Von Induktion zu «Teile und Herrsche»

Daniel Oehry, April 2022

Inhalt und Lernziele

Die vorliegende Unterrichtseinheit behandelt folgende Punkte.

- Das mathematische Beweisverfahren der vollständigen Induktion wird anhand eines Beispiels vorgestellt. Es geht dabei vor allem darum, die Begriffe Induktionsschritt und Induktionsanfang zu klären. Anhand von zwei Aufgaben wird das Verfahren danach eingeübt.
- In einem zweiten Schritt wird gezeigt, wie die Idee der Induktion zur Entwicklung von Algorithmen verwendet werden kann. Insbesondere wird gezeigt, wie mit Hilfe von Rekursionsgleichungen eine Aussage über die Komplexität eines Algorithmus gewonnen werden kann.
- «Teile und Herrsche» wird anhand eines Beispiels als eine Verallgemeinerung der Induktion vorgestellt.
- Mit Hilfe von Aufgaben werden verschiedene Typen von Rekursionsgleichungen vorgestellt.

Vorwissen

Für die Bearbeitung der Aufgaben ist ein sicherer Umgang mit Termumformungen notwendig. Es hilft, wenn die Klasse bereits etwas über Folgen und Reihen weiss, unbedingt notwendig ist das aber nicht. Bekannt sein sollten lineare und logarithmische Funktionen.

Aus der Informatik sollte grundlegendes Wissen über Algorithmen (z. B. binäre Suche) und deren Komplexität (inkl. O-Notation) vorhanden sein. Fehlt dieses, kann es im Rahmen der Aufgaben durchaus auch erarbeitet werden. In diesem Fall ist aber entsprechend mehr Zeit einzuplanen.

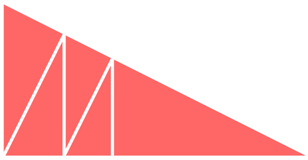
induktion.tex 3.5.2022 oda

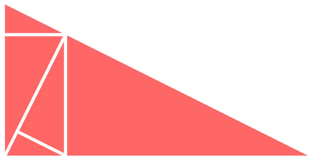
Von Induktion zu «Teile und Herrsche»

Eine Aufgabe zum Einstieg

Gegeben ist ein rechtwinkliges Dreieck. Dieses soll durch geradlinige Schnitte so zerteilt werden, dass lauter rechtwinklige Dreiecke entstehen. Zum Ausprobieren nehmen Sie am besten ein rechtwinkliges Papierdreieck und zerschneiden es mit einer Schere. Was ist alles möglich? Kann man so z. B. fünf oder sechs Dreiecke erhalten?

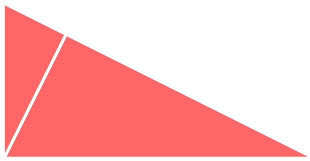
Mit etwas Ausprobieren lässt sich vermutlich rasch feststellen, dass das gegebene Dreieck mit n Schnitten in $n+1$ rechtwinklige Dreiecke zerteilt werden kann. Unten abgebildet sind zwei Beispiele für vier Schnitte, welche das Dreieck in fünf ähnliche rechtwinklige Dreiecke zerlegen.





Wir wollen uns im Folgenden nicht darum kümmern, wie viele verschiedene Möglichkeiten der Zerlegung es gibt. Wir wollen aber beweisen, dass es möglich ist, das gegebene rechtwinklige Dreieck in eine beliebige Anzahl von rechtwinkligen Dreiecken zu zerteilen.

Eine elegante Möglichkeit, dies zu tun, liefert die Mathematik mit der Beweismethode der sogenannten *vollständigen Induktion*. Dazu bezeichnen wir mit $A(n)$ die Aussage: „Ein rechtwinkliges Dreieck kann durch geradlinige Schnitte in n rechtwinklige Dreiecke zerteilt werden." Nun beweisen wir, dass aus $A(n)$ folgt, dass auch $A(n+1)$ gilt. Das ist aber schnell klar. Wir nehmen einfach eines der n rechtwinkligen Dreiecke und zerschneiden es wie abgebildet in zwei rechtwinklige Dreiecke. Dadurch erhalten wir natürlich $n+1$ rechtwinklige Dreiecke.



Man nennt $A(n) \implies A(n+1)$ den *Induktionsschritt*. Damit haben wir gezeigt, dass, wenn wir das Dreieck in drei rechtwinklige Dreiecke zerschneiden können, wir es auch in vier, fünf, usw. scheiden können. Die einzige offene Frage ist nun noch diejenige nach dem sogenannten *Induktionsanfang*. $A(2)$ ist aber auf jeden Fall korrekt, weil gemäss obigem Bild ein rechtwinkliges Dreieck natürlich in zwei rechtwinklige Dreiecke geteilt werden kann. Wir könnten sogar noch einen Schritt weiter zurückgehen und $A(1)$ als Induktionsanfang nehmen.

Weil nun einerseits $A(1)$ korrekt ist und wir gezeigt haben, dass $A(n) \implies A(n+1)$ gilt, folgt, dass ein rechtwinkliges Dreieck durch geradlinige Schnitte in eine beliebige Anzahl $n \in \mathbb{N}$ rechtwinkliger Dreiecke zerteilt werden kann.

Wir hätten unsere Bezeichnungen auch anders definieren können, so dass $D(n)$ die Anzahl Dreiecke angibt, welche bei n Schnitten entsteht:

$$D(0) = 1$$
$$D(n+1) = D(n) + 1$$

Daraus können wir nun eine Formel für $D(n)$ ableiten:

$$\begin{aligned} D(n+1) &= D(n)+1 \\ &= D(n-1)+1+1 = D(n-1)+2 \\ &= D(n-2)+3 \\ &= \ldots \\ &= D(n-k)+k+1 \end{aligned}$$

Beendet ist diese Gleichungskette, wenn $n-k=0$, also $n=k$. Dann erhalten wir als Endergebnis:

$$D(n+1) = \underbrace{D(0)}_{=1} + n + 1 = n+2$$

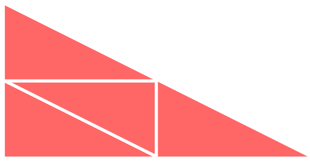
Somit folgt $D(n) = n+1$, d. h., exakt unsere Anfangsvermutung, dass wir mit n Schnitten in $n+1$ rechtwinklige Dreiecke zerlegen können. Streng betrachtet ist die oben durchgeführte Herleitung noch kein Beweis, d. h. wir erhalten nur eine Vermutung für $D(n)$ und müssen noch beweisen, dass diese Vermutung die Gleichungen $D(0) = 1$ und $D(n+1) = D(n)+1$ auch wirklich erfüllt. Das ist zum Glück kein grosses Problem. Wir nehmen die Vermutung $D(n) = n+1$, setzen diese ein und weisen das Verlangte so direkt nach.

$$\begin{aligned} D(0) &= 0+1 = 1 \quad \checkmark \\ D(n)+1 &= n+1+1 = n+2 = D(n+1) \quad \checkmark \end{aligned}$$

Bearbeiten Sie folgende Aufgaben auf eine ähnliche Art und Weise. Beweisen Sie jeweils ihre Vermutungen mit Hilfe des Verfahrens der vollständigen Induktion.

Aufgabe 1

Gegeben ist ein rechtwinkliges Dreieck. Dieses soll durch geradlinige Schnitte so zerteilt werden, dass lauter zueinander *kongruente* rechtwinklige Dreiecke entstehen. Nur mit einem Schnitt wird das nicht möglich sein, aber mit z. B. drei Schnitten klappt es, wie folgende Abbildung zeigt.



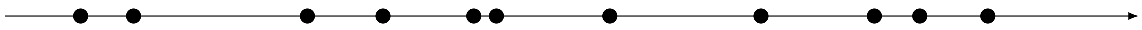
In vier kongruente Dreiecke kann also zerteilt werden. Welche weiteren Zerlegungen – uns interessiert die Anzahl der Dreiecke – sind möglich?

Aufgabe 2

Gegeben ist ein Quadrat. Dieses soll durch geradlinige Schnitte so zerteilt werden, dass lauter Quadrate entstehen. Die Quadrate müssen dabei nicht unbedingt kongruent sein. Welche Zerlegungen – uns interessiert auch hier die Anzahl der Quadrate – sind möglich?

Entwurf von Algorithmen

Das Prinzip der Induktion kann uns auch dabei helfen, effiziente Algorithmen zu finden. Wir betrachten als erstes Beispiel folgendes Problem: Gegeben sind n Punkte auf einer Geraden. Gesucht ist der minimale Abstand zwischen zwei Punkten.



Für die weitere Bearbeitung wollen wir annehmen, dass wir die Pixel-Koordinaten der Punkte in einer sortierten Liste als Eingabe haben. Für obiges Beispiel wäre das die Liste:

$$\{10, 17, 40, 50, 62, 65, 80, 100, 115, 121, 130\}$$

Naheliegende Lösungsidee

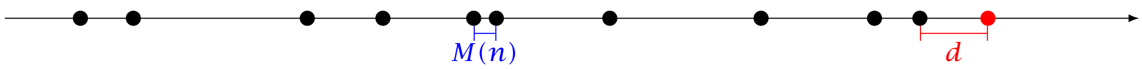
Die einfachste Lösung besteht wohl darin, die Liste von links nach rechts durchzugehen und jeweils die Differenz zweier aufeinanderfolgender Elemente zu berechnen. Ist diese Differenz kleiner als die bisherige, ersetzen wir die bisherige, andernfalls behalten wir sie.

Schritt	Differenz	Minimum
1	$17 - 10 = 7$	7
2	$40 - 17 = 23$	7
3	$50 - 40 = 10$	7
4	$62 - 50 = 12$	7
5	$65 - 62 = 3$	3
$\vdots$	$\vdots$	$\vdots$
10	$130 - 121 = 9$	3

Wir müssen insgesamt bei n Zahlen also $n - 1$ Subtraktionen und $n - 2$ Vergleiche durchführen. Dieser Algorithmus hat deshalb *lineare Komplexität*.

Induktion als Idee

Die Frage, welche sich bei einer induktiven Lösung stellt, ist folgende: Angenommen wir kennen das Minimum für n Punkte. Wie finden wir dann das Minimum für $n + 1$ Punkte? Da unsere Liste sortiert ist, beginnen wir einfach ganz links mit den ersten beiden Punkten. Danach nehmen wir den nächstgrösseren Wert hinzu und fahren so fort, bis wir mit der Liste durch sind. Beim Schritt von n zu $n + 1$ Punkten haben wir also als Ausgangslage die ersten n Punkte und kennen das Minimum $M(n)$. Nun kommt einfach der nächstgrössere Wert rechts hinzu. Das neue Minimum $M(n+1)$ finden wir, indem wir den Abstand d des neuen Punkts zum Punkt links davon berechnen und diesen mit unserem bisherigen Minimum $M(n)$ vergleichen. Es ist nun $M(n+1) = \min(M(n), d)$.



Wir müssen also bei jedem Schritt eine Differenz berechnen und einen Vergleich durchführen. Wir können für die Zeitkomplexität $T(n)$ des Algorithmus somit eine Rekursionsgleichung aufstellen:

$$T(n + 1) = T(n) + 2$$

Den Induktionsanfang haben wir bei $n = 2$. Wenn die Liste nur zwei Elemente enthält, dann entspricht das Minimum einfach der Differenz beider Elemente, d. h. $T(2) = 1$.

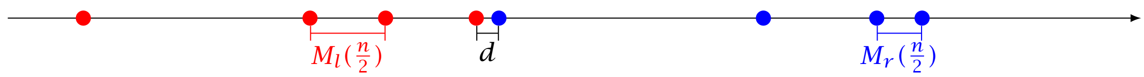
Aufgabe 3

Lösen Sie die Rekursionsgleichung $T(2) = 1$, $T(n+1) = T(n) + 2$. Was bekommen Sie für $T(n)$? Haben wir damit gegenüber der ersten Lösungsidee etwas gewonnen?

«Teile und Herrsche» als Idee

Vielleicht finden wir einen effizienteren Algorithmus, wenn wir die Menge der Punkte bei jedem Schritt halbieren? Damit das gut geht, wollen wir im Folgenden annehmen, dass die Anzahl n eine Potenz von 2 ist, d. h. $n = 2^k$ für ein $k \in \mathbb{N}$. Wir nehmen also an, dass wir die beiden Minima $M_l(\frac{n}{2})$ und $M_r(\frac{n}{2})$ der linken und rechten Hälfte kennen. Um nun das Minimum $M(n)$ zu bestimmen, müssen wir lediglich die Distanz d zwischen den beiden Teilen berechnen und dann mit den Minima der linken und rechten Hälfte vergleichen.

$$M(n) = \min\left(M_l\left(\frac{n}{2}\right), M_r\left(\frac{n}{2}\right), d\right)$$



Wenn wir diese Idee mit der vorherigen vergleichen, stellen wir fest, dass es sich um eine Verallgemeinerung der Induktion handelt: Wir teilen das Problem in kleinere Probleme auf, lösen diese separat und verwenden deren Lösung, um das Ursprungsproblem zu lösen. In der Informatik nennt man so eine Strategie «Teile und Herrsche».

Aufgabe 4

Können Sie ausgehend von obigen Überlegungen eine Rekursionsgleichung für die Komplexität des «Teile und Herrsche»-Algorithmus aufstellen und diese lösen? Haben wir nun etwas gewonnen?

Zum Abschluss steht noch ein neues Problem zur Bearbeitung an.

Aufgabe 5

Gegeben ist eine sortierte Liste von Zahlen, wobei jeder Wert, abgesehen von genau einem, doppelt vorkommt. Entwerfen Sie einen effizienten Algorithmus, welcher denjenigen Wert findet, der nur einmal vorkommt.

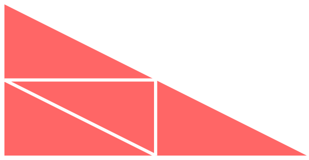
Zwei Beispiele solcher Listen:

$$\{1, 1, 2, 2, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8\}$$
$$\{10, 10, 17, 17, 18, 18, 19, 19, 21, 23, 23\}$$

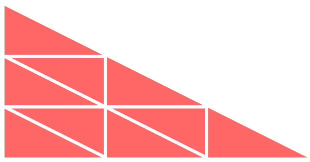
Für die Länge n einer solchen Liste gilt natürlich, dass $n = 2k + 1$ für $k \in \mathbb{N}$.

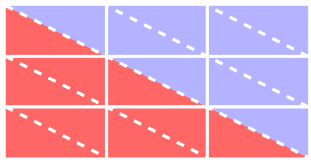
Lösungen

1. Wie bereits in der Aufgabenstellung gezeigt, können wir mit drei Schnitten das gegebene Dreieck in vier kongruente Dreiecke zerlegen.



Jetzt können wir natürlich in einem nächsten Schritt jedes dieser Dreiecke nochmals auf die gleiche Art unterteilen und erhalten so $4 \cdot 4 = 16$ kongruente Dreiecke. Dieses Vorgehen lässt sich formalisieren, indem wir die Bilder durchnummerieren und mit $A(n)$ die Anzahl kongruenter Dreiecke bezeichnen. In dieser Notation ist $A(1) = 4$ und $A(n+1) = 4 \cdot A(n)$. Wir könnten als Induktionsanfang auch das Ursprungsbild nehmen und dieses mit der Nummer 0 anschreiben. Damit ist $A(0) = 1$ und eine Vorschrift für $A(n)$ ist rasch gefunden. Es ist $A(n) = A(0) \cdot 4^n = 4^n$.
Das ist aber bei weitem noch nicht alles. Auch die unten links dargestellte Einteilung liefert kongruente Dreiecke. Es sind neun Stück, was wir leicht einsehen können. Eine schöne grafische Begründung ist beispielsweise im Bild rechts ersichtlich. Wir verdoppeln das rechtwinklige Dreieck zu einem Rechteck und teilen es in 3×3 Rechtecke, welche wir dann halbieren. Letztenlich haben wir also im Dreieck $2 \cdot \frac{3 \cdot 3}{2} = 3^2 = 9$ rechtwinklige Dreiecke.





Wir hätten als Induktionsschritt also auch $A(n+1) = 9 \cdot A(n)$ wählen können. Es lassen sich deshalb auch $9 \cdot 9 = 81$, bzw. allgemein 9^n kongruente Dreiecke erreichen.
Was wir entdeckt haben, lässt sich noch weiter ausbauen. Die Zerlegung kann nicht nur in vier oder neun rechtwinklige Dreiecke erfolgen. Wir können analog der Darstellung oben rechts beliebig in $m \cdot m = m^2$ Rechtecke zerschneiden und damit auch in m^2 kongruente Dreiecke zerlegen. Als Induktionsschritt können wir also allgemein $A(n+1) = m^2 \cdot A(n)$ schreiben, wobei $m \in \mathbb{N}$ eine beliebige natürliche Zahl ist.
Zusammengefasst ist somit jede Anzahl k an Dreiecken möglich, vorausgesetzt k lässt sich als Produkt von n Quadratzahlen schreiben, d. h. $k = m_1^2 \cdot m_2^2 \cdot \dots \cdot m_n^2$. Betrachten wir ein paar Beispiele genauer, fällt noch etwas auf:

$$\begin{aligned} k &= 4 \cdot 9 = 36 = 6^2 \\ k &= 4 \cdot 4 \cdot 16 = 256 = 16^2 \\ k &= 9 \cdot 25 \cdot 36 = 8100 = 90^2 \end{aligned}$$

Offensichtlich ist ein Produkt von Quadratzahlen immer auch selber eine Quadratzahl. Diese Festellung können wir als Behauptung formulieren und beweisen.

Behauptung: Sei $k = m_1^2 \cdot m_2^2 \cdot \dots \cdot m_n^2$ ein Produkt von n Quadratzahlen, dann gibt es ein $m \in \mathbb{N}$, so dass $k = m^2$.

Der Beweis kann direkt erfolgen.

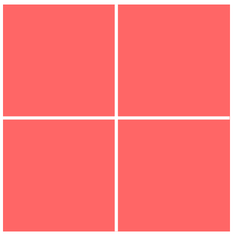
$$k = m_1^2 \cdot m_2^2 \cdot \dots \cdot m_n^2 = (\underbrace{m_1 \cdot m_2 \cdot \dots \cdot m_n}_{=m})^2 = m^2$$

Als Alternative zu diesem direkten Beweis können wir auch einen Induktionsbeweis führen. Wir bezeichnen dazu mit $Q(n)$ die Aussage: „Eine Zahl, welche als Produkt von n Quadratzahlen geschrieben werden kann, ist immer eine Quadratzahl.“
Der Induktionsanfang $Q(1)$ ist auf jeden Fall korrekt: Wenn eine Zahl $k = m_1^2$ aus nur einer Quadratzahl besteht, ist sie natürlich eine Quadratzahl. Wir müssen im Induktionsschritt nun zeigen, dass $Q(n+1)$ gilt, wenn $Q(n)$ korrekt ist.

$$k = \underbrace{m_1^2 \cdot m_2^2 \cdot \dots \cdot m_n^2}_{=q^2,\text{weil } Q(n) \text{ gilt}} \cdot m_{n+1}^2 = q^2 \cdot m_{n+1}^2 = (\underbrace{q \cdot m_{n+1}}_{=m})^2 = m^2 \quad \checkmark$$

Resümierend können wir bzgl. der gestellten Aufgabe also festhalten, dass ein gegebenes Dreieck durch geradlinige Schnitte in m^2 Dreiecke zerlegt werden kann, wobei $m \in \mathbb{N}$ eine beliebige natürliche Zahl ist.

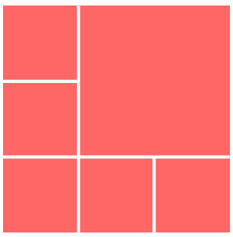
2. Auch hier ist eine erste Idee rasch gefunden.



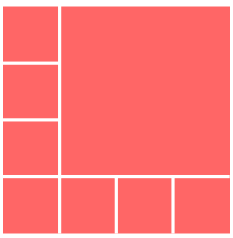
Da nur verlangt wird, dass es Quadrate sind, diese jedoch nicht kongruent sein müssen, kann im nächsten Schritt einfach eines der Quadrate in vier zerlegt werden. Die Anzahl der Quadrate vergrössert sich dabei um drei. Wir können für die Anzahl $A(n)$ der Quadrate im n-ten Schritt also folgenden Zusammenhang aufschreiben:

$$A(n+1) = A(n) + 3$$

Als Induktionsanfang nehmen wir $A(1) = 4$ und haben damit herausgefunden, dass wir das Quadrat in $4, 7, 10, 13, \ldots$ Quadrate zerlegen können.
Aber auch das ist noch nicht alles. Wir könnten für den Induktionsanfang auch wie folgt zerlegen und danach fortfahren, indem wir eines der Quadrate in vier zerlegen.



Wir haben also $A(1) = 6$ und $A(n+1) = A(n) + 3$. Damit stellen wir fest, dass wir das Quadrat in $6, 9, 12, 15, \ldots$ Quadrate zerlegen können.
Wir können unsere Bemühungen noch steigern, möglich ist auch $A(1) = 8$ als Induktionsanfang, wie folgendes Bild zeigt.



Als Folge davon können wir unsere Quadrate auch in $8, 11, 14, \dots$ Quadrate zerlegen.
Es gäbe noch mehr mögliche Induktionsanfänge, aber wir können aufhören, weil wir nun insgesamt gezeigt haben, dass eine Zerlegung in jede beliebige Anzahl grösser gleich sechs möglich ist.

3. Zu lösen ist die Rekursionsgleichung:

$$\begin{aligned} T(2) &= 1 \\ T(n+1) &= T(n) + 2 \end{aligned}$$

Wir können die Gleichung fortlaufend vereinfachen:

$$\begin{aligned} T(n+1) &= T(n) + 2 \\ &= T(n-1) + 2 + 2 = T(n-1) + 4 \\ &= T(n-2) + 6 \\ &= \dots \\ &= T(n-k) + 2(k+1) \end{aligned}$$

Beendet ist die Kette, wenn $n - k = 2$, also $k = n - 2$. Dann erhalten wir als Endergebnis:

$$T(n+1) = \underbrace{T(2)}_{=1} + 2(n-2+1) = 1 + 2n - 2 = 2n - 1$$

Es folgt:

$$T(n) = 2(n-1) - 1 = 2n - 3 \in \mathcal{O}(n)$$

Wir müssen nun noch zeigen, dass $T(n) = 2n - 3$ tatsächlich die Rekursionsgleichung löst, was wir durch Einsetzen rasch erledigen können.

$$\begin{aligned} T(2) &= 2 \cdot 2 - 3 = 1 \quad \checkmark \\ T(n) + 2 &= 2n - 3 + 2 = 2n - 1 = T(n+1) \quad \checkmark \end{aligned}$$

Dieser Algorithmus hat also ebenfalls *lineare Komplexität*, womit wir nur relativ wenig gewonnen haben.

4. Zur Bestimmung des Minimums ausgehend von den zwei Teillisten müssen wir eine Differenz berechnen und diese dann mit zwei Werten vergleichen. Das sind also drei Operationen. Wie bereits zuvor können wir als Induktionsanfang $T(2) = 1$ setzen. Somit erhalten wir folgende Rekursionsgleichung:

$$\begin{aligned} T(2) &= 1 \\ T(n) &= T\left(\frac{n}{2}\right) + T\left(\frac{n}{2}\right) + 3 = 2 \cdot T\left(\frac{n}{2}\right) + 3 \end{aligned}$$

Zur Lösung verfahren wir wieder wie bekannt, jedoch mit dem Unterschied, dass jeweils halbiert wird.

$$\begin{aligned} T(n) &= 2 \cdot T\left(\frac{n}{2}\right) + 3 \\ &= 2 \cdot \left(2 \cdot T\left(\frac{n}{4}\right) + 3\right) + 3 = 4 \cdot T\left(\frac{n}{4}\right) + 9 \\ &= 4 \cdot \left(2 \cdot T\left(\frac{n}{8}\right) + 3\right) + 9 = 8 \cdot T\left(\frac{n}{8}\right) + 21 \\ &= \dots \\ &= 2^k \cdot T\left(\frac{n}{2^k}\right) + 3 \cdot (2^k - 1) \end{aligned}$$

Beendet ist das Verfahren, wenn $\frac{n}{2^k} = 2$, also $2^k = \frac{n}{2}$. Es folgt:

$$T(n) = \frac{n}{2} + 3 \cdot \left(\frac{n}{2} - 1\right) = 2n - 3 \in \mathcal{O}(n)$$

Das ist tatsächlich eine Lösung der Rekursionsgleichung, was wir durch Einsetzen beweisen können.

$$\begin{aligned} T(2) &= 2 \cdot 2 - 3 = 1 \quad \checkmark \\ 2 \cdot T\left(\frac{n}{2}\right) + 3 &= 2 \cdot \left(2 \cdot \frac{n}{2} - 3\right) + 3 = 2n - 3 = T(n) \quad \checkmark \end{aligned}$$

Die Komplexität dieses Algorithmus ist somit also ebenfalls *linear*.

5. Die erste naheliegende Idee zur Lösung besteht wohl darin, einfach fortlaufend von links nach rechts Zweiergrüppchen zu kontrollieren. Sind die Elemente in diesem Grüppchen gleich, gehen wir einen Schritt weiter, sind sie ungleich, haben wir das einzelne Element gefunden und können es ausgeben. Im schlimmsten Fall müssen wir so bei $n = 2k + 1$ Zahlen k Vergleiche durchführen. Der Algorithmus hat deshalb *lineare Komplexität*.

$$1 \quad 1 \mid 2 \quad 2 \mid 3 \quad 4 \mid 4 \quad 5 \mid 5 \quad 6 \mid 6 \quad 7 \mid 7 \quad 8 \mid 8$$

Wenn wir uns die Zweiergrüppchen genauer anschauen, fällt auf, dass alle Zweiergrüppchen vor der gesuchten Zahl identische Elemente haben. Alle Zweiergrüppchen ab der gesuchten Zahl haben unterschiedliche Elemente. Die Idee für einen induktiven Ansatz besteht nun darin, bei einer gegebenen Liste das Grüppchen in der Mitte zu kontrollieren. Enthält dieses identische Elemente, wissen wir, dass wir im Teil nach diesem Grüppchen weitersuchen müssen. Enthält es ungleiche Elemente, wissen wir, dass wir im Teil bis und mit dem ersten Element dieses Grüppchens weitersuchen müssen. So lässt sich die Problemgrösse mit jedem Schritt ungefähr halbieren. Fertig sind wir, wenn die Liste nur noch ein Element enthält. Dieses ist dann die gesuchte Zahl.

$$1 \quad 1 \mid 2 \quad 2 \mid 3 \quad 4 \mid \boxed{4 \quad 5} \mid 5 \quad 6 \mid 6 \quad 7 \mid 7 \quad 8 \mid 8$$

$$1 \quad 1 \mid \boxed{2 \quad 2} \mid 3 \quad 4 \mid 4$$

$$\boxed{3 \quad 4} \mid 4$$

$$3$$

Wir wollen nun etwas über die Komplexität dieses Algorithmus herausfinden und stellen eine Rekursionsgleichung auf. In jedem Schritt ist genau ein Vergleich notwendig und wenn die Liste nur noch ein Element enthält, gibt es nichts mehr zu tun.

$$\begin{aligned} T(1) &= 0 \\ T(n) &= T\left(\frac{n}{2}\right) + 1 \end{aligned}$$

Die Lösung finden wir durch fortlaufendes Einsetzen:

$$
\begin{aligned}
T(n) &= T\left(\frac{n}{2}\right) + 1 \\
&= T\left(\frac{n}{4}\right) + 1 + 1 = T\left(\frac{n}{4}\right) + 2 \\
&= T\left(\frac{n}{8}\right) + 3 \\
&= \ldots \\
&= T\left(\frac{n}{2^k}\right) + k
\end{aligned}
$$

Fertig sind wir, wenn $n = 2^k$, bzw. $k = \log_2(n)$.

$$T(n) = T(1) + \log_2(n) = \log_2(n) \in \mathcal{O}(\log(n))$$

Abschliessend müssen wir noch beweisen, dass das wirklich eine Lösung der Rekursionsgleichung ist.

$$
\begin{aligned}
T(1) = \log_2(1) = 0 \quad \checkmark \\
T\left(\frac{n}{2}\right) + 1 = \log_2\left(\frac{n}{2}\right) + 1 = \log_2(n) - \underbrace{\log_2(2)}_{=1} + 1 = \log_2(n) = T(n) \quad \checkmark
\end{aligned}
$$

Dieser Algorithmus hat also *logarithmische Komplexität* und ist somit wesentlich besser.