

# Bachelor-/Masterarbeit

## Relaxierte parallele FIFO-Queue

### Beschreibung

FIFO-Queues sind als fundamentale Datenstruktur Baustein vieler Algorithmen wie Breitensuche in Graphen oder der Verwaltung von Ressourcen in Betriebssystemen. Daher spielen sie auch bei der Parallelisierung von Algorithmen eine wichtige Rolle. Parallele FIFO-Queues skalieren durch inhärente Zugriffskonflikte jedoch kaum und sind oftmals ein Bottleneck. Dieses Problem kann durch Lockerung (Relaxierung) der FIFO-Semantik gelöst werden. Die Idee hierbei ist, die Reihenfolge der Elemente in der Queue zu lockern, um Zugriffskonflikte zu reduzieren.

Anwendungen müssen in der Regel durch Relaxierung mehr Arbeit verrichten als mit korrekten FIFO-Queues. Idealerweise soll die Relaxierung dazu führen, dass die Anwendung durch bessere Skalierbarkeit dennoch schneller wird.

### Thema der Arbeit

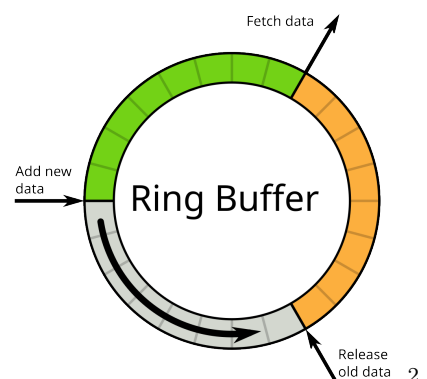
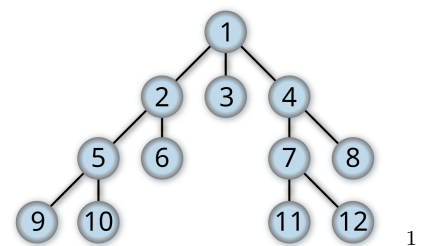
Die Aufgabe dieser Arbeit ist die Entwicklung und Implementierung einer skalierbaren, relaxierten FIFO-Queue. Im Rahmen dieser Arbeit soll ein Ringpuffer mit fester Größe als Grundlage dienen. Die Relaxierung kann erreicht werden, indem Elemente semantisch zu Blöcken zusammengefasst werden, die nur von einem Thread bearbeitet werden. Hier gilt es, ein Design zu entwickeln, welches den Grad der Relaxierung kontrollierbar macht und Gütequalitäten einhält.

Die FIFO-Queue soll in Anwendungen evaluiert werden und es soll herausgearbeitet werden, wie sich der Grad der Relaxierung auf die Performance der Anwendung auswirkt.

Bei Interesse kann auch ein Schwerpunkt auf die theoretische Analyse der Datenstruktur gelegt werden. Die Implementierung erfolgt bevorzugt in C++, kann bei geeigneten Vorkenntnissen aber auch in Rust erfolgen. Es existiert kein Framework, das für die Implementierung verwendet werden muss.

### Voraussetzungen

- Interesse an und solide Kenntnisse von Algorithmen und Datenstrukturen
- Programmierkenntnisse in C++
- Erfahrungen mit paralleler Programmierung sinnvoll



<sup>1</sup> [https://en.wikipedia.org/wiki/Breadth-first\\_search](https://en.wikipedia.org/wiki/Breadth-first_search)

<sup>2</sup> [https://upload.wikimedia.org/wikipedia/commons/d/d9/Ring\\_buffer.svg](https://upload.wikimedia.org/wikipedia/commons/d/d9/Ring_buffer.svg)