Algorithm 1 BFS(G,s). This algorithm can be used to perform a level-order tree traversal. Basically, change the condition “for each v adjacent to t” in line 10 to “for each child v of t”. Each node can be printed out at the time of exiting the queue at line 9.

**Input:** A graph G, and a starting source s

**Output:** none

```
1: for each node do
2:   node.color ← white
3:   node.distance ← infinity
4: end for {color and distance are not needed in the level-order tree traversal. Similar for line 11-14 and 18}
5: s.distance ← 0 {not needed in the level-order tree traversal}
6: Q ← empty
7: Q.enqueue(s)
8: while Q is not empty do
9:   t ← Q.dequeue()
10: for each v adjacent to t do
11:   if v.color = white then
12:     v.color ← gray
13:     v.distance ← t.distance + 1
14:     v.parent ← t
15:     Q.enqueue(v)
16:   end if
17: end for
18: t.color ← black
19: end while
```