

# Aufgabe zur Komplexität

(Klasse-QueueElement)

**Stichwörter:** Algorithmische Komplexität (O-Notation)

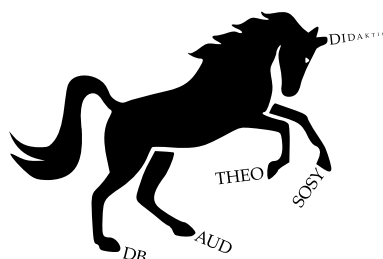
## Aufgabe zur Komplexität

Der Konstruktor `QueueElement(...)` und die Methode `setNext(...)` sowie `getNext(...)` haben  $\mathcal{O}(1)$ . Geben Sie die Zeitkomplexität der Methode `append(int content)` an, die einer Schlange ein neues Element anhängt.

```
public void append(int contents) {
    QueueElement newElement = new QueueElement(contents) ;
    if (first == 0) {
        first = newElement;
        last = newElement;
    } else {
        // Ein neues Element hinten anhängen.
        last.setNext(newElement);
        // Das angehängte Element als Letztes setzen.
        last = last.getNext();
    }
}
```

Lösungsvorschlag

Das Anhängen eines neuen Elements in die gegebene Warteschlange hat die konstanten Rechenzeitbedarf von  $\mathcal{O}(1)$ , egal wie lange die Schlange ist, da wir das letzte Element direkt ansprechen können.



### Die Bschlangaul-Sammlung

Hermine Bschlangauland Friends

Eine freie Aufgabensammlung mit Lösungen von Studierenden für Studierende zur Vorbereitung auf die 1. Staatsexamensprüfungen des Lehramts Informatik in Bayern.



Diese Materialsammlung unterliegt den Bestimmungen der Creative Commons Namensnennung-Nicht kommerziell-Share Alike 4.0 International-Lizenz.

Hilf mit! Die Hermine schafft das nicht allein! Das ist ein Community-Projekt! Verbesserungsvorschläge, Fehlerkorrekturen, weitere Lösungen sind herzlich willkommen - egal wie - per Pull-Request oder per E-Mail an [hermine.bschlangaul@gmx.net](mailto:hermine.bschlangaul@gmx.net). Der TeX-Quelltext dieses Dokuments kann unter folgender URL aufgerufen werden: [https://github.com/bschlangaul-sammlung/examens-aufgaben/blob/main/Module/30\\_AUD/50\\_Algorithmische-Komplexitaet/Aufgabe\\_Klasse-QueueElement.tex](https://github.com/bschlangaul-sammlung/examens-aufgaben/blob/main/Module/30_AUD/50_Algorithmische-Komplexitaet/Aufgabe_Klasse-QueueElement.tex)