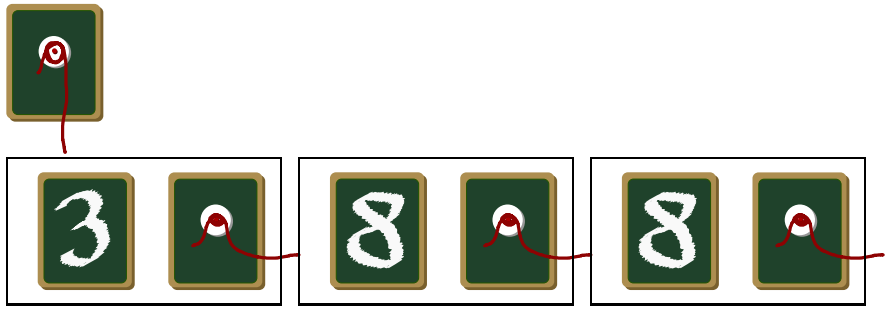
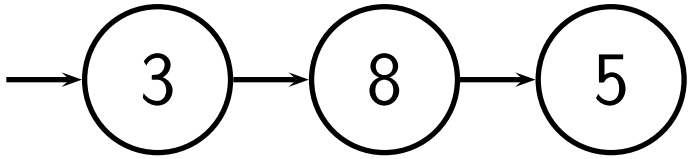
Tagesprogramm

Implementierung einfacher rekursiver Datenstrukturen

Stack als Alternative zur Rekursion

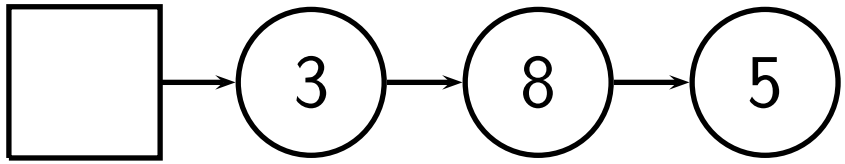
Verkettete Liste



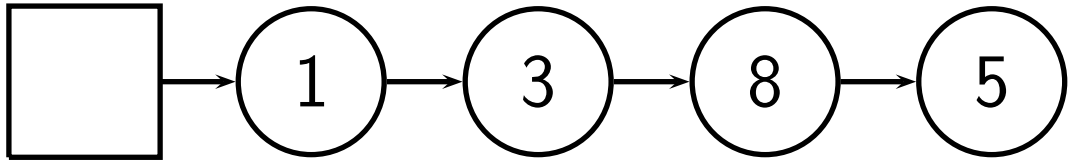


Verkettete Liste als Stack

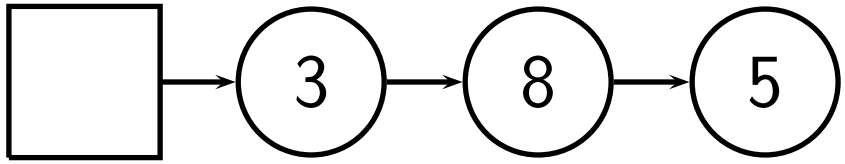
aktueller Zustand von `stack`:



nach `stack.push(1);`:

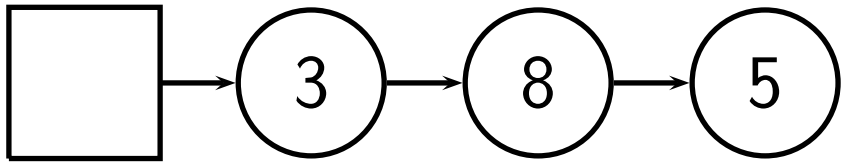


nach `stack.pop();`:

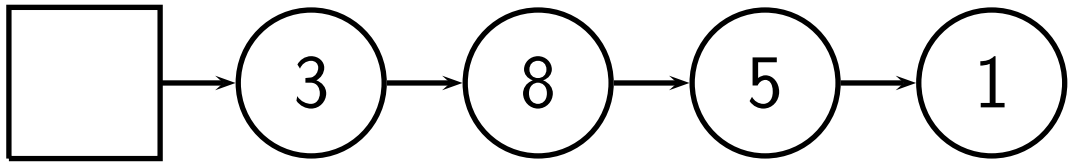


Verkettete Liste als FIFO-Queue

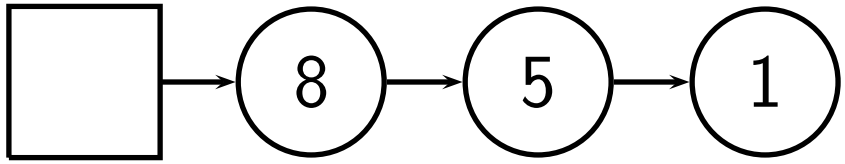
aktueller Zustand von `queue`:



nach `queue.add(1);`:



nach `queue.remove();`:



Aufgabe: Effizienz

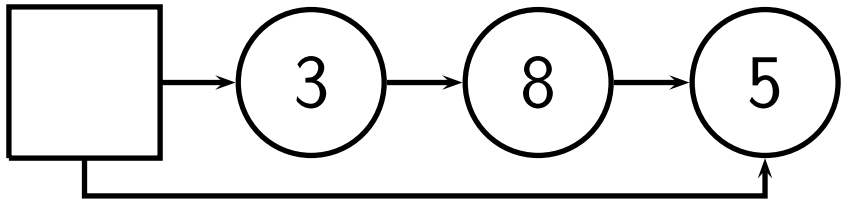
Finden Sie in Gruppen zu 2 bis 3 Personen eine Antwort auf folgende Frage:

Wie kann man vermeiden, dass `add` die gesamte Liste durchläuft?

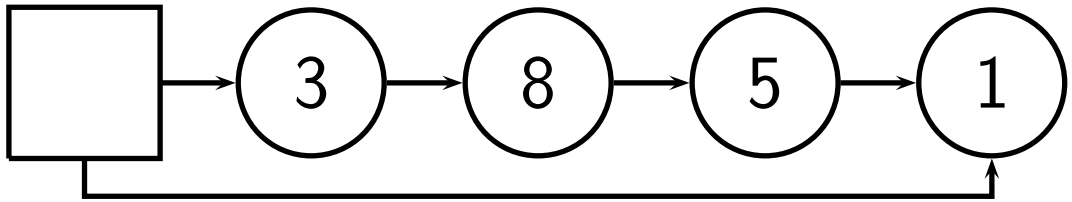
Zeit: 3 Minuten

Vereinfachter Zugriff als FIFO-Queue

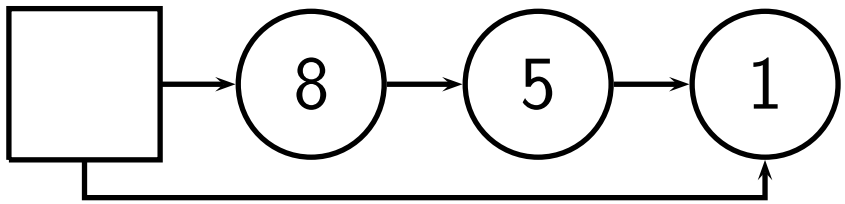
aktueller Zustand von `queue`:



nach `queue.add(1);`:

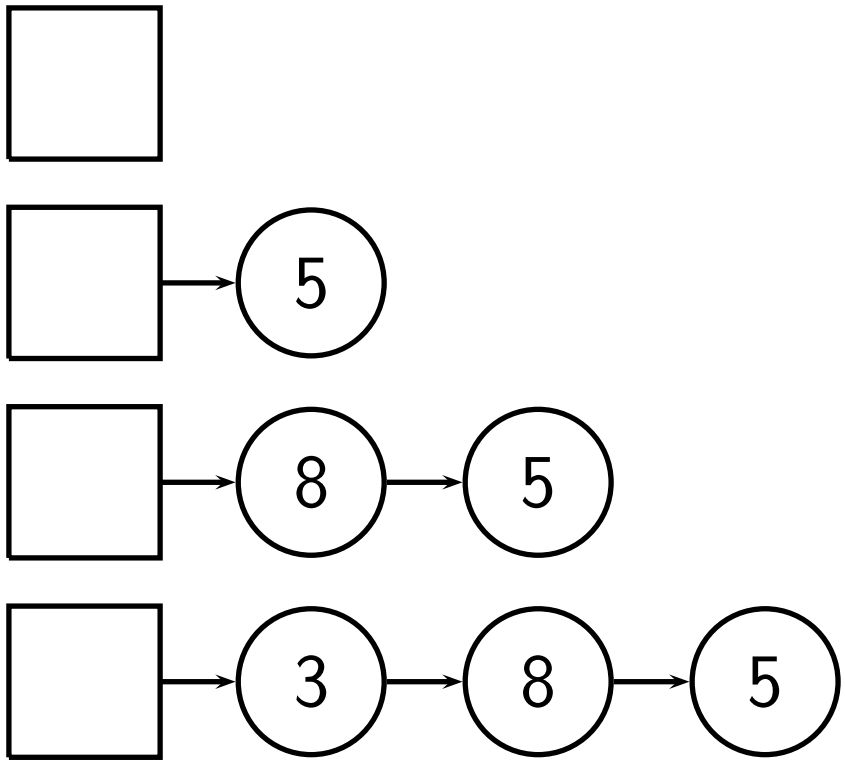


nach `queue.remove();`:



Fundiertheit

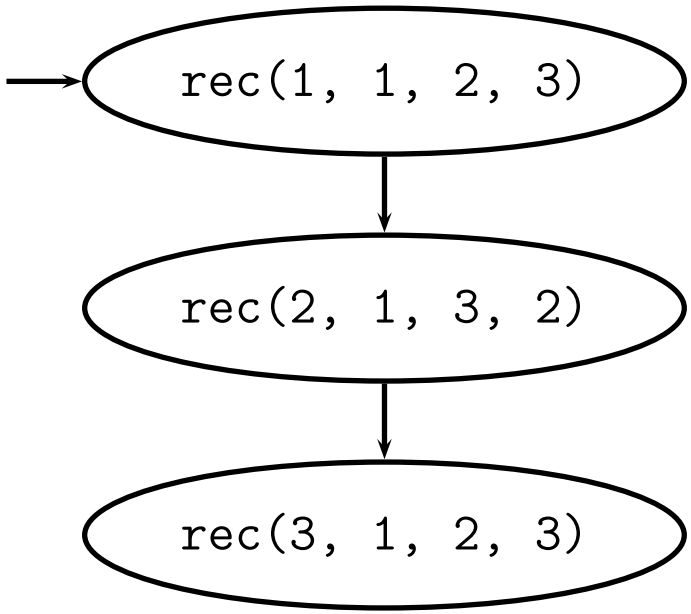
schrittweiser Aufbau:



Ende durch `null` gekennzeichnet

System-Stack

```
private static void rec(int i, int a, int b, int c) {
    if (i > 0) {
        rec(i-1, a, c, b);
        System.out.println("von " + a + " nach " + c);
        rec(i-1, b, a, c);
    }
}
```



Aufgabe: FIFO-Queue statt Stack

Versuchen Sie Folgendes:

Lösen Sie die Türme von Hanoi mit einer FIFO-Queue statt einem Stack.