

Komplexität

(Methode „magicStaff()“)

Stichwörter: Algorithmische Komplexität (O-Notation)

Welche Komplexität hat das Programmfragment?

```
public void magicStaff(int[] array) {
    for (int i = 0; i < array.length; i++) {
        int counter = 0;
        if (array[i] % 3 == 0) {
            break;
        }
        do {
            if (array[i] % 2 == 0) {
                array[i] += array[counter];
            }
        } while (counter++ < array.length);
    }
}
```

Code-Beispiel auf Github ansehen: [src/main/java/org/bschlangaul/aufgaben/aud/komplexitaet/Komplexitaet.java](https://github.com/bschlangaul/aufgaben/aud/komplexitaet/Komplexitaet.java)

Bestimmen Sie in Abhängigkeit von n die Komplexität des Programmabschnitts im

(a) Best-Case.

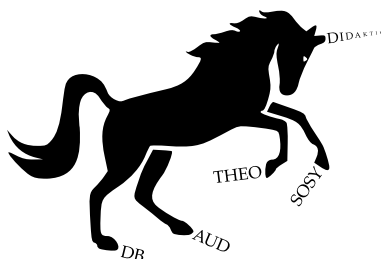
Lösungsvorschlag

$O(1)$: Wenn die erste Zahl im Feld `array` ohne Rest durch 3 teilbar ist, wird sofort aus der for-Schleife ausgestiegen (wegen der `break` Anweisung).

(b) Worst-Case.

Lösungsvorschlag

$O(n^2)$: Wenn keine Zahl aus `array` ohne Rest durch 3 teilbar ist, werden zwei Schleifen (`for` und `do while`) über die Anzahl n der Elemente des Felds durchlaufen.



Die Bschlangaul-Sammlung
Hermine Bschlangaul and 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. Der TeX-Quelltext dieser Aufgabe kann unter folgender URL aufgerufen werden: https://github.com/bschlangaul-sammlung/examens-aufgaben-tex/blob/main/Module/30_AUD/50_Algorithmische-Komplexitaet/Aufgabe_Methode-magicStaff.tex