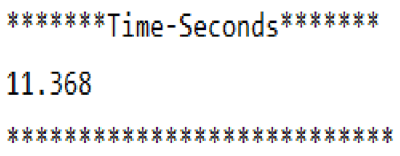
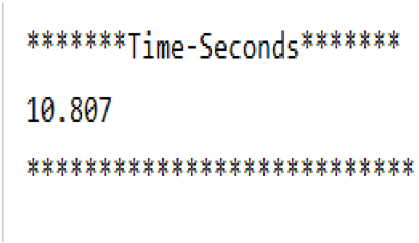
(Regular binary search tree)

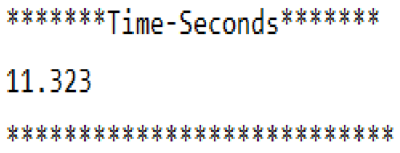
1) Elements added and deleted to BinarySearchTree(10000 Element).



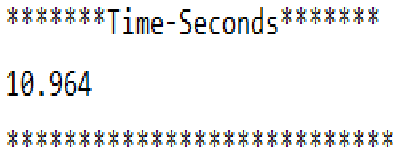
```
*******Time-Seconds*******
11.368
***************************
```



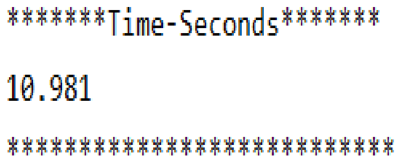
```
*******Time-Seconds*******
10.807
***************************
```



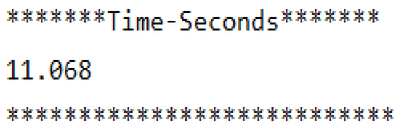
```
*******Time-Seconds*******
11.323
***************************
```



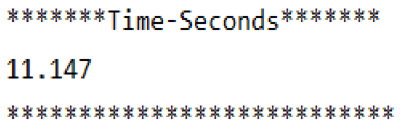
```
*******Time-Seconds*******
10.964
***************************
```



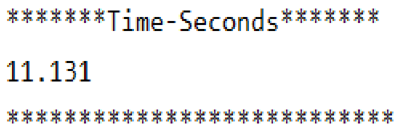
```
*******Time-Seconds*******
10.981
***************************
```



```
*******Time-Seconds*******
11.068
***************************
```



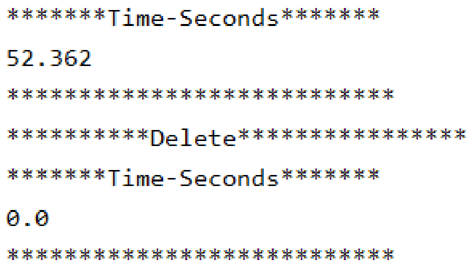
```
*******Time-Seconds*******
11.147
***************************
```



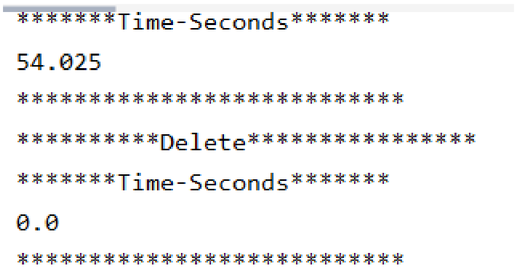
```
*******Time-Seconds*******
11.131
***************************
```

->Similar results were obtained in other running programs.

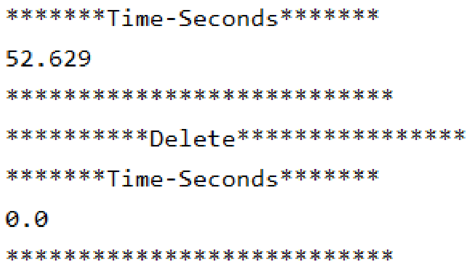
2) Elements added and deleted to BinarySearchTree(20000 Element).



```
*******Time-Seconds*******
52.362
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```



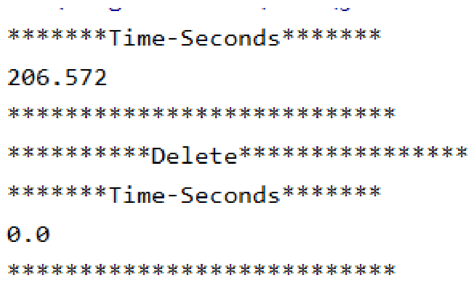
```
*******Time-Seconds*******
54.025
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```



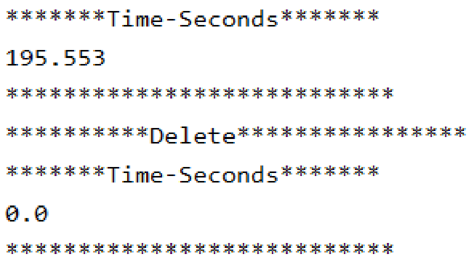
```
*******Time-Seconds*******
52.629
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

3) Elements added and deleted to BinarySearchTree(40000 Element).



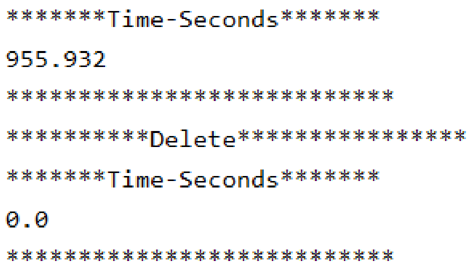
```
*******Time-Seconds*******
206.572
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```



```
*******Time-Seconds*******
195.553
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

4) Elements added and deleted to BinarySearchTree(80000 Element).

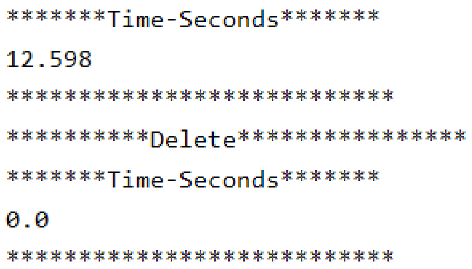


```
*******Time-Seconds*******
955.932
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

(Red-Black tree implementation in the book)

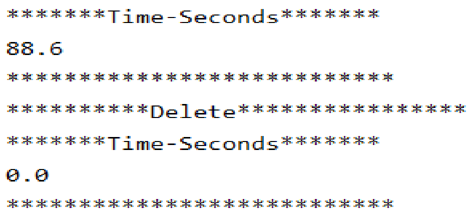
1) Elements added and deleted to RedBlackTree(10000).



```
*******Time-Seconds*******
12.598
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

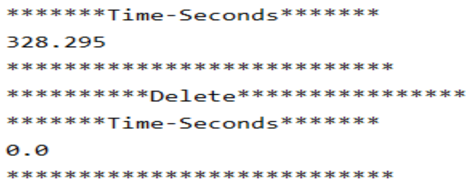
2) Elements added and deleted to RedBlackTree(20000).



```
*******Time-Seconds*******
88.6
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

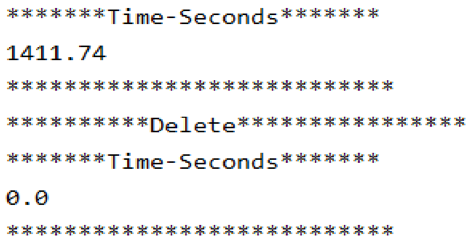
3)Elements added and deleted to RedBlackTree(40000).



```
*******Time-Seconds*******
328.295
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

5) Elements added and deleted to RedBlackTree(80000).

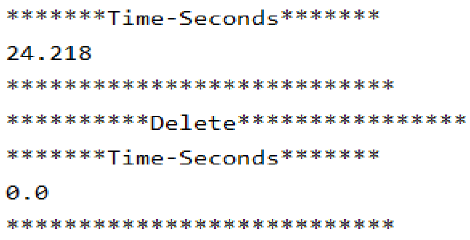


```
*******Time-Seconds*******
1411.74
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

(Red Black tree implementation in java)

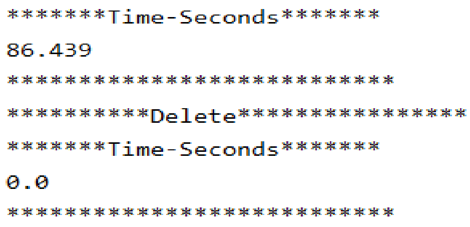
1) Elements added and deleted to RedBlackTree in Java(10000).



```
*******Time-Seconds*******
24.218
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

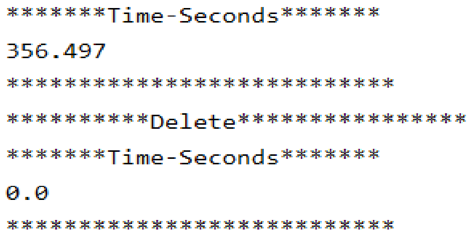
2) Elements added and deleted to RedBlackTree in Java(20000).



```
*******Time-Seconds*******
86.439
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

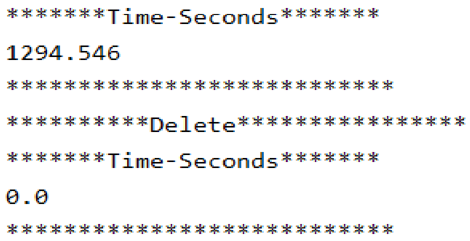
3) Elements added and deleted to RedBlackTree in Java(40000).



```
*******Time-Seconds*******
356.497
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

4) Elements added and deleted to RedBlackTree in Java(80000).

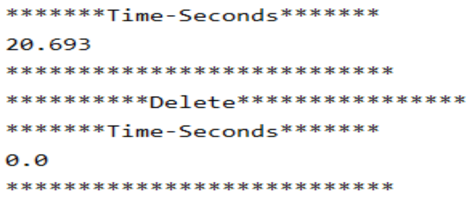


```
*******Time-Seconds*******
1294.546
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

(Skip list implementation in the book)

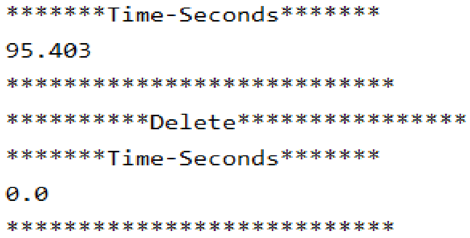
1) Elements added and deleted to SkipList(10000).



```
*******Time-Seconds*******
20.693
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

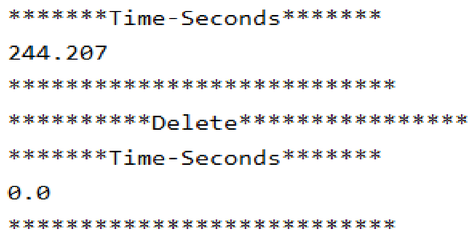
2) Elements added and deleted to SkipList(20000).



```
*******Time-Seconds*******
95.403
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

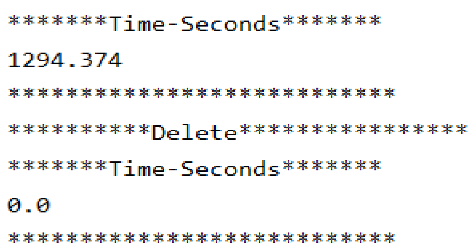
->Similar results were obtained in other running programs.

3) Elements added and deleted to SkipList(40000).



```
*******Time-Seconds*******
244.207
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

4) Elements added and deleted to SkipList(80000).

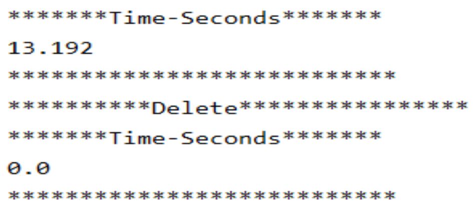


```
*******Time-Seconds*******
1294.374
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

(Skip list implementation in java)

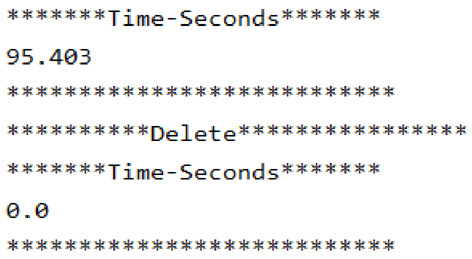
1) Elements added and deleted to SkipList in Java(10000).



```
*******Time-Seconds*******
13.192
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

->Similar results were obtained in other running programs.

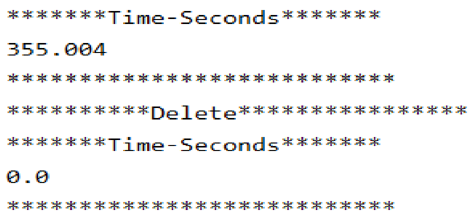
2)Elements added and deleted to SkipList in Java(20000).



```
*******Time-Seconds*******
95.403
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

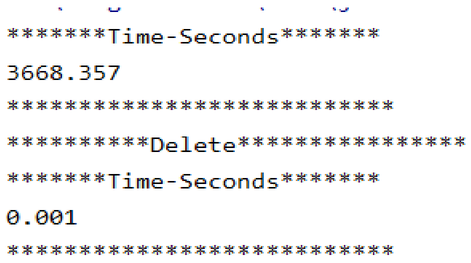
->Similar results were obtained in other running programs.

3)Elements added and deleted to SkipList in Java(40000).



```
*******Time-Seconds*******
355.004
***************************
**********Delete****************
*******Time-Seconds*******
0.0
***************************
```

4)Elements added and deleted to SkipList in Java(80000).

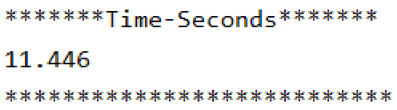


```
*******Time-Seconds*******
3668.357
***************************
**********Delete****************
*******Time-Seconds*******
0.001
***************************
```

->Similar results were obtained in other running programs.

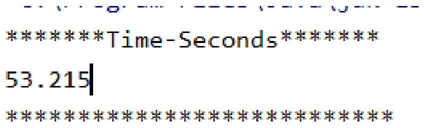
(B-tree implementation in the book)

1) Elements added to B-Tree(10000).



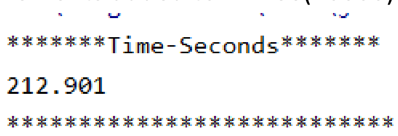
```
*******Time-Seconds*******
11.446
***************************
```

2) Elements added to B-Tree(20000).



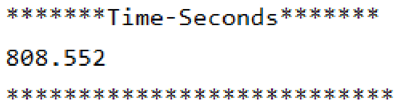
```
*******Time-Seconds*******
53.215
***************************
```

3) Elements added to B-Tree(40000).



```
*******Time-Seconds*******
212.901
***************************
```

4) Elements added to B-Tree(80000).



```
*******Time-Seconds*******
808.552
***************************
```