

# Breitensuche Algorithmus

1. Bestimme den Knoten, an dem die Suche beginnen soll, und speichere ihn in einer Warteschlange ab.
2. Entnimm einen Knoten vom Beginn der Warteschlange und markiere ihn.
  - Falls das gesuchte Element gefunden wurde, brich die Suche ab und liefere "gefunden" zurück.
  - Anderenfalls hänge alle bisher unmarkierten Nachfolger dieses Knotens ans Ende der Warteschlange an.
3. Wenn die Warteschlange leer ist, dann wurde jeder Knoten bereits untersucht. Beende die Suche und liefere "nicht gefunden" zurück.
4. Wiederhole ab Schritt 2.

== Beispiel ==

## Quellcode

```
1. void breitensuche(Graph g, String node) {
2. Map<String, Integer> abstand = new HashMap<String, Integer>();
3. Queue<String> queue = new LinkedList<String>();
4. abstand.put(node, 0);
5. queue.add(node);
6. while (!queue.isEmpty()) {
7. String s = queue.poll();
8. int d = abstand.get(s); // muss enthalten sein
9. for (Iterator<String> it = g.edgeIterator(s); it.hasNext();) {
10. String expanded = it.next();
11. if (!abstand.containsKey(expanded)) {
12. abstand.put(expanded, d+1);
13. queue.add(expanded);
14. }
15. }
16. action(s); // zum Beispiel Ausgabe
17. }
18. }
```

Alles anzeigen