

Algorithmen und Datenstrukturen



TECHNISCHE
UNIVERSITÄT
DARMSTADT



SYSTEMS

Stefan Roth, SS 2025

02

Sortieren

Folien beruhen auf der Veranstaltung von Prof. Marc Fischlin und Christian Janson aus dem SS 2024

Korrektheitsbeweis Merge Sort

Alternative Betrachtung via
Induktionsschritt von $i \rightarrow i+1$

Korrektheit Merge (IIa)

Induktionsschritt $i \rightarrow i+1$

Invariante gilt nach Voraussetzung vor Iteration i

Iteration setzt $B[i]$ auf kleineren bzw. einzigen Wert $A[p], A[q]$

Nach Voraus. $B[i-1] \leq A[p], A[q]$, so dass $B[i-1] \leq B[i]$,
also ist $B[0..i]$ nach Iteration auch sortiert

$i = p - \text{left} + q - (\text{mid} + 1)$, $p \leq \text{mid} + 1$, $q \leq \text{right} + 1$,
 $B[0..i-1]$ ist sortiert und besteht aus $A[\text{left}..p-1]$, $A[\text{mid}+1..q-1]$.
Ferner: $B[i-1] \leq A[p], A[q]$

```
2  FOR i=0 TO right-left DO // merge all elements
3      IF q>right OR (p<=mid AND A[p]<=A[q]) THEN
4          B[i]=A[p];
5          p=p+1;
6      ELSE //next element at q
7          B[i]=A[q];
8          q=q+1;
9  FOR i=0 TO right-left DO A[i+left]=B[i]; //copy back
```

Korrektheit Merge (IVa)

Induktionsschritt $i \rightarrow i+1$

Invariante gilt nach Voraussetzung vor Iteration i

Iteration setzt $B[i]$ auf kleineren bzw. einzigen Wert $A[p], A[q]$

Da Teil-Arrays vorsortiert, gilt nach 4 bzw. 7 (vor Erhöhen von p bzw. q):
 $B[i] = \min\{A[p], A[q]\} \leq A[p], A[p+1], A[q], A[q+1]$

...

$i = p - \text{left} + q - (\text{mid} + 1)$, $p \leq \text{mid} + 1$, $q \leq \text{right} + 1$,
 $B[0 \dots i-1]$ ist sortiert und besteht aus $A[\text{left} \dots p-1]$, $A[\text{mid} + 1 \dots q-1]$.
Ferner: $B[i-1] \leq A[p], A[q]$

```
2  FOR i=0 TO right-left DO // merge all elements
3      IF q>right OR (p<=mid AND A[p]<=A[q]) THEN
4          B[i]=A[p];
5          p=p+1;
6      ELSE //next element at q
7          B[i]=A[q];
8          q=q+1;
9  FOR i=0 TO right-left DO A[i+left]=B[i]; //copy back
```

Korrektheit Merge (Va)

Induktionsschritt $i \rightarrow i+1$

Invariante gilt nach Voraussetzung vor Iteration i

Zähler des kopierten Werts wird erhöht,
also $B[0..i]$ wegen Voraussetzung aus angegebener Menge

...

$i = p - \text{left} + q - (\text{mid} + 1)$, $p \leq \text{mid} + 1$, $q \leq \text{right} + 1$,
 $B[0..i-1]$ ist sortiert und besteht aus $A[\text{left}..p-1]$, $A[\text{mid}+1..q-1]$.
Ferner: $B[i-1] \leq A[p]$, $A[q]$

```
2  FOR i=0 TO right-left DO // merge all elements
3      IF q>right OR (p<=mid AND A[p]<=A[q]) THEN
4          B[i]=A[p];
5          p=p+1;
6      ELSE //next element at q
7          B[i]=A[q];
8          q=q+1;
9  FOR i=0 TO right-left DO A[i+left]=B[i]; //copy back
```

Korrektheit Merge (Vla)

Induktionsschritt $i \rightarrow i+1$

Invariante gilt nach Voraussetzung vor Iteration i

Zähler i wird um eins erhöht, und entweder p oder q auch um eins

Wenn $p \geq \text{mid}+1$ bzw. $q \geq \text{right}+1$ wird die Teilliste nicht mehr gewählt, also Zählerwerte nicht mehr erhöht

$i = p - \text{left} + q - (\text{mid} + 1)$, $p \leq \text{mid} + 1$, $q \leq \text{right} + 1$,
 $B[0 \dots i-1]$ ist sortiert und besteht aus $A[\text{left} \dots p-1]$, $A[\text{mid}+1 \dots q-1]$.
Ferner: $B[i-1] \leq A[p]$, $A[q]$

```
2  FOR i=0 TO right-left DO // merge all elements
3      IF q>right OR (p<=mid AND A[p]<=A[q]) THEN
4          B[i]=A[p];
5          p=p+1;
6      ELSE //next element at q
7          B[i]=A[q];
8          q=q+1;
9  FOR i=0 TO right-left DO A[i+left]=B[i]; //copy back
```