6.5

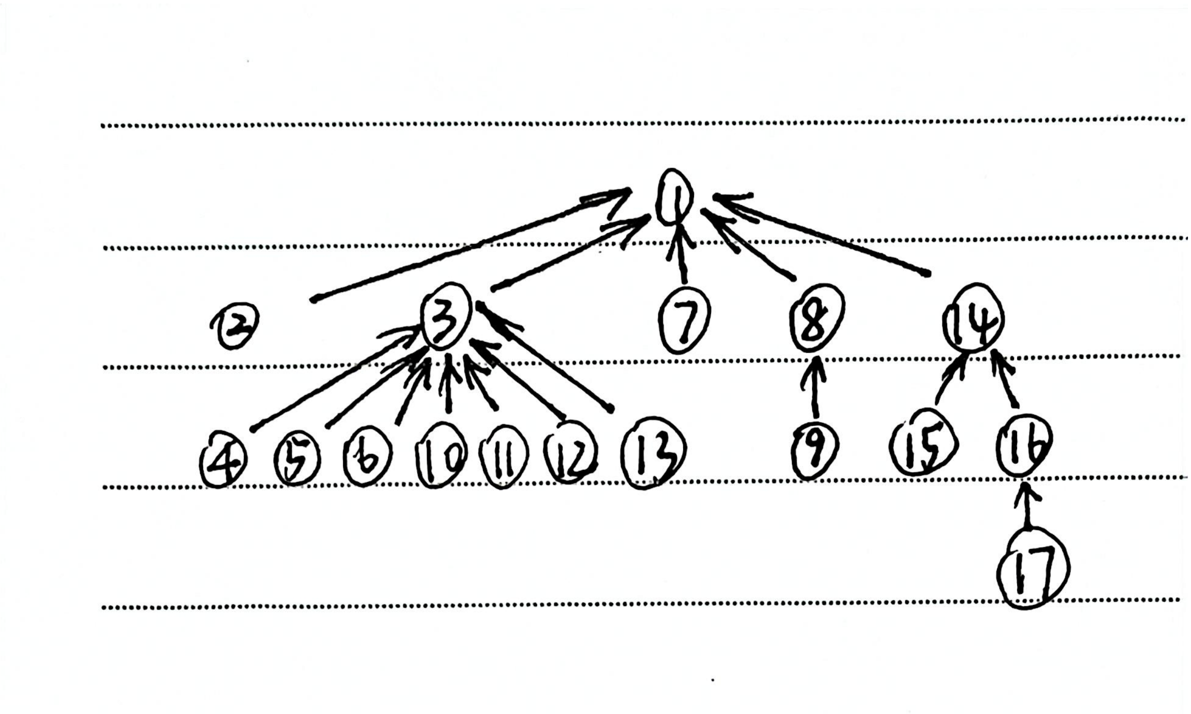
```
template <class T>
ostream& operator<<(ostream& out, const Set<T>& s) {
    traverse(s, out);
    return out;
}

template <class T>
void traverse(const Set<T>& s, ostream& out) {
    if (!s.Empty()) {
        if (s.Elemtype == 0) {
            out << s.GetName() << "{";
        }
        else if (s.Elemtype == 1) {
            out << s.GetData();
            if (s.GetNext() != NULL) {
                out << ",";
            }
        }
        else {
            traverse(s.GetSubSet(), out);
            if (s.GetNext() != NULL) {
                out << ",";
            }
        }
        traverse(s.GetNext(), out);
    }
    else {
        out << "}";
    }
}
```

如果集合中包含有子集合，各个子集合之间没有重复的元素，采用广义表结构比较合适。

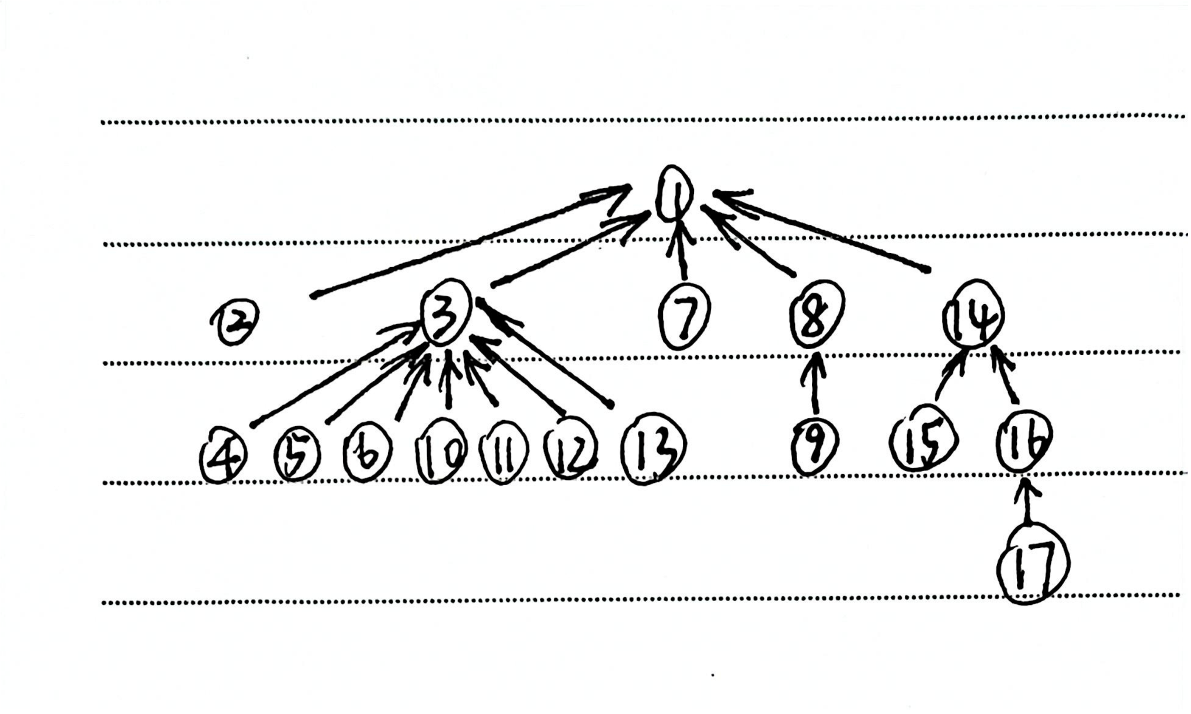
6.6

(1)



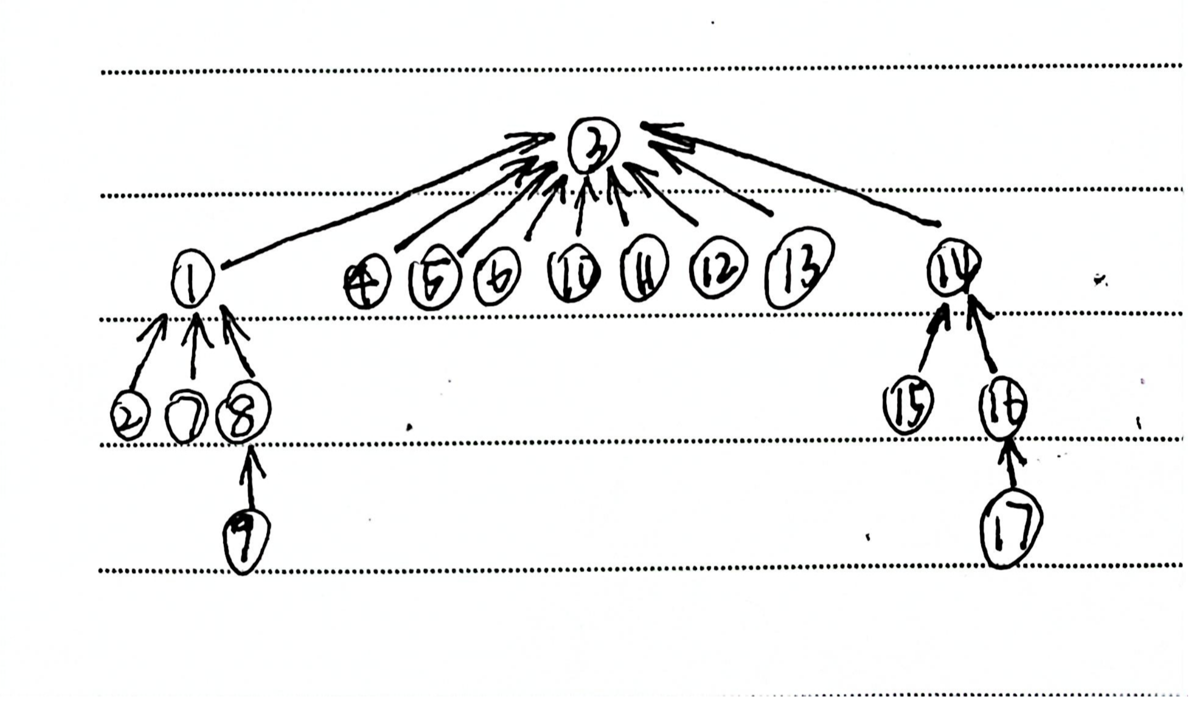
1
2
3
7
8
14
4
5
6
10
11
12
13
9
15
16
17

(2)



1
2
3
7
8
14
4
5
6
10
11
12
13
9
15
16
17

(3)



6.9

(1)

0	1	2	3	4	5	6	7	8	9	10	11	12
78		15	03		57	45	20	31		23	36	12
1		1	1		1	1	1	4		1	2	1

$$AVL_{succ} = \frac{1+1+1+1+1+1+4+1+2+1}{10} = \frac{7}{5}$$

$$AVL_{unsucc} = \frac{2+1+3+2+1+5+4+3+2+1+5+4+3}{13} = \frac{36}{13}$$

(2)

0	1	2	3	4	5	6	7	8	9	10	11	12
78		15	03		57	45	20	31	36	23		12
1		1	1		1	1	1	3	5	1		1

$$AVL_{succ} = \frac{1+1+1+1+1+1+3+5+1+1}{10} = \frac{8}{5}$$