

자료구조

10

과목명 : 자료구조

분반 : 06

담당교수 : 박정희

학번 : 202002546

이름 : 임우진

목차

- 1. 요구사항**
- 2. 메소드**
- 3. 결과 캡처**

1. 요구사항

트리를 입력 받아 이진 탐색 트리 자료구조에 삽입하고
k번째 작은 수를 삭제하여 트리를 중위 순회하며
출력하는 문제이다.

필요 클래스 :

MainJava.java

Node.java

BST.java

2. 메소드

강의자료 5.1를 참고하셨습니다. (put,min,delete,deleteMin)

put()메소드와 delete_kthSmallest()메서드를 구현해야 한다.

put():

```
34      public void put(Key k, Value v){root = put(root,k,v);}
      3개 사용 위치
35      public Node put(Node n, Key k, Value v) {
36          if(n==null) return new Node(k,v);
37          int t = n.getKey().compareTo(k);
38          if(t>0) n.setLeft(put(n.getLeft(),k,v));
39          else if(t<0) n.setRight(put(n.getRight(),k,v));
40          else n.setValue(v);
41          return n;
42      }
```

delete_kthSmallest():

```
80      public void delete_kthSmallest(int k) {
81          if(root == null){
82              System.out.println("empty tree");
83          }
84          count = k;
85
86          delete_kthSmallest(root);
87
88          delete((Key)temp.getKey());
89      }
90
91      3개 사용 위치
92      public void delete_kthSmallest(Node delete){
93          if(delete != null){
94              delete_kthSmallest(delete.getLeft());
95              count--;
96              if(count == 0){
97                  temp = delete;
98              }
99              delete_kthSmallest(delete.getRight());
100          }
101      }
```

43	▼	<code>public Key min(){</code>
44		<code> if(root==null)return null;</code>
45		<code> return (Key)min(root).getKey();</code>
46		<code>}</code>
		3개 사용 위치
47	@ ▼	<code>private Node min(Node n){</code>
48		<code> if(n.getLeft()==null) return n;</code>
49	👍	<code> return min(n.getLeft());</code>
50		<code>}</code>

```

60 > public void delete(Key k) { root = delete(root,k); }
    3개 사용 위치
63 public Node delete(Node n, Key k){
64     if(n==null) return null;
65     int t = n.getKey().compareTo(k);
66     if(t>0){
67         n.setLeft(delete(n.getLeft(),k));
68     }else if(t<0){
69         n.setRight(delete(n.getRight(),k));
70     }else{
71         if(n.getRight()==null) return n.getLeft();
72         if(n.getLeft()==null) return n.getRight();
73         Node target = n;
74         n = min(target.getRight());
75         n.setRight(deleteMin(target.getRight()));
76         n.setLeft(target.getLeft());
77     }
78     return n;
79 }

```

```

51 public void deleteMin(){
52     if(root==null) System.out.println("empty tree");
53     root = deleteMin(root);
54 }
    3개 사용 위치
55 @ public Node deleteMin(Node n){
56     if(n.getLeft()==null) return n.getRight();
57     n.setLeft(deleteMin(n.getLeft()));
58     return n;
59 }

```

3. 결과 캡처

```
C:\Users\8146q\.jdk\openjdk-19.0.2\bin\java.exe "  
0 1 2 3 4 5 7 8 9 10  
1 2 3 4 5 7 8 9 10  
= = = = = = =  
1 2 3 4 5 6 7 8 9 10  
1 2 3 4 6 7 8 9 10  
= = = = = = =
```