```
1 from queuelinked import LinkedQueue
 2
 3 class BinaryTree:
 4
       class _Node:
           __slots__ = '_element', '_left', '_right'
 5
 6
 7
           def __init__(self,element, left=None, right=None):
8
               self._element = element
9
               self._left = left
10
               self._right = right
11
       def __init__(self):
12
13
           self._root = None
           self._size = 0
14
15
       def maketree(self, e, left, right):
16
           self._root = self._Node(e,left._root,right._root)
17
           left._root = None
18
           right._root = None
19
20
       def levelorder(self):
21
           Q = LinkedQueue()
22
           t = self._root
23
24
           print(t._element,end='--')
           Q.enqueue(t)
25
26
           while not Q.is_empty():
27
               t = Q.dequeue()
28
29
               if t._left:
                    print(t._left._element, end='--')
30
31
                    Q.enqueue(t._left)
               if t._right:
32
                    print(t._right._element, end='--')
33
                    Q.enqueue(t._right)
34
35
36
       def inorder(self, troot):
37
           if troot:
               self.inorder(troot._left)
38
               print(troot._element, end='--')
39
               self.inorder(troot._right)
40
41
       def preorder(self, troot):
42
43
           if troot:
44
               print(troot._element,end='--')
               self.preorder(troot._left)
45
46
               self.preorder(troot._right)
47
```

```
def postorder(self, troot):
49
           if troot:
               self.postorder(troot._left)
50
51
               self.postorder(troot._right)
               print(troot._element, end='--')
52
53
54 a = BinaryTree()
55 x = BinaryTree()
56 y = BinaryTree()
57 z = BinaryTree()
58 r = BinaryTree()
59 s = BinaryTree()
60 t = BinaryTree()
61
62 x.maketree(40,a,a)
63 y.maketree(60,a,a)
64 z.maketree(20, x, a)
65 r.maketree(50,a,y)
66 s.maketree(30, r, a)
67 t.maketree(10,z,s)
68
69 t.levelorder()
70 print()
71 t.preorder(t._root)
72 print()
73 t.inorder(t._root)
74 print()
75 t.postorder(t._root)
76 print()
77
78
```