms QPH insert: 22ms AVL insert: 38ms AVL insert: 38ms  From Search: 24ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  **** Insert ****  BST insert : 31ms  QPH insert : 24ms  AVL insert : 39ms  **** Search ****  BST Search ****	QPH insert: 4ms AVL insert: 6ms  ***** Search : 4ms BST Search: 1ms AVL Search: 5ms n = 100000  **** Insert: 30ms QPH insert: 20ms AVL insert: 39ms AVL insert: 39ms AVL insert: 34ms BST Search BST Search: 24ms	QPH insert : 4ms AVL insert : 5ms  ***** Search ***** BST Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert **** BST insert : 23ms AVL insert : 41ms  **** Search **** BST Search : 21ms
WL insert :  Search SST Search : WL Search : WL Search : In = 100000 Insert : SST insert : WH insert : Search SST Search : WH Search : WH Search : WH Search :	5ms 4ms 1ms 5ms 30ms 19ms 40ms 20ms 7ms	QPH insert: 3ms AVL insert: 5ms  *****Search: 5ms QPH Search: 2ms AVL Search: 4ms n = 100000  *****Insert: 33ms QPH insert: 22ms AVL insert: 38ms AVL insert: 38ms AVL 5earch: 24ms BST Search: 24ms QPH Search: 7ms AVL Search: 10ms	AVL insert : 5ms  ***** Search *****  8ST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  ***** Insert : 31ms  QPH insert : 24ms  AVL insert : 39ms  **** Search *****  BST Search ****  BST Search : 28ms  QPH Search : 9ms  AVL Search : 10ms	QPH insert: 4ms AVL insert: 6ms  ***** Search ***** BST Search: 1ms AVL Search: 5ms n = 100000  **** Insert *** BST insert: 30ms QPH insert: 20ms AVL insert: 39ms AVL insert: 39ms AVL Search *** BST Search *** BST Search *** Search *** QPH Search: 7ms AVL Search: 9ms	QPH insert : 4ms AVL insert : 5ms  ***** Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert : 39ms QPH insert : 23ms AVL insert : 41ms  ***** Search : 41ms  BST Search : 41ms  ***** Search : 21ms QPH Search : 8ms AVL Search : 10ms
WL insert :  Search SST Search : WL Search : WL Search : Insert : SST insert : WH insert : WH insert : WL insert : WL Search :	5ms 4ms 1ms 5ms 30ms 19ms 40ms 20ms 9ms	QPH insert: 3ms AVL insert: 5ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  ***** Insert : 31ms  QPH insert : 24ms  AVL insert : 39ms  ***** Search ***  BST Search : 28ms  QPH Search : 28ms  QPH Search : 10ms  n = 1000000	QPH insert : 4ms AVL insert : 6ms	QPH insert: 4ms AVL insert: 5ms  ***** Search: 5ms QPH Search: 3ms AVL Search: 4ms  n = 100000  ***** Insert: 39ms AVL insert: 23ms AVL insert: 41ms  ***** Search BST Search: 21ms QPH Search: 8ms AVL Search: 10ms  n = 1000000
WL insert :  """ Search : SET Search : WL Search : WL Search : Insert : WL search : WH Search : HE SEA	5ms 4ms 1ms 5ms 30ms 19ms 40ms 20ms 7ms 9ms	QPH insert: 3ms AVL insert: 5ms  ***** Search: 5ms QPH Search: 2ms AVL Search: 4ms  n = 1800000 ***** Insert: 33ms QPH insert: 33ms QPH insert: 38ms AVL insert: 38ms AVL insert: 10ms BST Search: 24ms QPH Search: 10ms n = 1000000 ***** Insert: 10ms n = 1000000	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  **** Insert : 31ms  QPH insert : 24ms  AVL insert : 39ms  **** Search ****  BST Search : 28ms  QPH Search : 9ms  AVL Search : 10ms  n = 1000000  **** Insert ****	QPH insert : 4ms AVL insert : 6ms	QPH insert : 4ms AVL insert : 5ms  ***** Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert : 39ms QPH insert : 23ms AVL insert : 41ms  ***** Search : 41ms  BST Search : 41ms  ***** Search : 21ms QPH Search : 8ms AVL Search : 10ms
WL insert :  """ Search : SET Search : WL Search : WL Search : Insert : WL search : WH Search : HE SEA	5ms 4ms 1ms 5ms 30ms 19ms 40ms 20ms 7ms 9ms	QPH insert: 3ms AVL insert: 5ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  ***** Insert : 31ms  QPH insert : 24ms  AVL insert : 39ms  ***** Search ***  BST Search : 28ms  QPH Search : 28ms  QPH Search : 10ms  n = 1000000	QPH insert: 4ms AVL insert: 6ms  ***** Search **** BST Search: 1ms AVL Search: 5ms  n = 100000  ******* BST insert: 30ms QPH insert: 20ms AVL insert: 39ms AVL insert: 39ms  BST Search: 24ms QPH Search: 7ms AVL Search: 9ms  n = 1000000  ***** Insert** BST insert** BST insert** BST Search: 585ms	QPH insert: 4ms AVL insert: 5ms  ***** Search: 5ms  BST Search: 3ms AVL Search: 4ms  n = 100000  ***** Insert: 39ms AVL insert: 23ms AVL insert: 41ms  ***** Search BST Search: 21ms AVL Search: 21ms AVL Search: 10ms  n = 1000000
WL insert :  Search SST Search : WL Search : WL Search : Insert : SST Insert : WL insert :	5ms 4ms 1ms 5ms 30ms 19ms 40ms 20ms 7ms 9ms	QPH insert: 3ms AVL insert: 5ms  ***** Search: 5ms QPH Search: 2ms AVL Search: 4ms  n = 1800000 ***** Insert: 33ms QPH insert: 33ms QPH insert: 38ms AVL insert: 38ms AVL insert: 10ms BST Search: 24ms QPH Search: 10ms n = 1000000 ***** Insert: 10ms n = 1000000	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  **** Insert : 31ms  QPH insert : 24ms  AVL insert : 39ms  **** Search ****  BST Search : 28ms  QPH Search : 9ms  AVL Search : 10ms  n = 1000000  **** Insert ****	QPH insert : 4ms AVL insert : 6ms	QPH insert : 4ms AVL insert : 5ms  ****** Search ***** BST Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert **** BST insert : 23ms AVL insert : 41ms  ***** Search **** BST Search **** BST Search : 21ms QPH Search : 8ms AVL Search : 10ms  n = 1000000  ***** Insert *****
WL insert :  Search SST Search :  WL Search :  WL Search :  Insert :  SST insert :  Search SST Search :  WL Search :  Search SST Search :  WL Search :  Insert :  Search :  SST Insert :  PM Hinsert :  SST insert :  PM Hinsert :  SST insert :  SEARCH :  SEAR	5ms 4ms 1ms 5ms 1ms 5ms 19ms 40ms 20ms 7ms 9ms 592ms 2290ms	QPH insert: 3ms AVL insert: 5ms  *****Search: 5ms QPH Search: 2ms AVL Search: 4ms n = 100000  *****Insert: 33ms QPH insert: 22ms AVL insert: 38ms QPH insert: 38ms QPH insert: 38ms AVL Search: 10ms  *****Search: 24ms AVL Search: 10ms n = 1000000  *****Insert: 590ms	AVL insert : 5ms	QPH insert: 4ms AVL insert: 6ms  ***** Search **** BST Search: 1ms AVL Search: 5ms  n = 100000  ******* BST insert: 30ms QPH insert: 20ms AVL insert: 39ms AVL insert: 39ms  BST Search: 24ms QPH Search: 7ms AVL Search: 9ms  n = 1000000  ***** Insert** BST insert** BST insert** BST Search: 585ms	QPH insert : 4ms AVL insert : 5ms  ***** Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 1000000  ***** Insert : 39ms QPH insert : 23ms AVL insert : 41ms  ***** Search : 41ms  ***** Search : 21ms QPH Search : 8ms AVL Search : 10ms  n = 10000000  ***** Insert : 581ms
Search :  """ Search :  """ Search :  """ Search :  """ Insert :  """ Search :  "" Search :  """ Sea	5ms 4ms 1ms 5ms 30ms 19ms 40ms 20ms 7ms 9ms 592ms 661ms	QPH insert: 3ms AVL insert: 5ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  ***** Insert : 31ms  QPH insert : 24ms  AVL insert : 39ms  ***** Search ***  BST Search : 28ms  QPH Search : 9ms  AVL Search : 10ms  n = 1000000  ***** Insert : 595ms  QPH insert : 2346ms  AVL insert : 697ms	QPH insert: 4ms AVL insert: 6ms  ***** Search: 4ms QPH Search: 1ms AVL Search: 5ms  n = 100000 ***** Insert: 30ms QPH insert: 20ms AVL insert: 39ms AVL insert: 39ms  ***** Search BST Search: 24ms QPH Search: 7ms AVL Search: 9ms  n = 1000000 ***** Insert  ***** Search BST insert: 585ms QPH insert: 2302ms AVL insert: 662ms	QPH insert : 4ms AVL insert : 5ms  ***** Search ***** BST Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert : 39ms QPH insert : 33ms AVL insert : 41ms  ***** Search *** BST Search : 21ms QPH Search : 10ms  n = 1000000  ***** Insert : **** BST insert : 581ms QPH insert : 2212ms
Search ST Search: WVL Search: WVL Search: N = 100000 Insert: ST Insert: ST Insert: SEARCH: SEA	5ms 4ms 1ms 5ms 30ms 19ms 40ms 20ms 7ms 9ms 592ms 661ms	QPH insert: 3ms AVL insert: 5ms  ***** Search: 5ms QPH Search: 2ms AVL Search: 4ms n = 100000 ***** Insert: 33ms QPH insert: 22ms AVL insert: 38ms QPH insert: 12ms AVL insert: 13ms QPH Search: 7ms AVL 5earch: 10ms n = 1000000 ***** Insert BST insert: 590ms QPH insert: 2252ms QPH insert: 2252ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  **** Insert : 31ms  QPH insert : 24ms  AVL insert : 39ms  **** Search **  BST Search : 28ms  QPH Search : 9ms  AVL Search : 10ms  n = 1000000  **** Insert **  BST insert : 595ms  QPH insert : 2346ms	QPH insert : 4ms AVL insert : 6ms	QPH insert : 4ms AVL insert : 5ms  ***** Search ***** BST Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert : 39ms QPH insert : 33ms AVL insert : 41ms  ***** Search *** BST Search : 21ms QPH Search : 10ms  n = 1000000  ***** Insert : **** BST insert : 581ms QPH insert : 2212ms
Search :  Search :  SPH Search :  AVL Search :  AVL Search :  AVL Search :	5ms 4ms 1ms 5ms 19ms 19ms 40ms 20ms 7ms 9ms 592ms 2296ms 661ms	QPH insert: 3ms AVL insert: 5ms	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  ***** Insert : 31ms  QPH insert : 24ms  AVL insert : 39ms  ***** Search ***  BST Search : 28ms  QPH Search : 9ms  AVL Search : 10ms  n = 1000000  ***** Insert : 595ms  QPH insert : 2346ms  AVL insert : 697ms	QPH insert: 4ms AVL insert: 6ms  ***** Search: 4ms QPH Search: 1ms AVL Search: 5ms  n = 100000 ***** Insert: 30ms QPH insert: 20ms AVL insert: 39ms AVL insert: 39ms  ***** Search BST Search: 24ms QPH Search: 7ms AVL Search: 9ms  n = 1000000 ***** Insert  ***** Search BST insert: 585ms QPH insert: 2302ms AVL insert: 662ms	QPH insert : 4ms AVL insert : 5ms  ****** Search ****** BST Search : 3ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert **** BST insert : 23ms AVL insert : 41ms  AVL insert : 41ms  ***** Search **** BST Search : 21ms QPH Search : 10ms AVL Search : 10ms AVL Search : 10ms AVL Search : 21ms AVL Search : 68ms AVL Search : 68ms AVL Search : 681ms AVL insert : 681ms AVL insert : 681ms AVL insert : 661ms
AVL insert :  """ Search :  AVL Search :	5ms 4ms 1ms 5ms 30ms 19ms 40ms 20ms 7ms 9ms 592ms 2290ms 661ms 93ms	QPH insert: 3ms AVL insert: 5ms  ***** Search: 5ms QPH Search: 2ms AVL Search: 4ms  n = 100000  ***** Insert: 33ms QPH insert: 33ms QPH insert: 38ms AVL insert: 10ms BST Search: 7ms AVL insert: 10ms  n = 1000000  ***** Insert: 590ms QPH insert: 2522ms AVL insert: 673ms  ****** Search	AVL insert : 5ms  ***** Search *****  BST Search : 4ms  QPH Search : 2ms  AVL Search : 5ms  n = 100000  **** Insert : 31ms  QPH insert : 24ms  AVL insert : 39ms  **** Search ***  BST Search : 28ms  QPH Search : 9ms  AVL Search : 10ms  n = 1000000  **** Insert : 595ms  QPH insert : 2346ms  AVL insert : 697ms  **** Search ****	QPH insert : 4ms AVL insert : 6ms  ***** Search : 4ms QPH Search : 1ms AVL Search : 5ms  n = 100000  **** Insert : 30ms QPH insert : 20ms AVL insert : 39ms  **** Search : 24ms QPH Search : 7ms AVL Search : 9ms  n = 1000000  **** Insert : 585ms QPH insert : 585ms QPH insert : 585ms AVL insert : 662ms  ***** Search *****	QPH insert : 4ms AVL insert : 5ms  ***** Search : 5ms QPH Search : 3ms AVL Search : 4ms  n = 100000  ***** Insert : 39ms QPH insert : 23ms AVL insert : 41ms  **** Search : 21ms QPH Search : 21ms QPH Search : 10ms  n = 1000000  **** Insert : 581ms QPH insert : 581ms QPH insert : 581ms QPH insert : 2212ms AVL insert : 681ms

#### 10번 실행하여 시간의 평균을 계산하였다.

Insert	n=10000	n=100000	n=1000000
BST	6.1	32.8	586.6
QPH	3.8	22.2	2256.3
AVL	5.2	39.8	673.4

Search	n=10000	n=100000	n=1000000
BST	4.4	22	95.3
QPH	1.9	7.3	19.8
AVL	4.3	9.5	61.3



n=10000 Insert (빠른순) : qph > avl > bst n=10000 Search (빠른순) : qph > avl > bst n=100000 Insert (빠른순) : qph > bst > avl n=100000 Search (빠른순) : qph > avl > bst n=1000000 Insert (빠른순) : bst > avl > qph n=1000000 Search (빠른순) : qph > avl > bst

삽입 시간의 경우 n=100000 까지는 QuadraticProbingHashTable이 다른 알고리즘보다 적은 시간이 걸리는걸 볼 수 있었지만, n이 1000000과 같이 큰 수가 되자 시간이 급증하였다. 탐색 시간의 경우 전체적으로 QuadraticProbingHashTable가 적은 시간에 수행되는 것을 볼 수 있다. n이 10000일 때는 BinarySearchTree와 AVLTree의 차이가 별로 없었지만 n이 커질수록 차이가 생겼다.

수가 커질 때, Insert에서는 QuadraticProbingHashTable이 성능이 안좋게 측정되었지만 Search를 할 때는 다른 알고리즘보다 월등히 빠르므로 탐색용도로 쓴다면 QuadraticProbingHashTable 알고리즘을 사용하는 것이 좋을 것 같다.