

Objektreferenz

- Ein Array mit Pointern (als Attribut der Klasse)
- Verweis auf Objekte (auf dem Heap)
- Umsortieren ist einfach, da nur Pointer kopiert werden.

Klasse DataModel

```
1 class DataModel
2 {
3 public:
4     DataModel();
5     static const unsigned short arraySizeEntry = 20;
6     void insertEntry(int pos, int val);
7     void removeEntry(int pos);
8     int findEntryPos(int pos);
9     int findEntryVal(int val);
10    bool appendEntryBack(int val);
11
12 private:
13    Entry* entryArray[arraySizeEntry]; // Links auf die Objekte
14                                // in der Warteschlange.
15 }
```

Ring-Puffer

- Array ist linear.
- Problem: Kontinuierliche Messwerte,
alte werden automatisch überschrieben.
- logisch: Ring
- zwei “Zeiger” (schreiben / lesen)

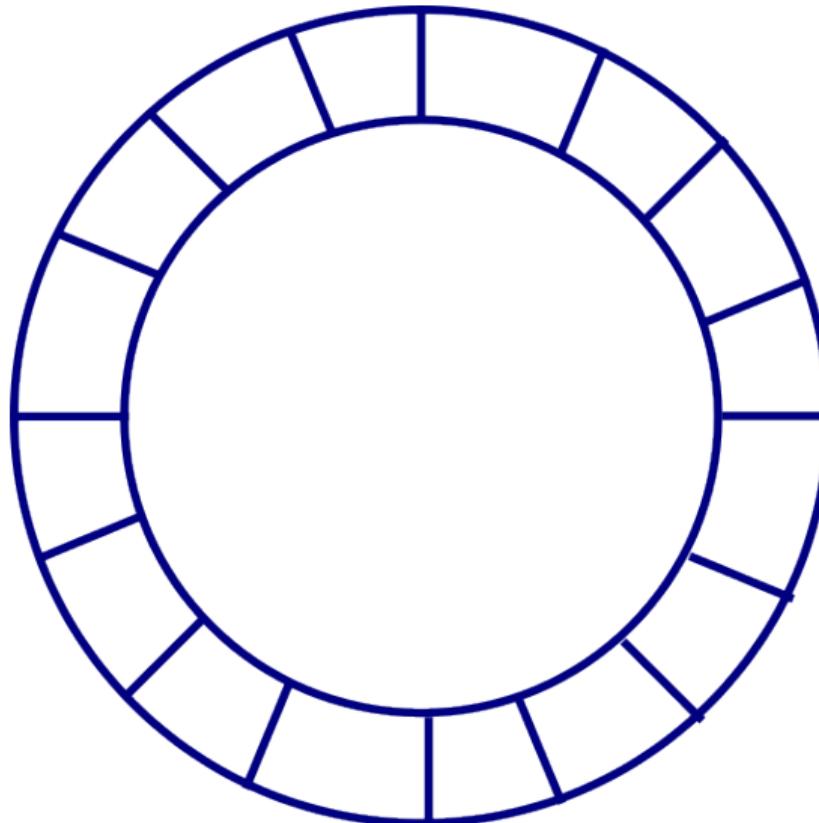
Ring-Puffer

- Array ist linear.
- Problem: Kontinuierliche Messwerte,
alte werden automatisch überschrieben.
- logisch: Ring
- zwei “Zeiger” (schreiben / lesen)
- reale Struktur: Array (in der Regel)

Realisierung

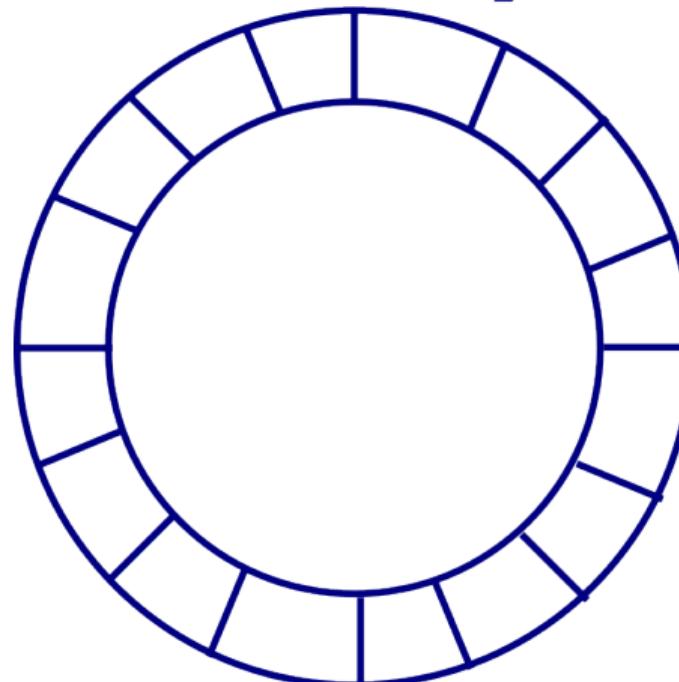
- Array für Werte (z.B. Array für Int)
- Zeiger lesen und Zeiger schreiben
- Lesen darf schreiben nicht überholen
- Wenn schreiben am Ende des Arrays ist,
Sprung auf Pos 0. Ggf. Lese-Zeiger verschieben.

Visualisierung



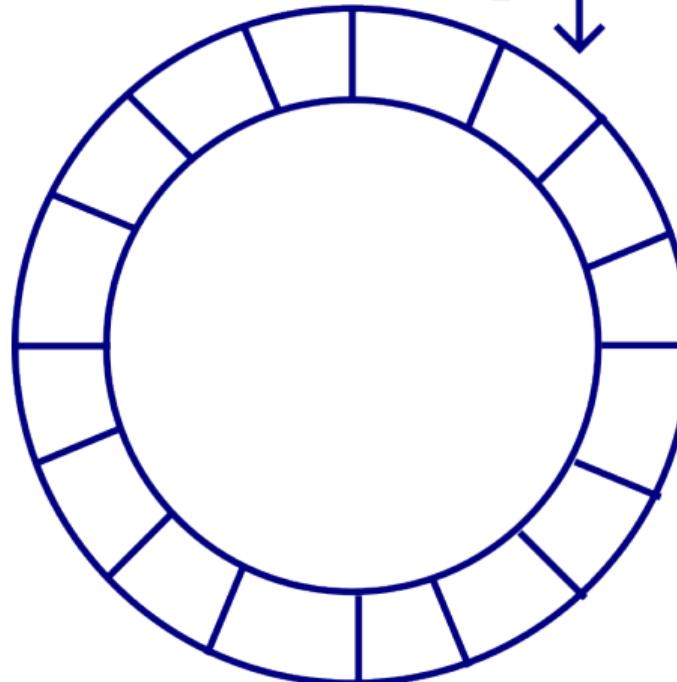
Visualisierung

↓ L, S

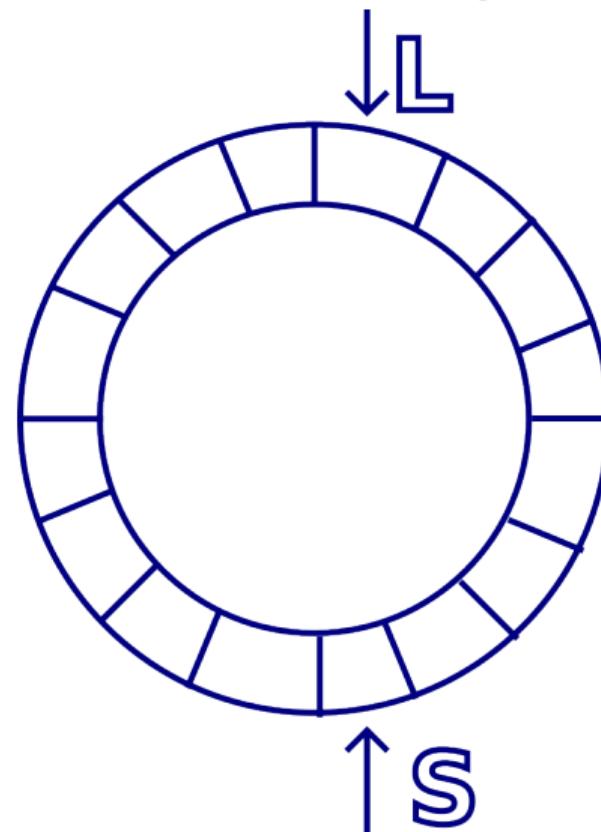


Visualisierung

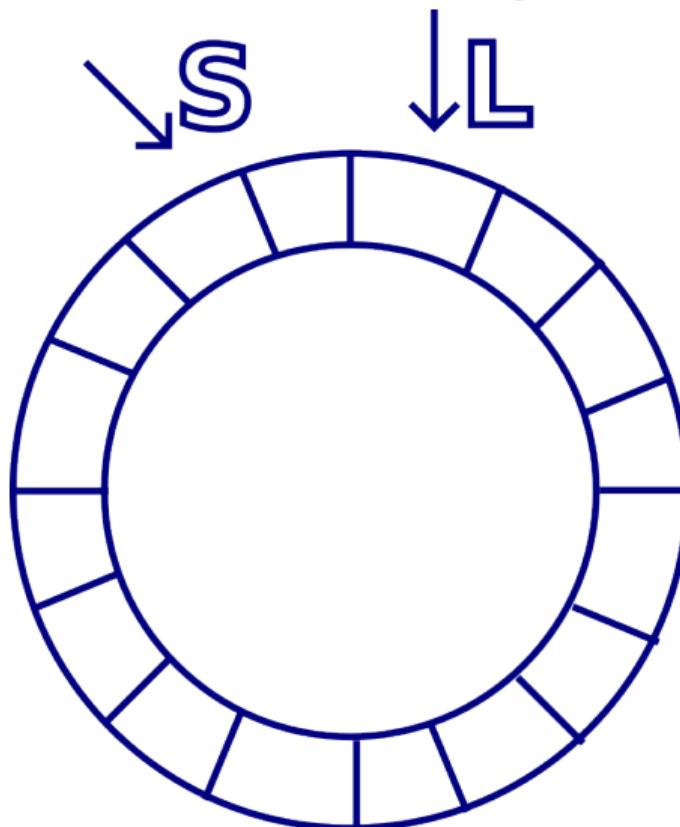
↓ L, S



Visualisierung

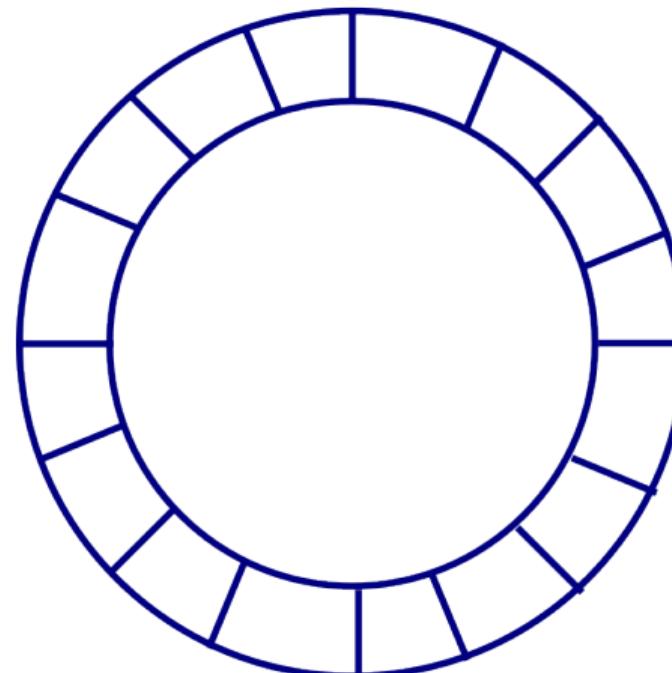


Visualisierung



Visualisierung

S ↓ L



Visualisierung

