Beweis von Satz 5.7:

Zunächst runden wir n auf die nächste Zweierpotenz
(In O-Notation ändert sich dadurch nichts, $2^{k-1} \leq n \leq 2^k$,
also ändert sich n maximal um einen Faktor 2.)
Wir erhalten also jeweils zwei Subarrays der Größe $\frac{n}{2}$.

Jetzt betrachten wir $T(n)$, die Laufzeit von Mergesort
für einen n-elementigen Array A.

Dann haben wir:

Divide: $O(1)$
Conquer: $2 \cdot T(\frac{n}{2})$
Combine: $O(n)$

Damit ergibt sich folgende Rekursionsbeziehung:

$$T(n) = \begin{cases} O(1), & n = 1 \\ O(1) + 2 \cdot T(\frac{n}{2}) + O(n) = 2T(\frac{n}{2}) + O(n) & \text{für } n \geq 2 \end{cases}$$

Das ist eine „Rekursionsgleichung". (Mehr dazu später!)

Für geeignete Konstanten c, d können wir auch schreiben:

$$T(n) = 2T\left(\frac{n}{2}\right) + c \cdot n \quad , \quad n \geq 2 \quad , \quad c \in \mathbb{R}$$
$$T(1) = O(1) = d$$

Nun müssen wir zeigen, dass $T(n) \leq g \cdot n \log n$
für ein geeignetes g.

Das zeigen wir per Induktion!

Induktionsanfang:

$n=2$: $\quad T(2) = 2 \cdot T(1) + c \cdot 2 = 2 \cdot d + 2 \cdot c = 2(d+c)$

Außerdem ist $\quad g \cdot \underbrace{n}_{2} \underbrace{\log n}_{1} = 2 \cdot g \overset{!}{\geq} 2 \cdot (d+c)$.

wenn wir $g \geq (d+c)$ wählen (was $g \geq c$ impliziert),
dann ist das erfüllt.

Induktionsannahme:

Die Behauptung gelte für $\frac{n}{2} = 2^{k-1}$, d.h.

$$T\left(\frac{n}{2}\right) \leq g \cdot \frac{n}{2} \cdot \log\left(\frac{n}{2}\right)$$

Induktionsschritt: $\quad k-1 \to k$, d.h. von $\frac{n}{2} \to n$:

$$\begin{aligned}
T(n) &= 2 \cdot T\left(\frac{n}{2}\right) + c \cdot n \\
&\leq 2 \cdot \left(g \cdot \frac{n}{2} \cdot \log\left(\frac{n}{2}\right)\right) + c \cdot n \\
&= g \cdot n \cdot \log\left(\frac{n}{2}\right) + c \cdot n \\
&= g \cdot n \log n \underbrace{- g \cdot n + c \cdot n}_{\leq 0,\ \text{da } g \geq c} \\
&\leq g \cdot n \log n .
\end{aligned}$$

Also ist $T(n) \in O(n \log n)$, wie behauptet.

$\square$