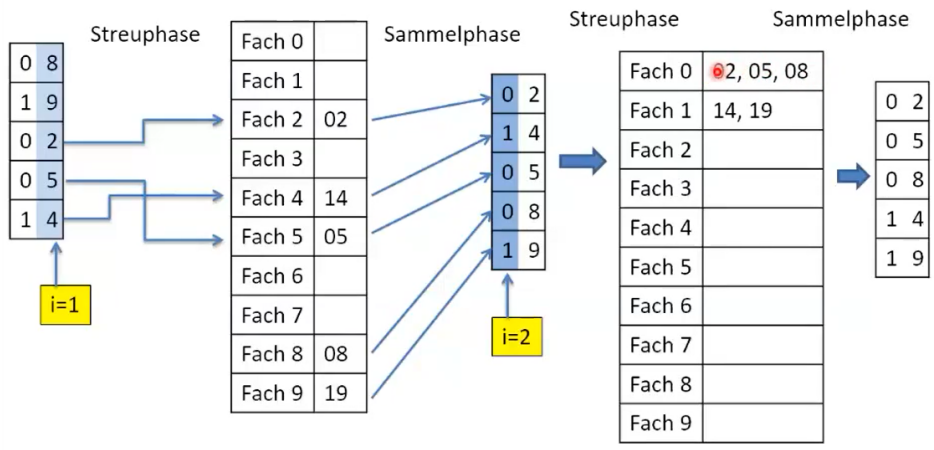
Eigenschaften

- nicht vergleichsbasiert
- Annahme
  - Anzahl der (Dezimal)Stellen konstant
  - kleine Anzahl an Stellen
- effizienter für große Datenmengen
- hoher Speicheraufwand
- womöglich hybrides Sortierverfahren
  - n groß => RadixSort
  - n klein => InsertionSort

Verfahren

- Streuphase
  - n aufteilen in Fächer/Buckets
  - 1 Fach für jeden möglichen Wert
    - 10 für dezimal
    - 2 für binär
- Sammelphase
  - Fächer zusammenfügen
  - sortiert nach i-ter Stelle
- wiederholen bis alle Stellen durchlaufen
- führende Nullen für Zahlen mit zu wenig Stellen
-



Sortieren von *n* Dezimalzahlen der Länge *d:*

```
RADIXSORT (A, d)
1: FOR i = 1 TO d
2:     Ordne A nach i-ter Ziffer v.h. in Fächer ein (Streuphase)
3:     Fasse die Fächer in aufsteigender Reihenfolge
           wieder in A zusammen (Sammelphase)
```

T(n) = O(d n) … linear, wenn d als konstant betrachtet wird!

- Nach den ersten k Durchläufen sind die Zahlen, eingeschränkt auf die letzten k Ziffern, sortiert
- **Wichtig**: Die vorige Reihenfolge innerhalb der Fächer muss aufrechterhalten werden