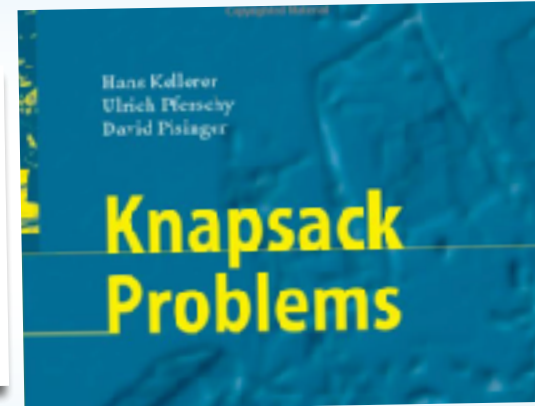




$$\begin{aligned} &\text{maximize} && \sum_{j=1}^n p_j x_j \\ &\text{subject to} && \sum_{j=1}^n w_j x_j \leq c, \\ &&& x_j = 0 \text{ or } 1, \quad j = 1, \dots, n. \end{aligned}$$



1 Einführung: Knapsack-Probleme

*Algorithmen und Datenstrukturen 2
Sommer 2022*

Prof. Dr. Sándor Fekete

Teilaufgaben?!

Teilaufgaben?!

Problem 1.3 (FRACTIONAL KNAPSACK).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe $z_i > 0$ Gewinn $p_i > 0$
- Größenschranke Z

Gesucht:

Für jedes Objekt ein Wert

$$x_i \in [0, 1]$$

sodass

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

Teilaufgaben?!

Problem 1.3 (FRACTIONAL KNAPSACK).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe $z_i > 0$ Gewinn $p_i > 0$
- Größenschranke Z

Gesucht:

Für jedes Objekt ein Wert

$$x_i \in [0, 1]$$

sodass

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

Teilaufgaben?!

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i$$
$$\sum_{i=1}^n p_i x_i$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i$$
$$\sum_{i=1}^n p_i x_i$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 4$$
$$\sum_{i=1}^n p_i x_i = 4$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 4$$
$$\sum_{i=1}^n p_i x_i = 4$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 12$$

$$\sum_{i=1}^n p_i x_i = 9$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 12$$

$$\sum_{i=1}^n p_i x_i = 9$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 28$$

$$\sum_{i=1}^n p_i x_i = 19$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 28$$

$$\sum_{i=1}^n p_i x_i = 19$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 48$$

$$\sum_{i=1}^n p_i x_i = 28$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 48$$

$$\sum_{i=1}^n p_i x_i = 28$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 56$$

$$\sum_{i=1}^n p_i x_i = 30$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 56$$

$$\sum_{i=1}^n p_i x_i = 30$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 96$$

$$\sum_{i=1}^n p_i x_i = 40$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 96$$

$$\sum_{i=1}^n p_i x_i = 40$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 96$$

$$\sum_{i=1}^n p_i x_i = 40$$

x_{15}

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	40	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 96$$

$$\sum_{i=1}^n p_i x_i = 40$$

x_{15}

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 96$$

$$\sum_{i=1}^n p_i x_i = 40$$

x_{15}

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 120$$

$$\sum_{i=1}^n p_i x_i = 40$$

x_{15}

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	10	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 120$$

$$\sum_{i=1}^n p_i x_i = 40$$

$$x_{15} = 0,6$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 120$$

$$\sum_{i=1}^n p_i x_i = 40$$

$$x_{15} = 0,6$$

Teilaufgaben?!

nach Wert $\left(\frac{z_i}{p_i}\right)$ sortieren:

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 120$$

$$\sum_{i=1}^n p_i x_i = 46$$

$$x_{15} = 0,6$$

Algorithmus

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Eingabe: $z_1, \dots, z_n, Z, p_1, \dots, p_n$

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Eingabe: $z_1, \dots, z_n, Z, p_1, \dots, p_n$

Ausgabe: $x_1, \dots, x_n \in [0, 1]$

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Eingabe: $z_1, \dots, z_n, Z, p_1, \dots, p_n$

Ausgabe: $x_1, \dots, x_n \in [0, 1]$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Eingabe: $z_1, \dots, z_n, Z, p_1, \dots, p_n$

Ausgabe: $x_1, \dots, x_n \in [0, 1]$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Eingabe: $z_1, \dots, z_n, Z, p_1, \dots, p_n$

Ausgabe: $x_1, \dots, x_n \in [0, 1]$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

1: Sortiere $\{1, \dots, n\}$ nach $\frac{z_i}{p_i}$ aufsteigend;

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Eingabe: $z_1, \dots, z_n, Z, p_1, \dots, p_n$

Ausgabe: $x_1, \dots, x_n \in [0, 1]$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

- 1: Sortiere $\{1, \dots, n\}$ nach $\frac{z_i}{p_i}$ aufsteigend;
Dies ergibt die Permutation $\pi(1), \dots, \pi(n)$.

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Eingabe: $z_1, \dots, z_n, Z, p_1, \dots, p_n$

Ausgabe: $x_1, \dots, x_n \in [0, 1]$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

- 1: Sortiere $\{1, \dots, n\}$ nach $\frac{z_i}{p_i}$ aufsteigend;
Dies ergibt die Permutation $\pi(1), \dots, \pi(n)$.
Setze $j = 1$.

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Eingabe: $z_1, \dots, z_n, Z, p_1, \dots, p_n$

Ausgabe: $x_1, \dots, x_n \in [0, 1]$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

- 1: *Sortiere* $\{1, \dots, n\}$ *nach* $\frac{z_i}{p_i}$ *aufsteigend*;
Dies ergibt die Permutation $\pi(1), \dots, \pi(n)$.
Setze $j = 1$.
- 2: **while** $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$ **do**

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Eingabe: $z_1, \dots, z_n, Z, p_1, \dots, p_n$

Ausgabe: $x_1, \dots, x_n \in [0, 1]$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

- 1: *Sortiere* $\{1, \dots, n\}$ *nach* $\frac{z_i}{p_i}$ *aufsteigend*;
Dies ergibt die Permutation $\pi(1), \dots, \pi(n)$.
Setze $j = 1$.
- 2: **while** $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$ **do**
- 3: $x_{\pi(j)} := 1$

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Eingabe: $z_1, \dots, z_n, Z, p_1, \dots, p_n$

Ausgabe: $x_1, \dots, x_n \in [0, 1]$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

- 1: *Sortiere* $\{1, \dots, n\}$ *nach* $\frac{z_i}{p_i}$ *aufsteigend;*
Dies ergibt die Permutation $\pi(1), \dots, \pi(n)$.
Setze $j = 1$.
- 2: **while** $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$ **do**
- 3: $x_{\pi(j)} := 1$
- 4: $j := j + 1$

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Eingabe: $z_1, \dots, z_n, Z, p_1, \dots, p_n$

Ausgabe: $x_1, \dots, x_n \in [0, 1]$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

1: Sortiere $\{1, \dots, n\}$ nach $\frac{z_i}{p_i}$ aufsteigend;

Dies ergibt die Permutation $\pi(1), \dots, \pi(n)$.

Setze $j = 1$.

2: **while** $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$ **do**

3: $x_{\pi(j)} := 1$

4: $j := j + 1$

5: Setze $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$

Algorithmus

Algorithmus 1.4. (*Greedy-Algorithmus*)

Eingabe: $z_1, \dots, z_n, Z, p_1, \dots, p_n$

Ausgabe: $x_1, \dots, x_n \in [0, 1]$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

- 1: *Sortiere* $\{1, \dots, n\}$ nach $\frac{z_i}{p_i}$ *aufsteigend*;
Dies ergibt die *Permutation* $\pi(1), \dots, \pi(n)$.
Setze $j = 1$.
 - 2: **while** $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$ **do**
 - 3: $x_{\pi(j)} := 1$
 - 4: $j := j + 1$
 - 5: *Setze* $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$
 - 6: **return**
-

Das klappt immer!

Das klappt immer!

Satz 1.5. *Algorithmus 1.4 liefert eine optimale Lösung für Problem 1.3.*

Das klappt immer!

Satz 1.5. *Algorithmus 1.4 liefert eine optimale Lösung für Problem 1.3.*

Beweis:

Das klappt immer!

Satz 1.5. Algorithmus 1.4 liefert eine optimale Lösung für Problem 1.3.

Beweis:

O.B.d.A. sei $\sum_{i=1}^n z_{\pi(i)} > Z$ - sonst nimmt man einfach alles.

Das klappt immer!

Satz 1.5. Algorithmus 1.4 liefert eine optimale Lösung für Problem 1.3.

Beweis:

O.B.d.A. sei $\sum_{i=1}^n z_{\pi(i)} > Z$ - sonst nimmt man einfach alles.

(1) Wir bekommen eine zulässige Lösung.

Das klappt immer!

Satz 1.5. Algorithmus 1.4 liefert eine optimale Lösung für Problem 1.3.

Beweis:

O.B.d.A. sei $\sum_{i=1}^n z_{\pi(i)} > Z$ - sonst nimmt man einfach alles.


- (1) Wir bekommen eine zulässige Lösung.
- (2) Wir bekommen eine optimale Lösung.

(1) Zulässigkeit

(1) Zulässigkeit


```
Setze  $j = 1$ .  
2: while  $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$  do  
3:    $x_{\pi(j)} := 1$   
4:    $j := j + 1$   
5: Setze  $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$   
6: return
```

(1) Zulässigkeit



```
Setze  $j = 1$ .  
2: while  $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$  do  
3:    $x_{\pi(j)} := 1$   
4:    $j := j + 1$   
5: Setze  $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$   
6: return
```

(1) Zulässigkeit



```
Setze  $j = 1$ .
2: while  $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$  do
3:    $x_{\pi(j)} := 1$ 
4:    $j := j + 1$ 
5: Setze  $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$ 
6: return
```

Sei j^* der letzte Index, für den die **while**-Bedingung getestet wird.

(1) Zulässigkeit

Setze $j = 1$.

2: **while** $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$ **do**

3: $x_{\pi(j)} := 1$

4: $j := j + 1$

5: **Setze** $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$

6: **return**

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 120$$

$$\sum_{i=1}^n p_i x_i = 46$$

$$x_{15} = 0,6$$

Sei j^* der letzte Index, für den die **while**-Bedingung getestet wird.

(1) Zulässigkeit

Setze $j = 1$.

2: **while** $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$ **do**

3: $x_{\pi(j)} := 1$

4: $j := j + 1$

5: **Setze** $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$

6: **return**

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 120$$

$$\sum_{i=1}^n p_i x_i = 46$$

$$x_{15} = 0,6$$

Sei j^* der letzte Index, für den die **while**-Bedingung getestet wird.

Dann ist

(1) Zulässigkeit

Setze $j = 1$.

2: **while** $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$ **do**

3: $x_{\pi(j)} := 1$

4: $j := j + 1$

5: **Setze** $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$

6: **return**

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$$\sum_{i=1}^n z_i x_i = 120$$

$$\sum_{i=1}^n p_i x_i = 46$$

$$x_{15} = 0,6$$

Sei j^* der letzte Index, für den die **while**-Bedingung getestet wird.

Dann ist
$$\sum_{i=1}^{j^*-1} z_{\pi(i)} \leq Z$$

(1) Zulässigkeit

```

Setze  $j = 1$ .
2: while ( $\sum_{i=1}^j z_{\pi(i)} \leq Z$ ) do
3:    $x_{\pi(j)} := 1$ 
4:    $j := j + 1$ 
5: Setze  $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$ 
6: return
    
```

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$\sum_{i=1}^n z_i x_i = 120$
 $\sum_{i=1}^n p_i x_i = 46$
 $x_{15} = 0,6$

Sei j^* der letzte Index, für den die **while**-Bedingung getestet wird.

Dann ist $\sum_{i=1}^{j^*-1} z_{\pi(i)} \leq Z$ und

(1) Zulässigkeit

```

Setze  $j = 1$ .
2: while ( $\sum_{i=1}^j z_{\pi(i)} \leq Z$ ) do
3:    $x_{\pi(j)} := 1$ 
4:    $j := j + 1$ 
5: Setze  $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$ 
6: return
    
```

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$\sum_{i=1}^n z_i x_i = 120$
 $\sum_{i=1}^n p_i x_i = 46$
 $x_{15} = 0,6$

Sei j^* der letzte Index, für den die **while**-Bedingung getestet wird.

Dann ist $\sum_{i=1}^{j^*-1} z_{\pi(i)} \leq Z$ und $\sum_{i=1}^{j^*} z_{\pi(i)} > Z$

(1) Zulässigkeit



```

Setze  $j = 1$ .
2: while ( $\sum_{i=1}^j z_{\pi(i)} \leq Z$ ) do
3:    $x_{\pi(j)} := 1$ 
4:    $j := j + 1$ 
5: Setze  $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$ 
6: return
    
```

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$\sum_{i=1}^n z_i x_i = 120$
 $\sum_{i=1}^n p_i x_i = 46$
 $x_{15} = 0,6$

Sei j^* der letzte Index, für den die **while**-Bedingung getestet wird.

Dann ist $\sum_{i=1}^{j^*-1} z_{\pi(i)} \leq Z$ und $\sum_{i=1}^{j^*} z_{\pi(i)} > Z$

Also ist

(1) Zulässigkeit



Setze $j = 1$.
 2: **while** $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$ **do**
 3: $x_{\pi(j)} := 1$
 4: $j := j + 1$
 5: Setze $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$
 6: **return**

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$\sum_{i=1}^n z_i x_i = 120$
 $\sum_{i=1}^n p_i x_i = 46$
 $x_{15} = 0,6$

Sei j^* der letzte Index, für den die **while**-Bedingung getestet wird.

Dann ist $\sum_{i=1}^{j^*-1} z_{\pi(i)} \leq Z$ und $\sum_{i=1}^{j^*} z_{\pi(i)} > Z$

Also ist $x_{\pi(1)}, \dots, x_{\pi(j^*-1)} = 1$

(1) Zulässigkeit



Setze $j = 1$.
 2: **while** $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$ **do**
 3: $x_{\pi(j)} := 1$
 4: $j := j + 1$
 5: Setze $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$
 6: **return**

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$\sum_{i=1}^n z_i x_i = 120$
 $\sum_{i=1}^n p_i x_i = 46$
 $x_{15} = 0,6$

Sei j^* der letzte Index, für den die **while**-Bedingung getestet wird.

Dann ist $\sum_{i=1}^{j^*-1} z_{\pi(i)} \leq Z$ und $\sum_{i=1}^{j^*} z_{\pi(i)} > Z$

Also ist $x_{\pi(1)}, \dots, x_{\pi(j^*-1)} = 1$ und

(1) Zulässigkeit

Setze $j = 1$.
 2: **while** $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$ **do**
 3: $x_{\pi(j)} := 1$
 4: $j := j + 1$
 5: **Setze** $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$
 6: **return**

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$\sum_{i=1}^n z_i x_i = 120$
 $\sum_{i=1}^n p_i x_i = 46$
 $x_{15} = 0,6$

Sei j^* der letzte Index, für den die **while**-Bedingung getestet wird.

Dann ist $\sum_{i=1}^{j^*-1} z_{\pi(i)} \leq Z$ und $\sum_{i=1}^{j^*} z_{\pi(i)} > Z$

Also ist $x_{\pi(1)}, \dots, x_{\pi(j^*-1)} = 1$ und $x_{\pi(j^*)} \in [0, 1[$

(1) Zulässigkeit

```

Setze  $j = 1$ .
2: while  $(\sum_{i=1}^j z_{\pi(i)} \leq Z)$  do
3:    $x_{\pi(j)} := 1$ 
4:    $j := j + 1$ 
5: Setze  $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$ 
6: return
  
```

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$\sum_{i=1}^n z_i x_i = 120$
 $\sum_{i=1}^n p_i x_i = 46$
 $x_{15} = 0,6$

Sei j^* der letzte Index, für den die **while**-Bedingung getestet wird.

Dann ist $\sum_{i=1}^{j^*-1} z_{\pi(i)} \leq Z$ und $\sum_{i=1}^{j^*} z_{\pi(i)} > Z$

Also ist $x_{\pi(1)}, \dots, x_{\pi(j^*-1)} = 1$ und $x_{\pi(j^*)} \in [0, 1[$ und

(1) Zulässigkeit

```

Setze  $j = 1$ .
2: while ( $\sum_{i=1}^j z_{\pi(i)} \leq Z$ ) do
3:    $x_{\pi(j)} := 1$ 
4:    $j := j + 1$ 
5: Setze  $x_{\pi(j)} := \frac{Z - \sum_{i=1}^{j-1} z_{\pi(i)}}{z_{\pi(j)}}$ 
6: return
    
```

i	6	4	13	12	9	3	15	8	16	10	1	14	5	11	2	7
z_i	4	8	16	20	8	40	24	40	24	32	20	20	16	28	32	32
p_i	4	5	10	9	2	10	6	9	4	5	3	3	2	3	3	2

$\sum_{i=1}^n z_i x_i = 120$
 $\sum_{i=1}^n p_i x_i = 46$
 $x_{15} = 0,6$

Sei j^* der letzte Index, für den die **while**-Bedingung getestet wird.

Dann ist $\sum_{i=1}^{j^*-1} z_{\pi(i)} \leq Z$ und $\sum_{i=1}^{j^*} z_{\pi(i)} > Z$

Also ist $x_{\pi(1)}, \dots, x_{\pi(j^*-1)} = 1$ und $x_{\pi(j^*)} \in [0, 1[$ und $\sum_{i=1}^{j^*} x_{\pi(i)} z_{\pi(i)} = Z$

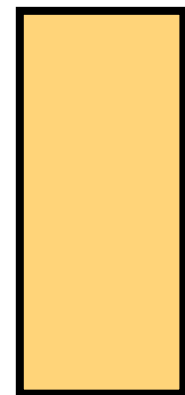
(2) Optimalität

(2) Optimalität

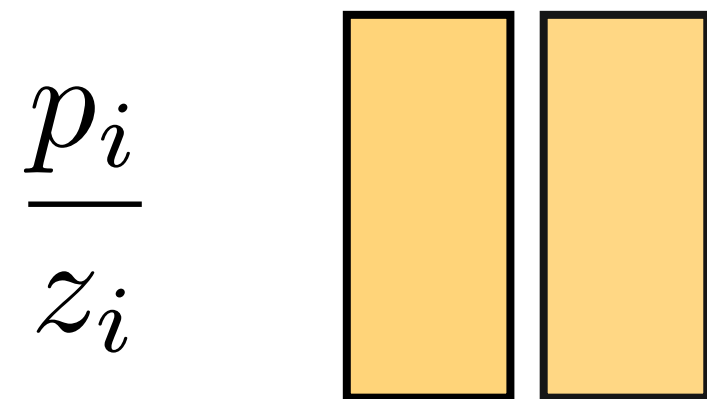
$$\frac{p_i}{z_i}$$

(2) Optimalität

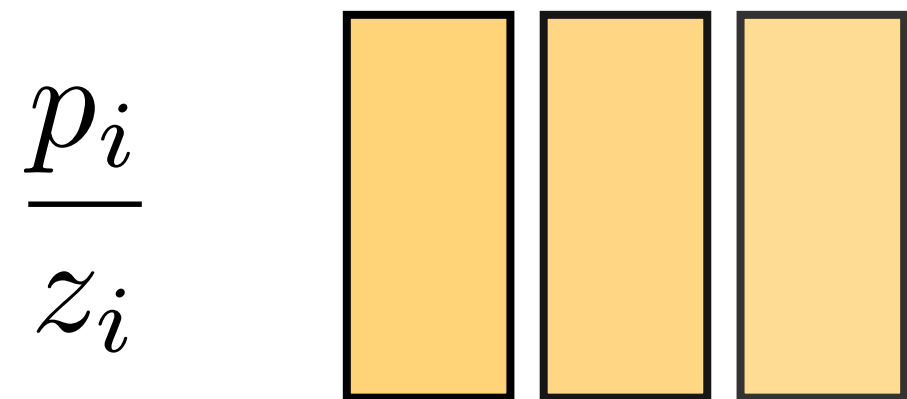
$$\frac{p_i}{z_i}$$



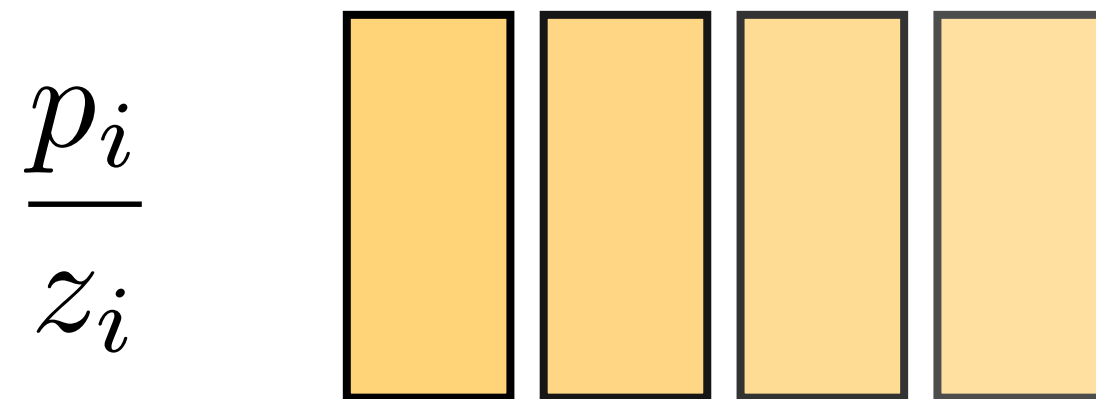
(2) Optimalität



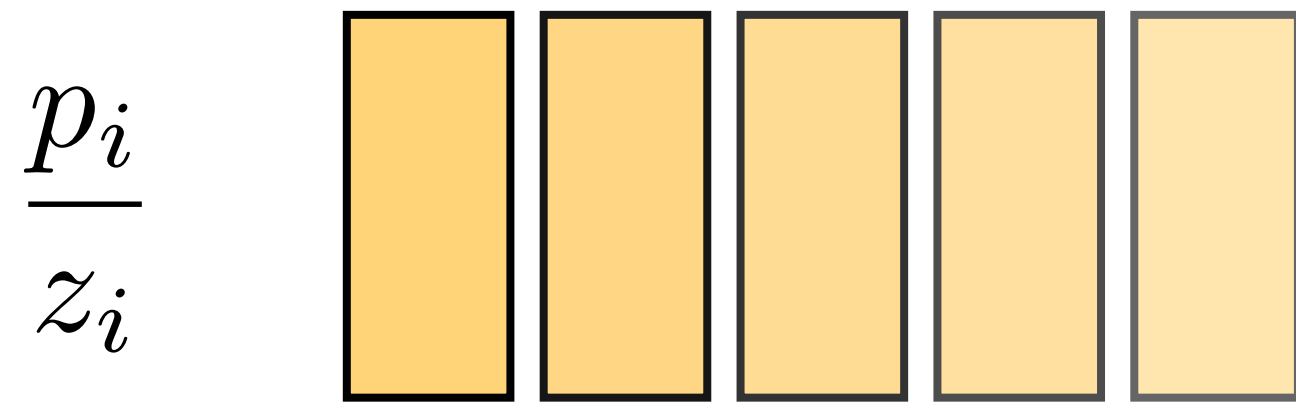
(2) Optimalität



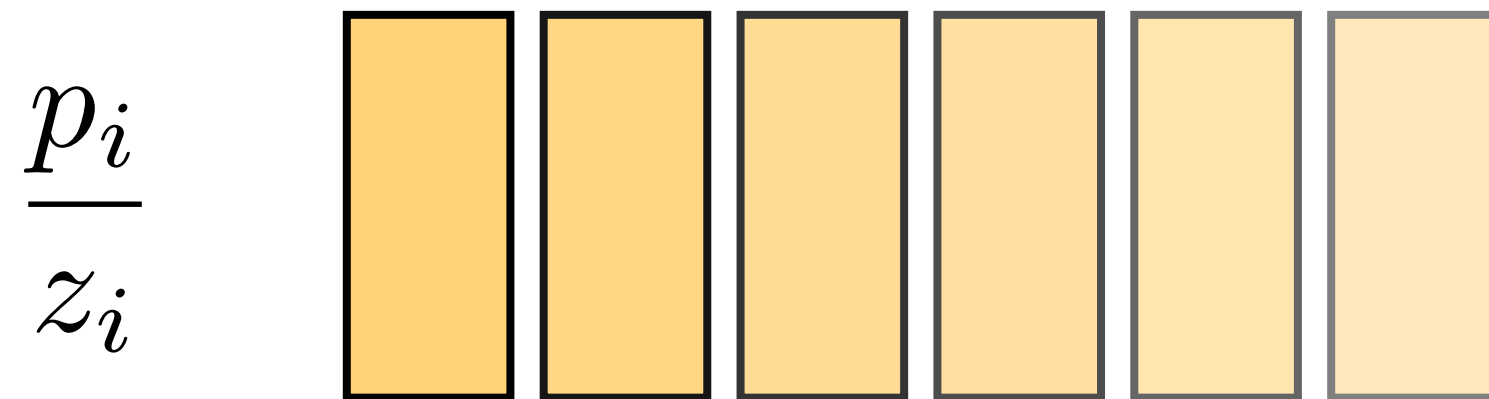
(2) Optimalität



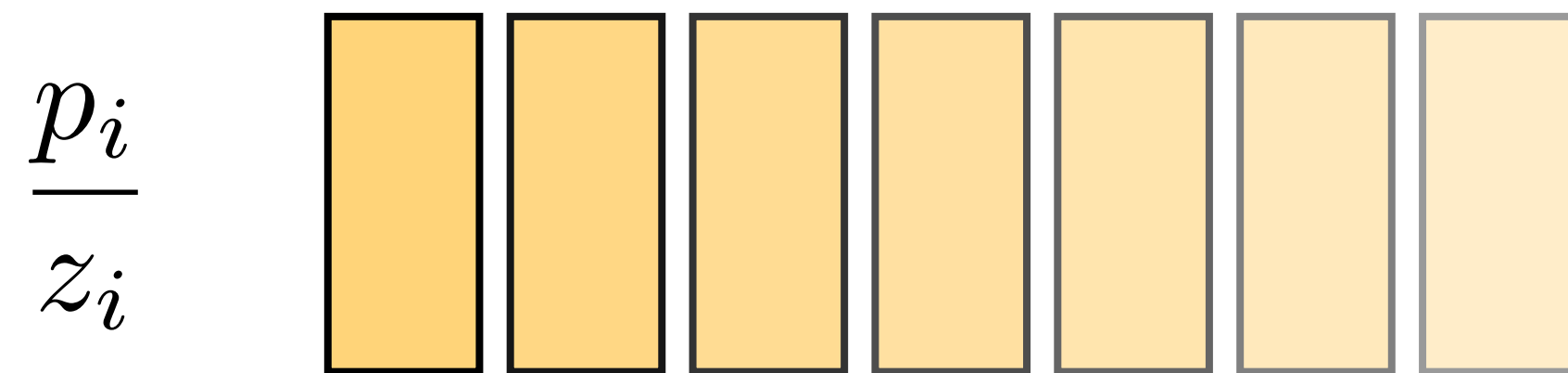
(2) Optimalität



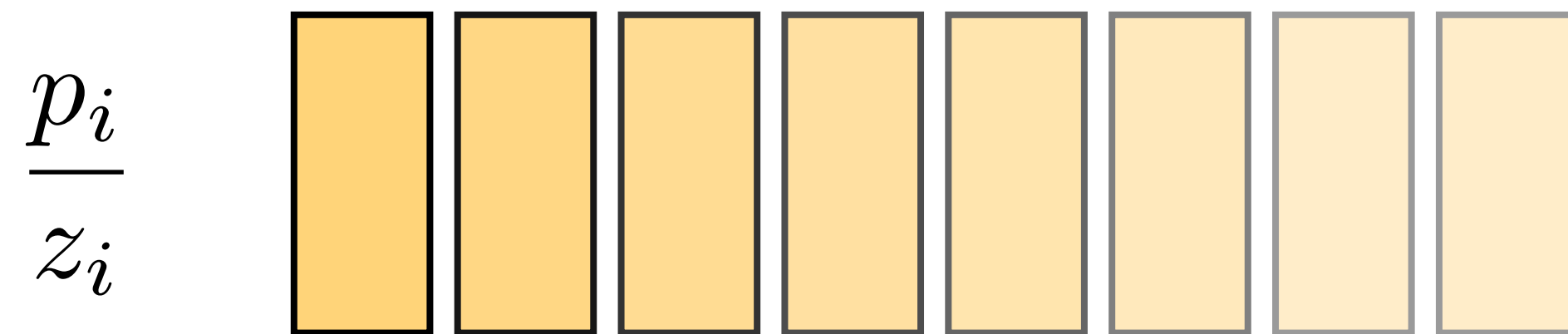
(2) Optimalität



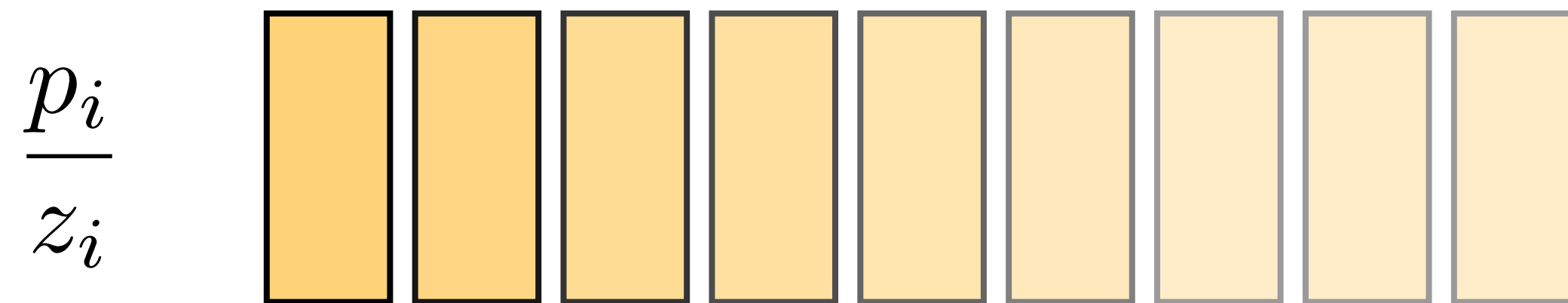
(2) Optimalität



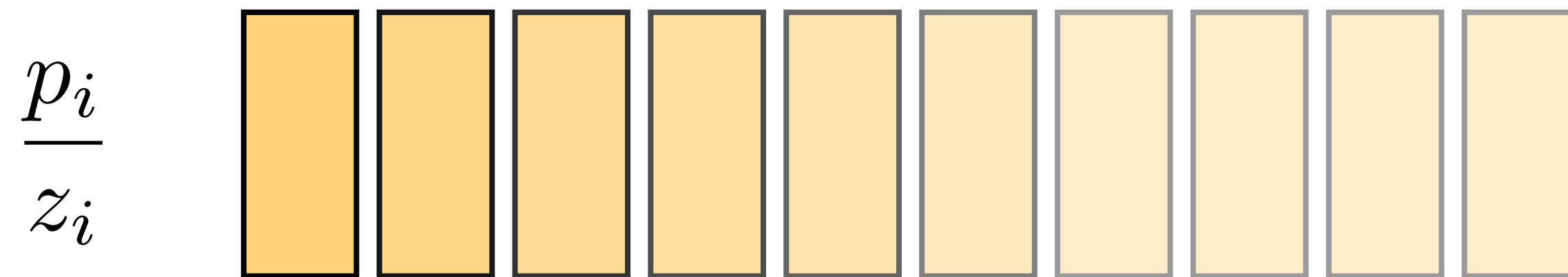
(2) Optimalität



(2) Optimalität



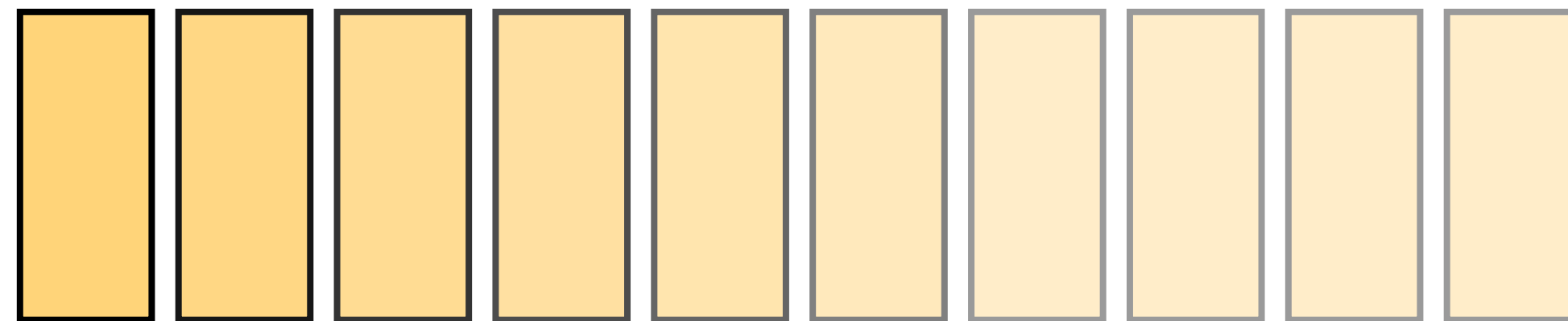
(2) Optimalität



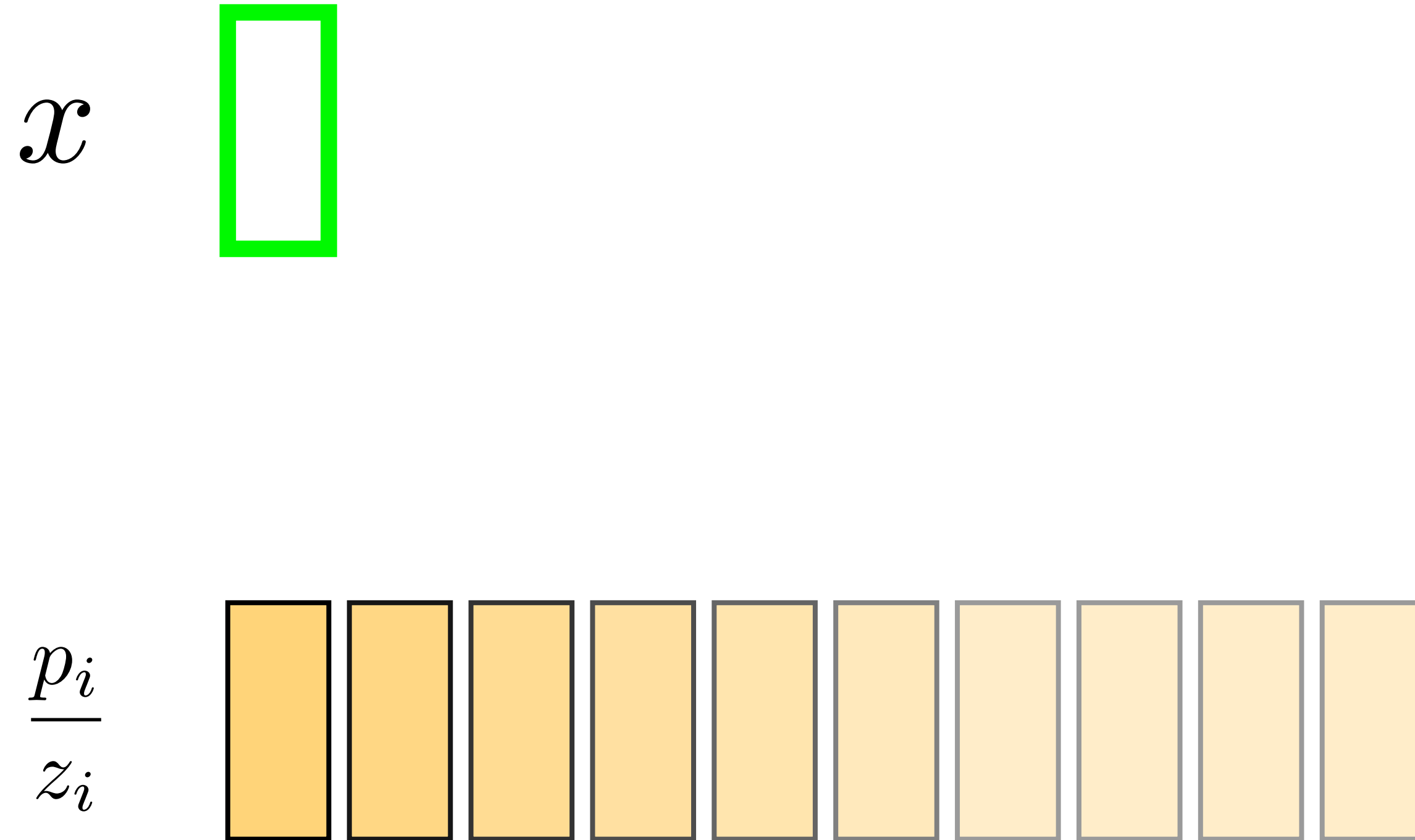
(2) Optimalität

x

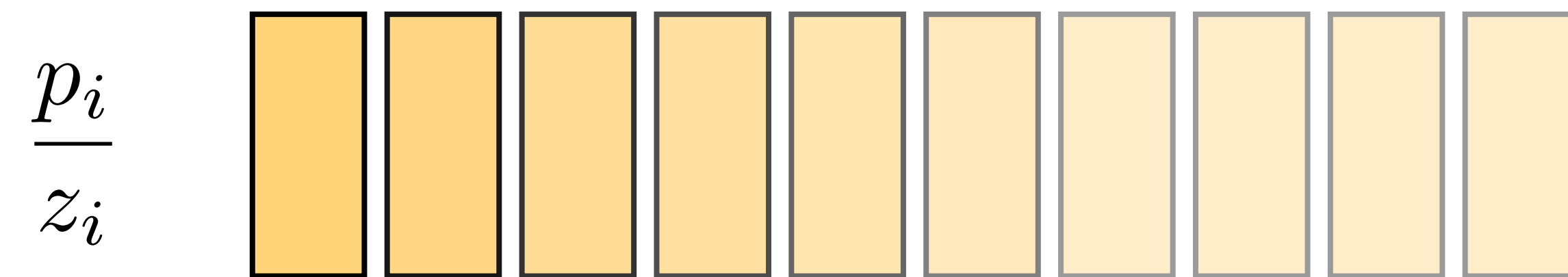
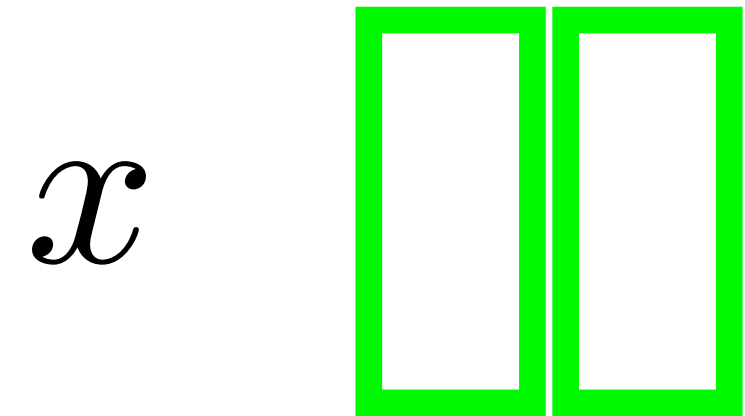
$\frac{p_i}{z_i}$



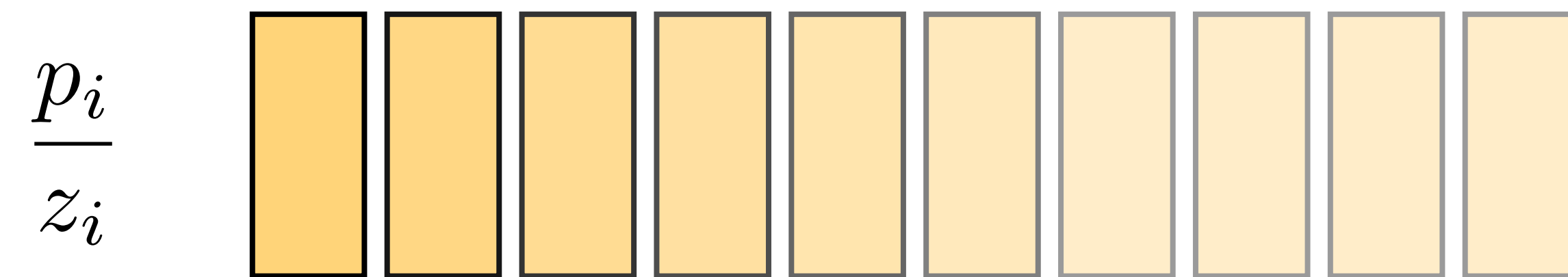
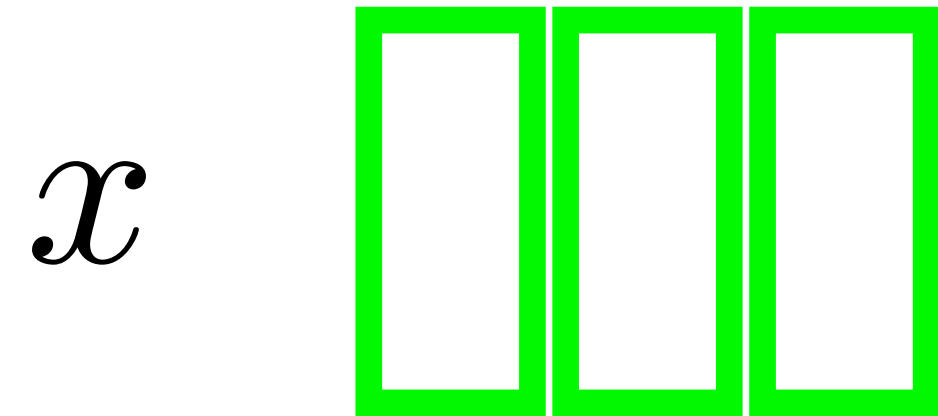
(2) Optimalität



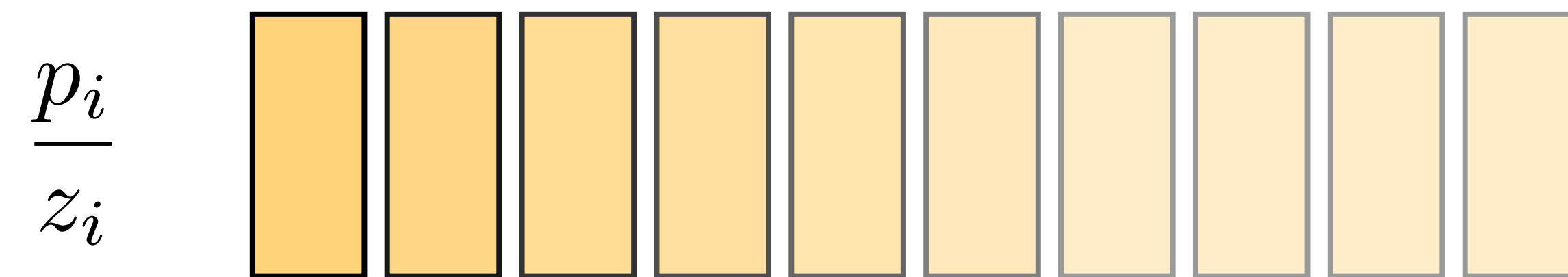
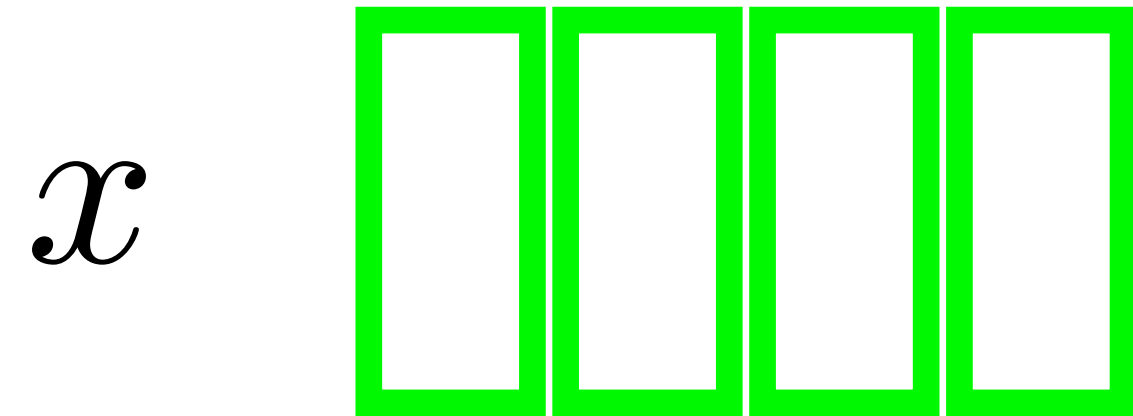
(2) Optimalität



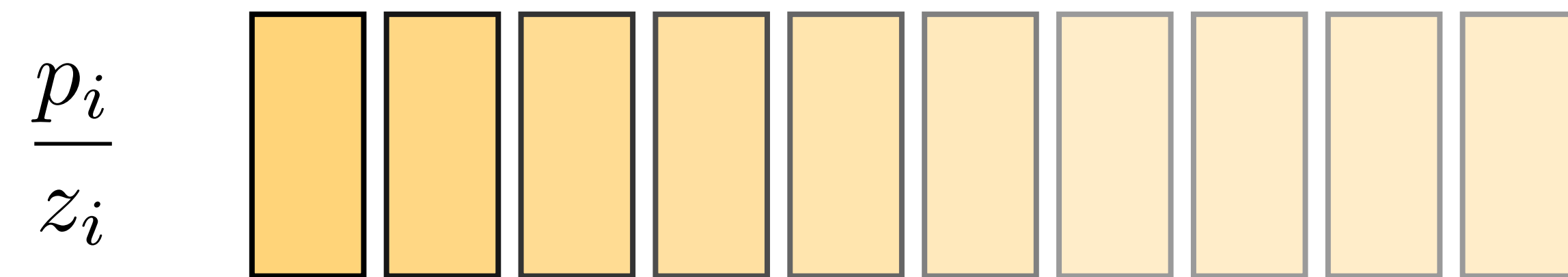
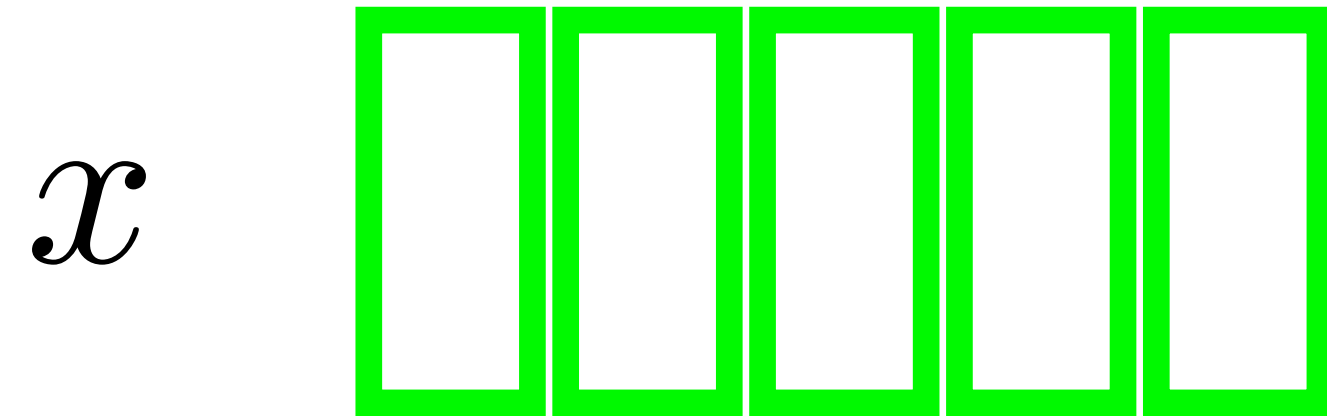
(2) Optimalität



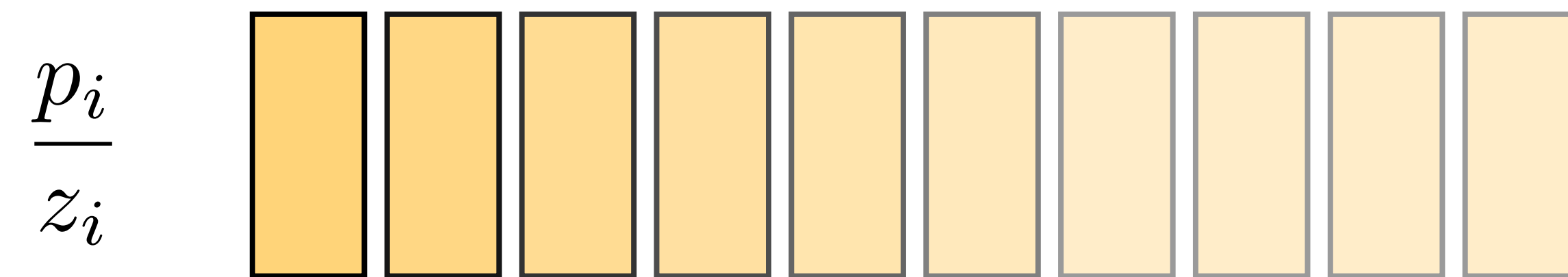
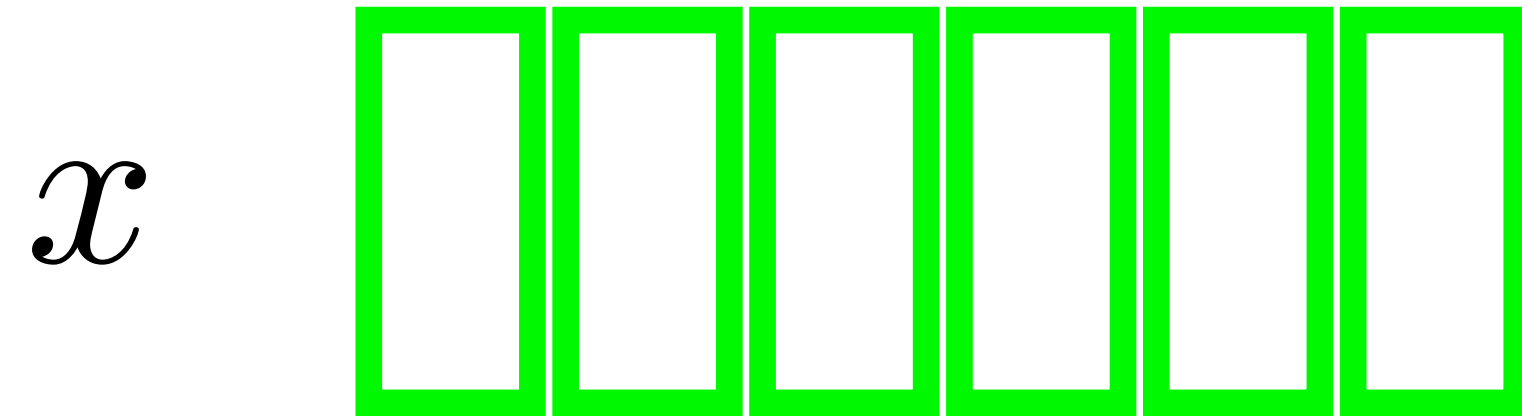
(2) Optimalität



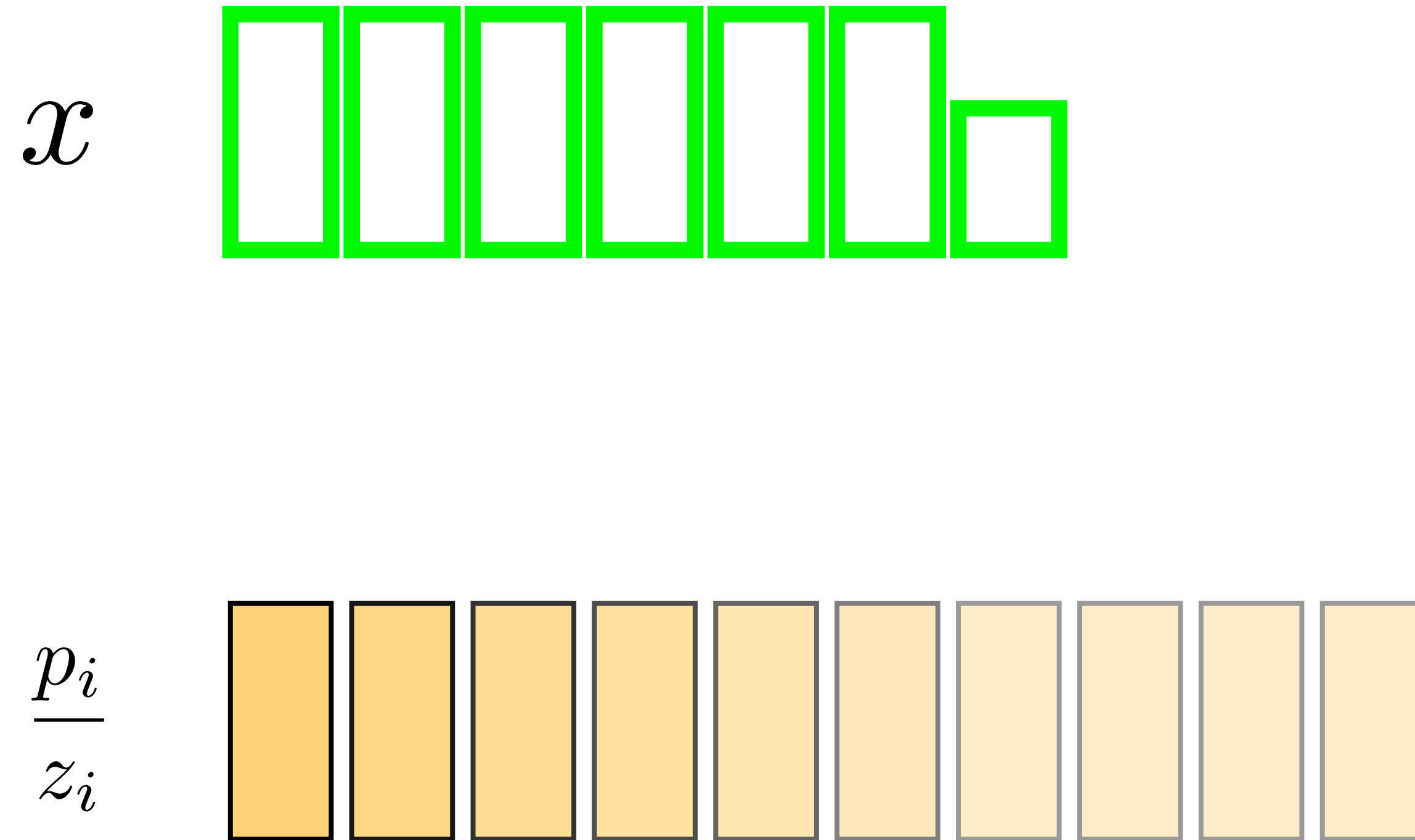
(2) Optimalität



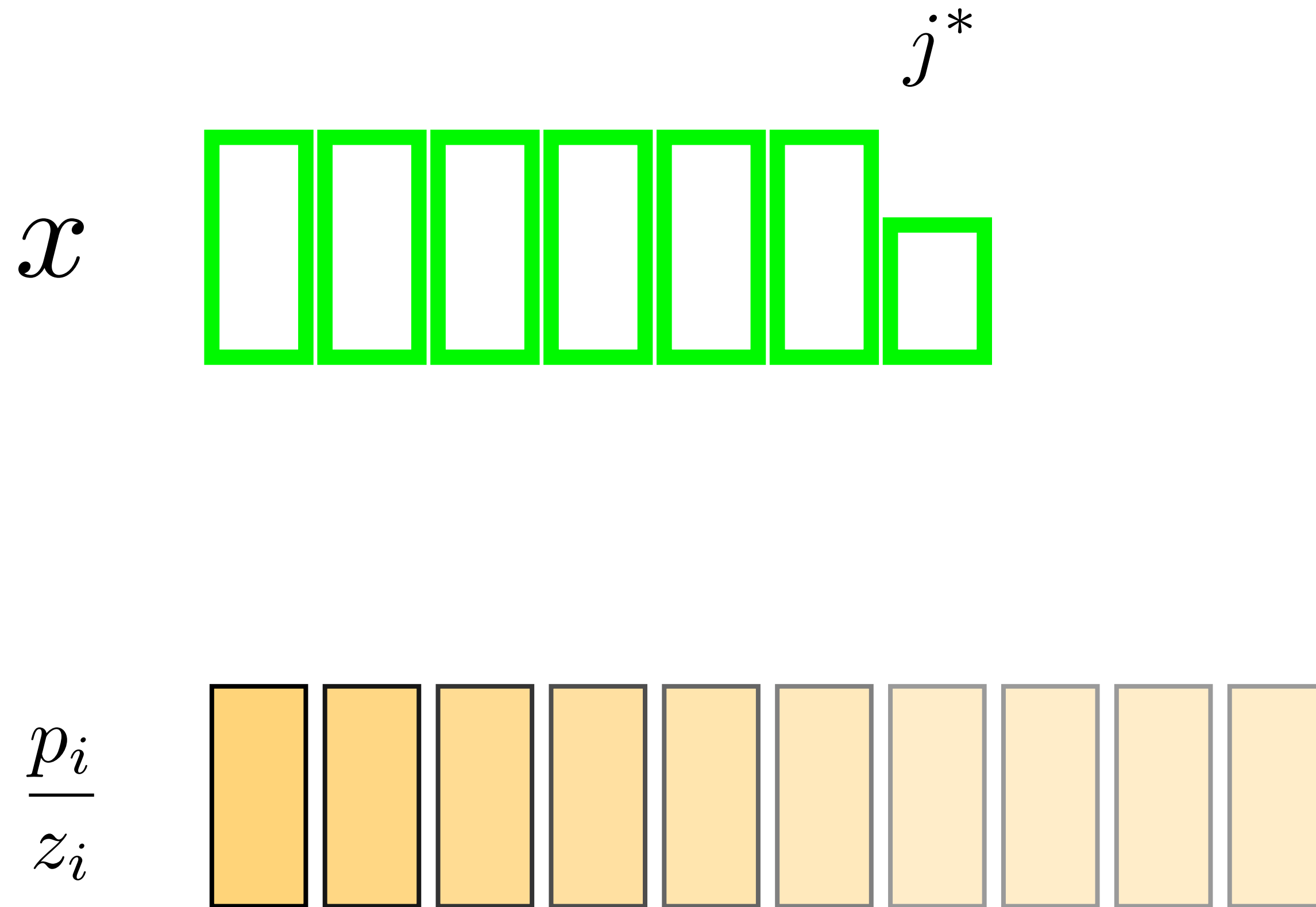
(2) Optimalität



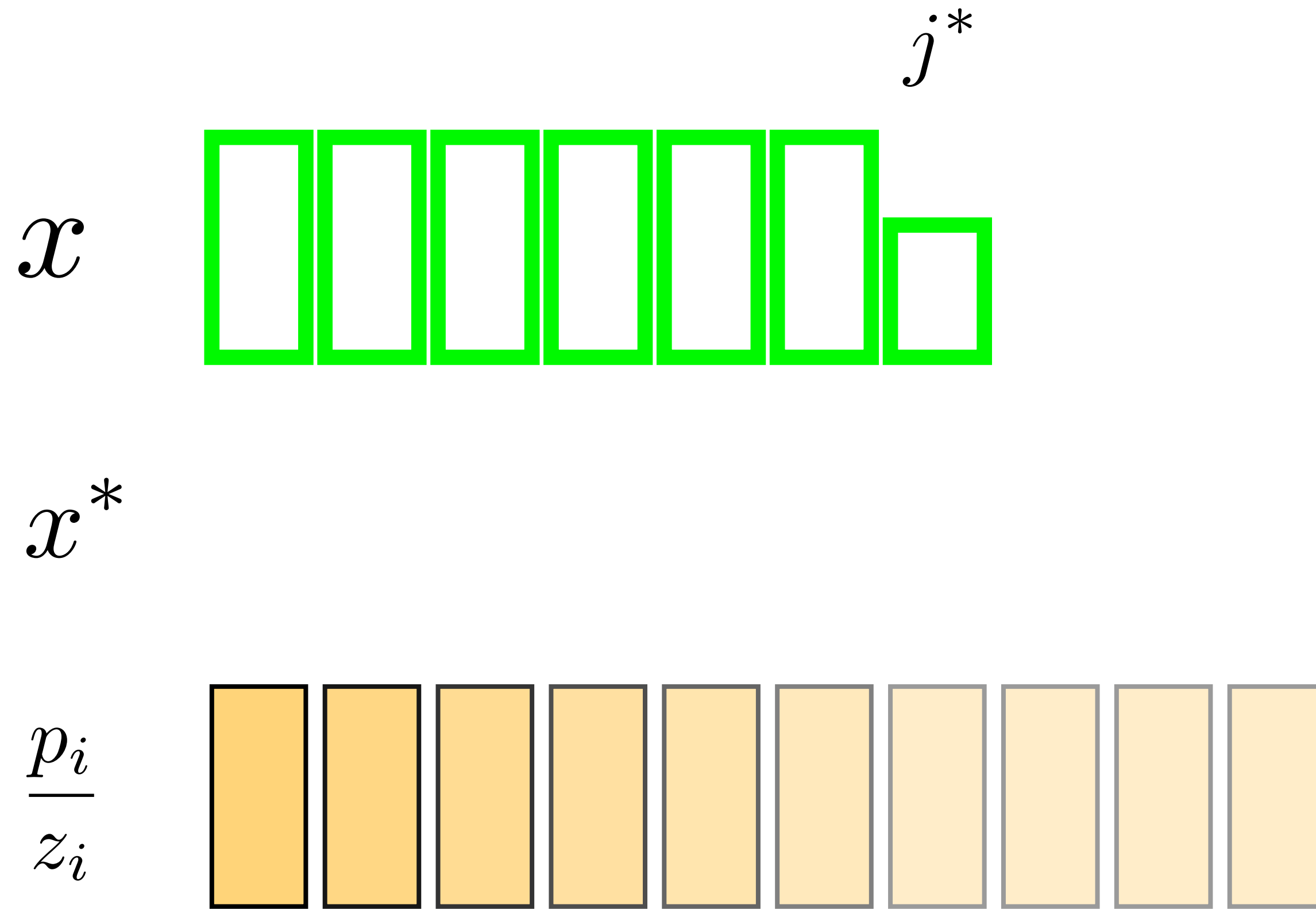
(2) Optimalität



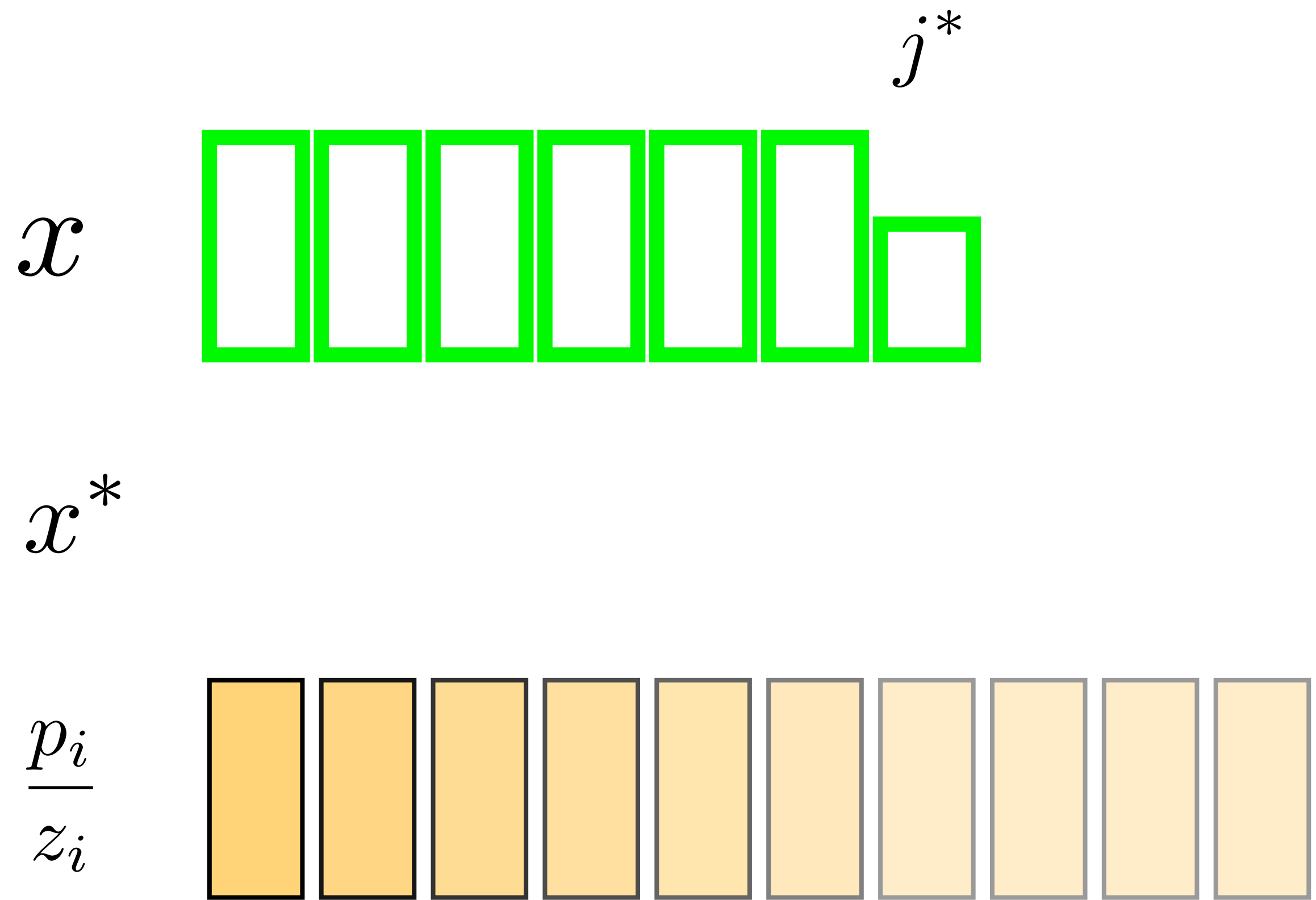
(2) Optimalität



(2) Optimalität

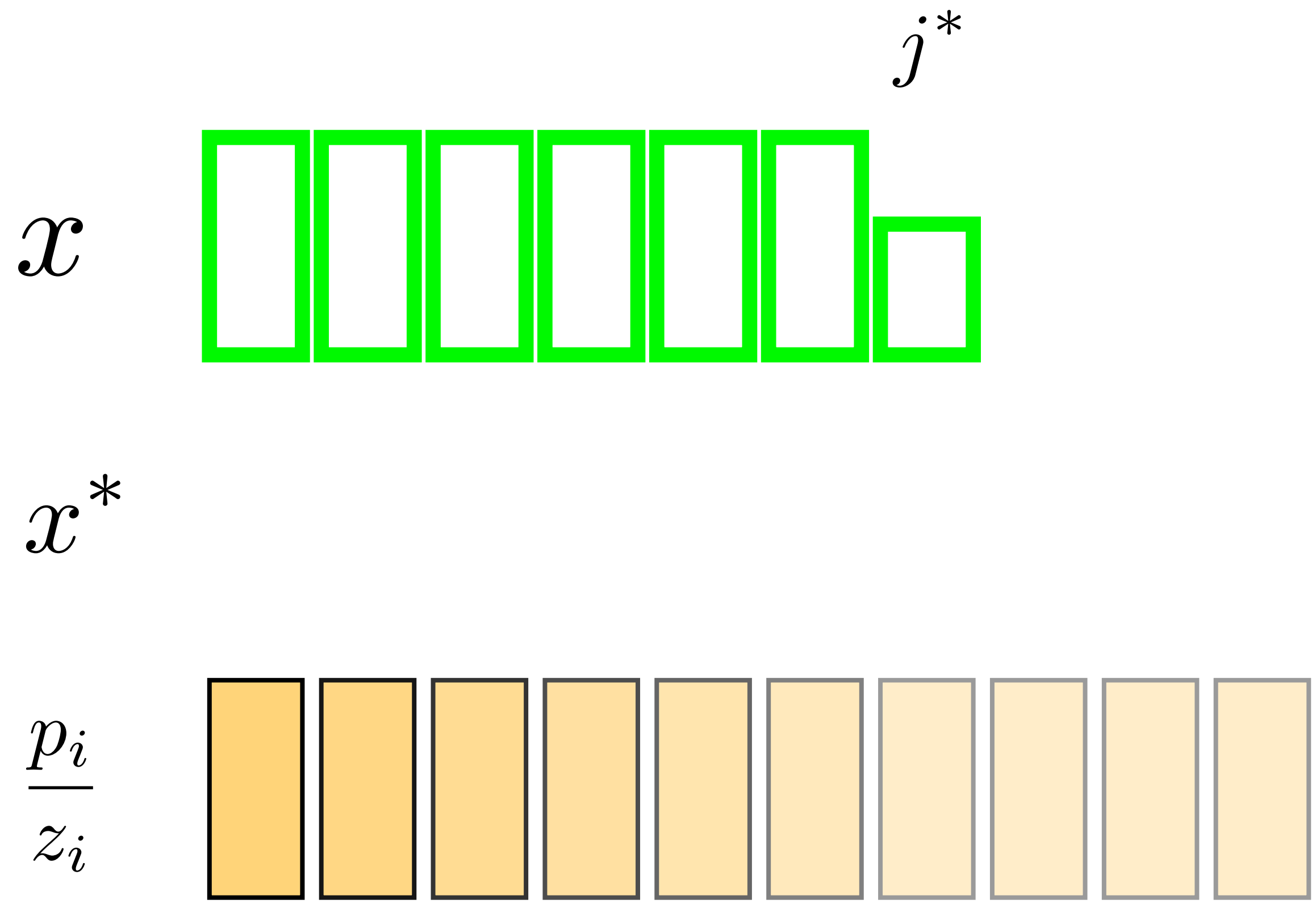


(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

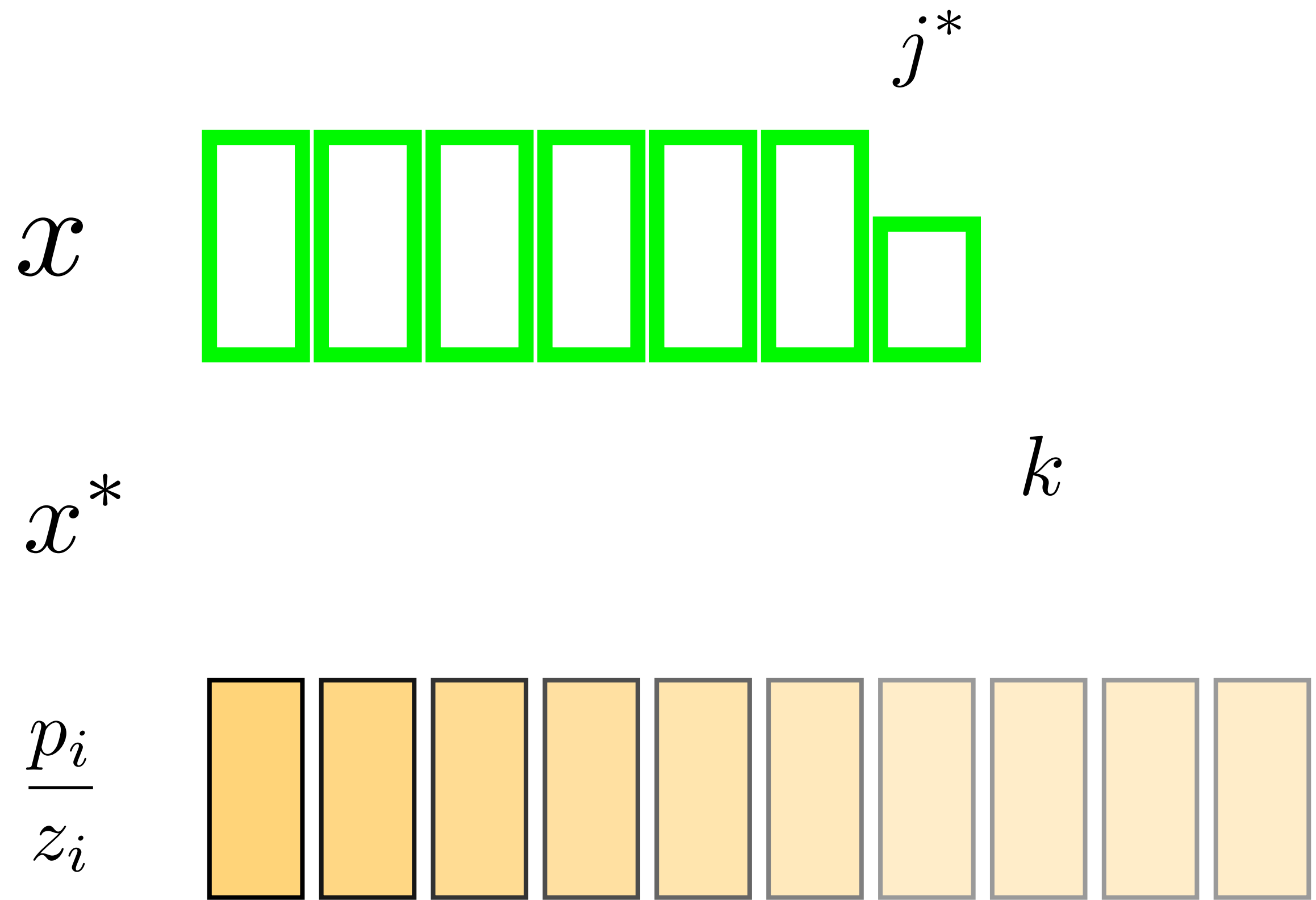
(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

$$p_i > 0 :$$

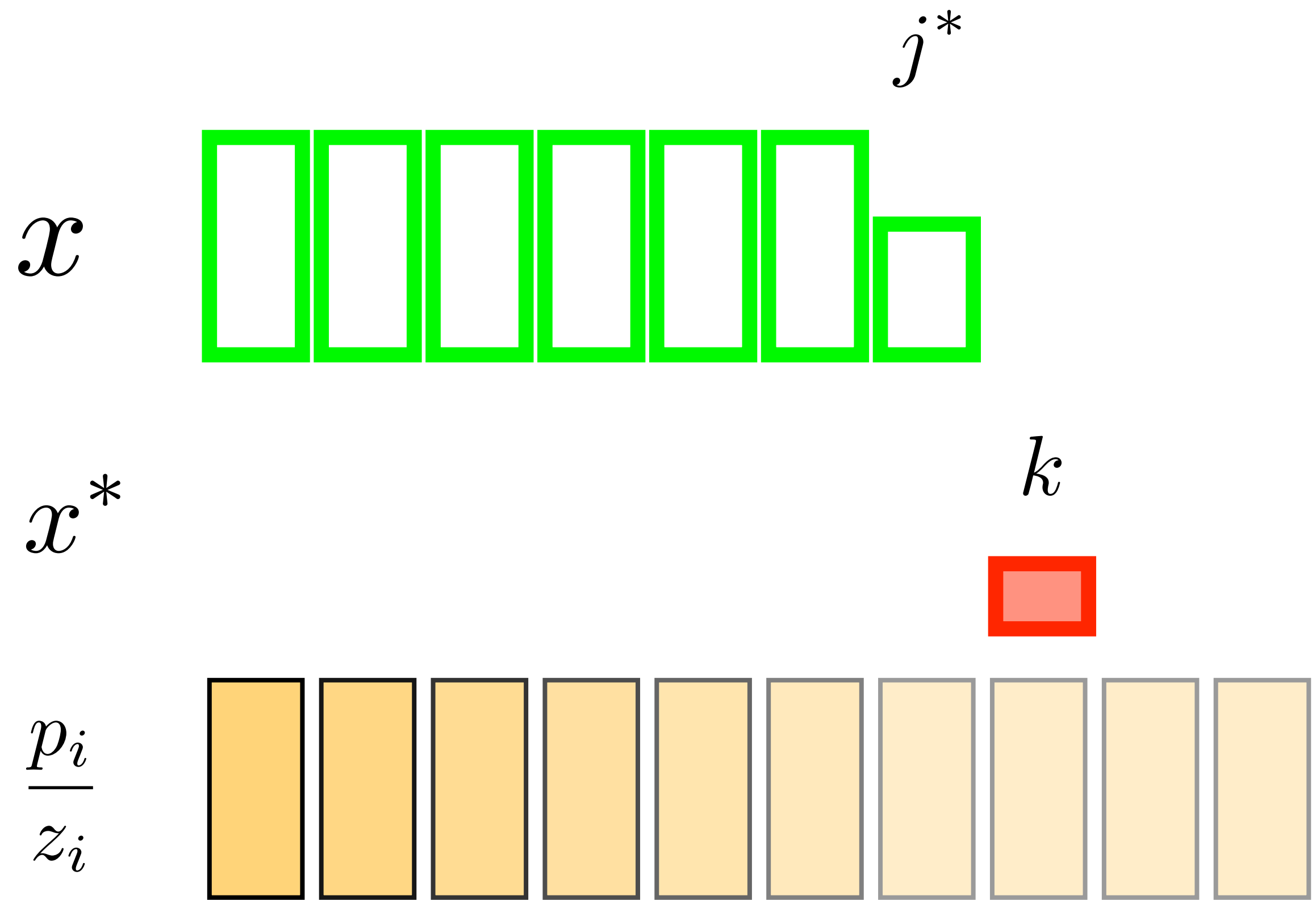
(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

$$p_i > 0 :$$

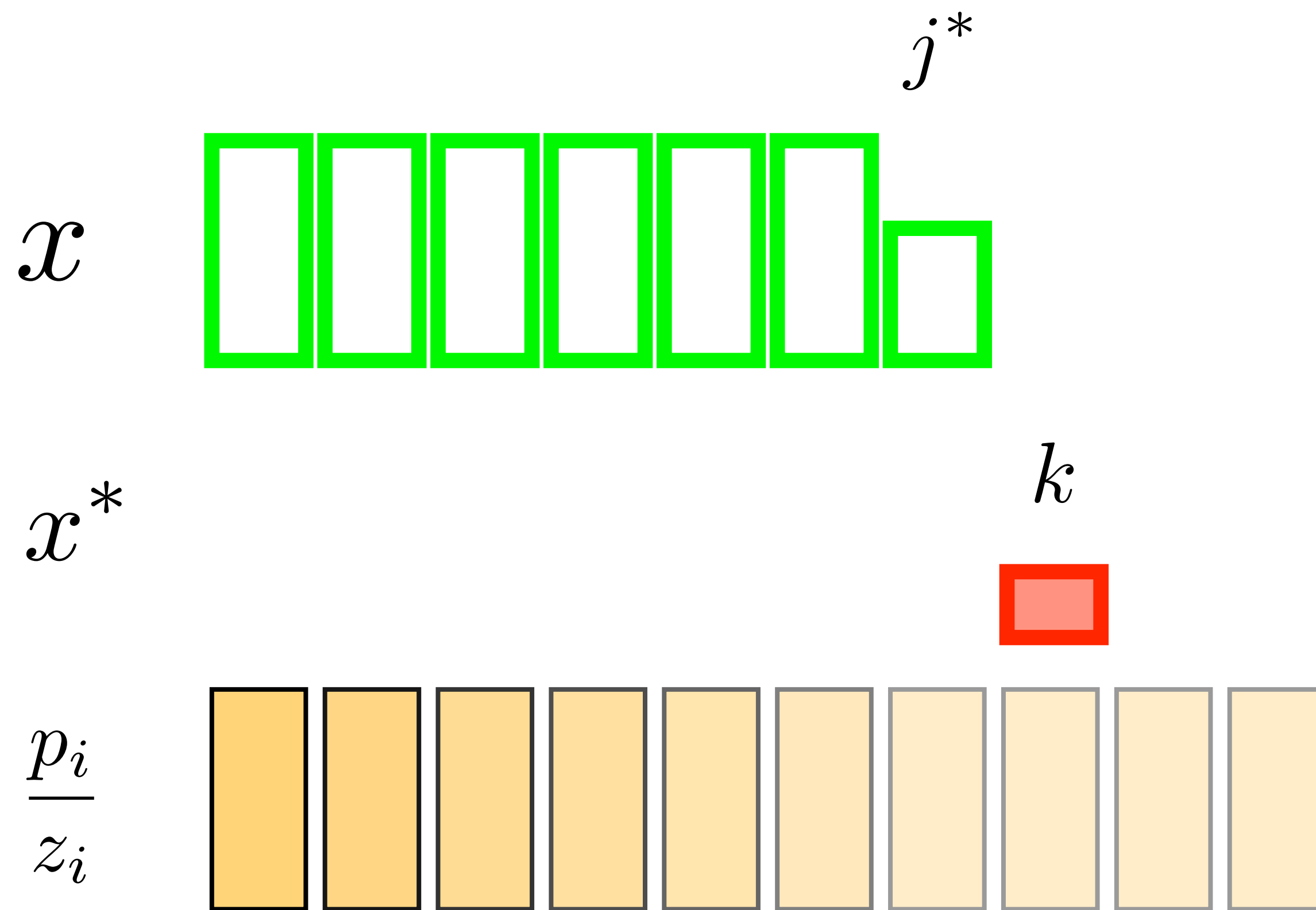
(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

$$p_i > 0 :$$

(2) Optimalität

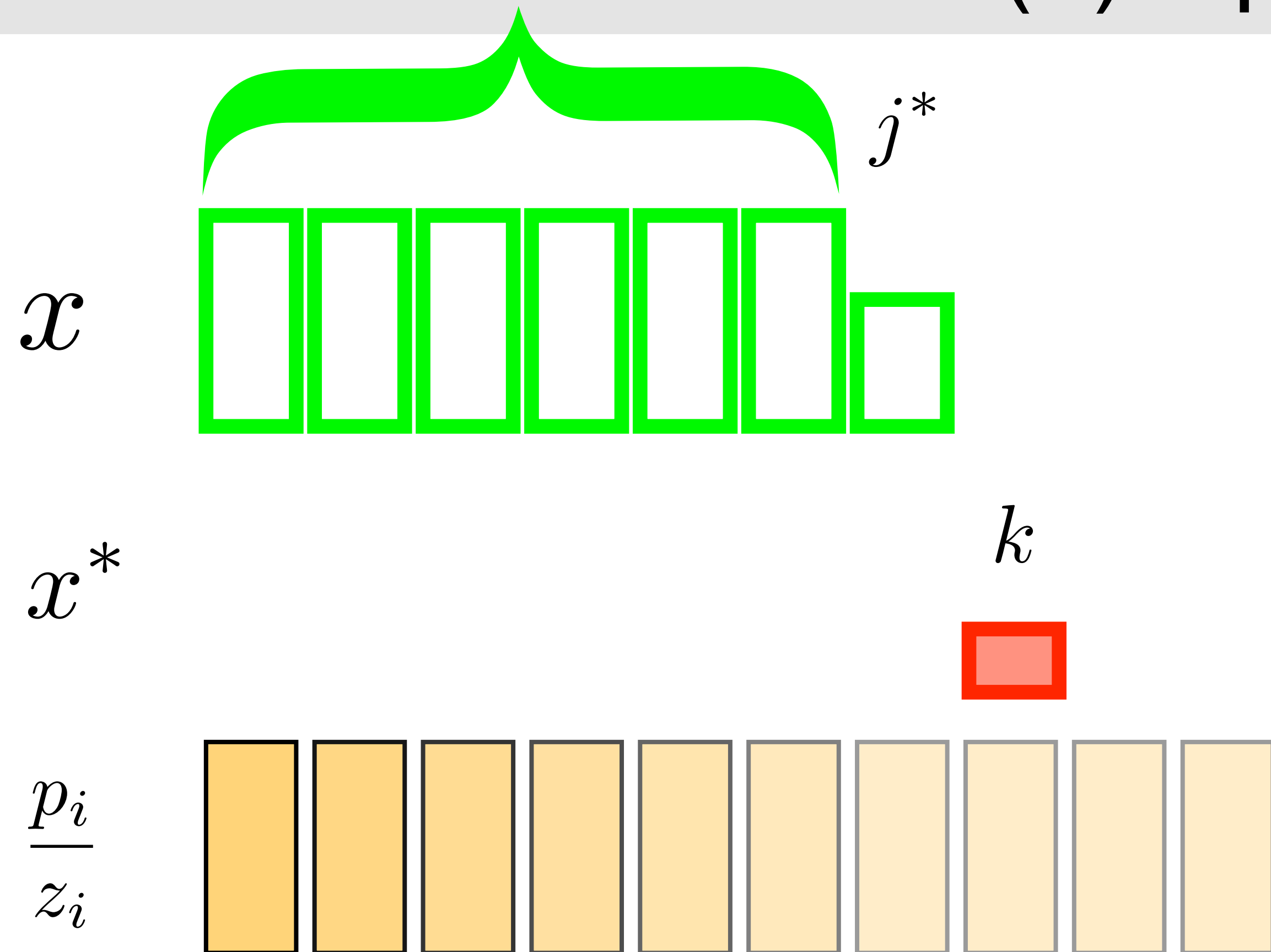


$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

(2) Optimalität

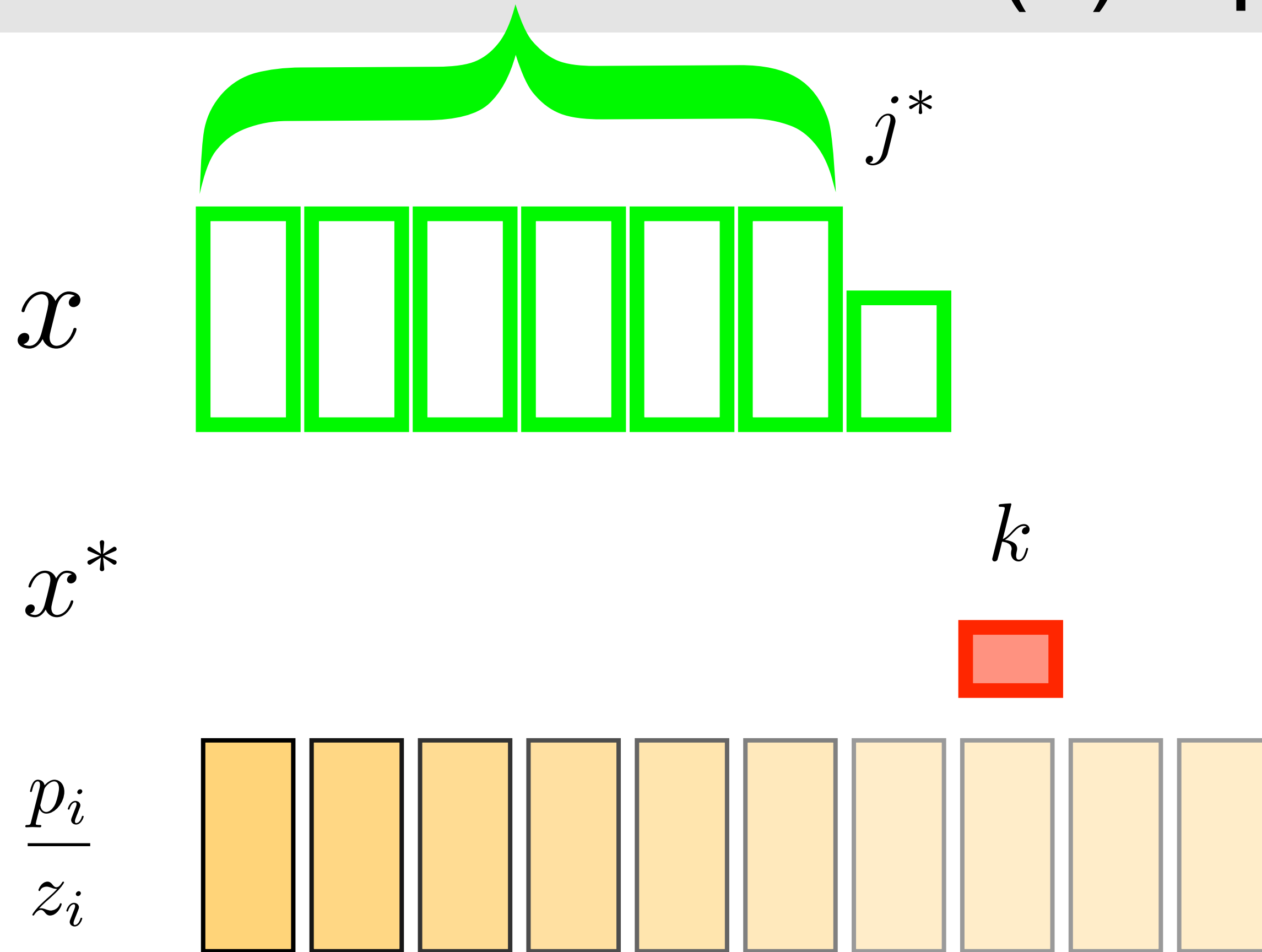


$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

(2) Optimalität



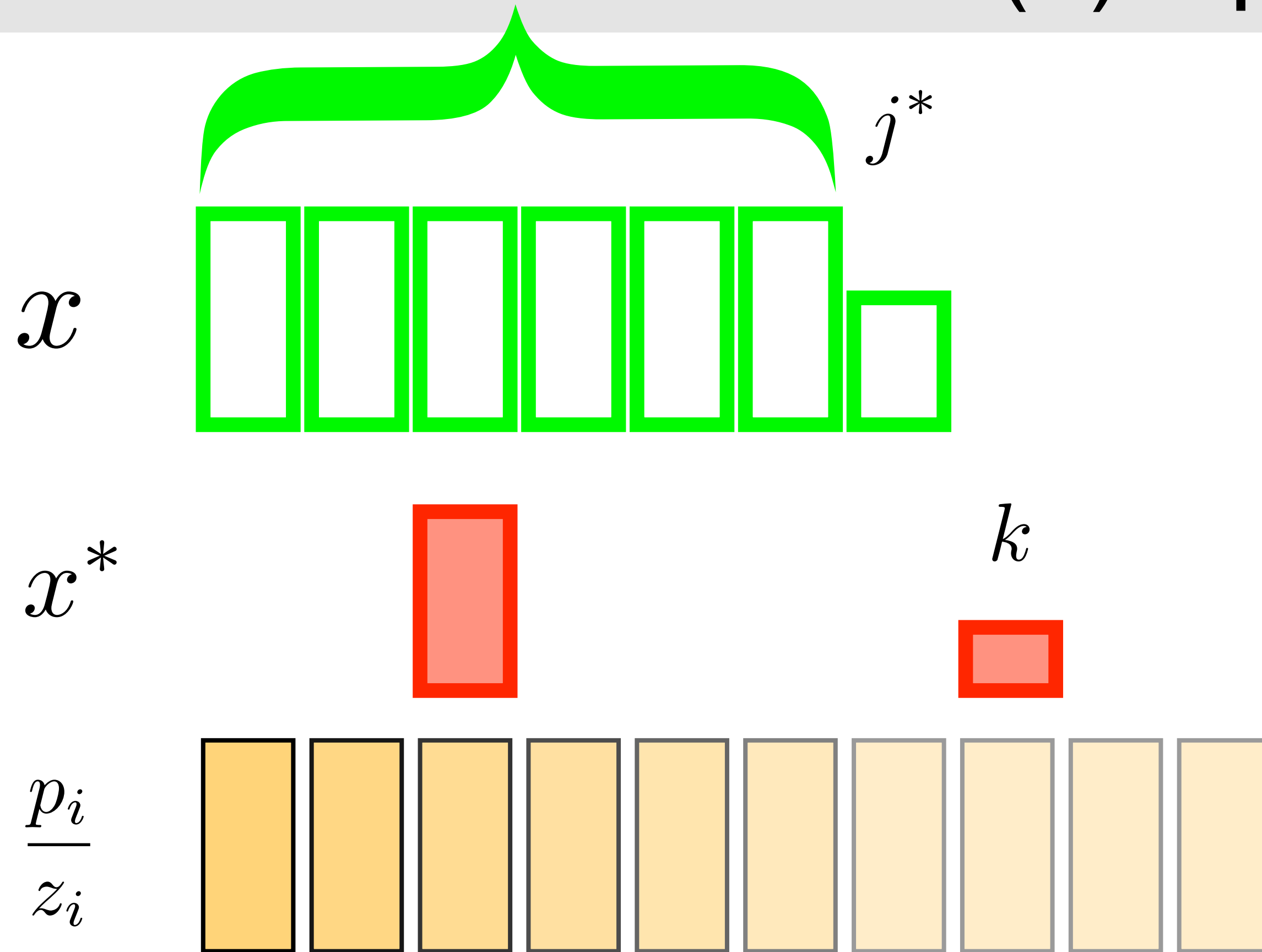
$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

(2) Optimalität



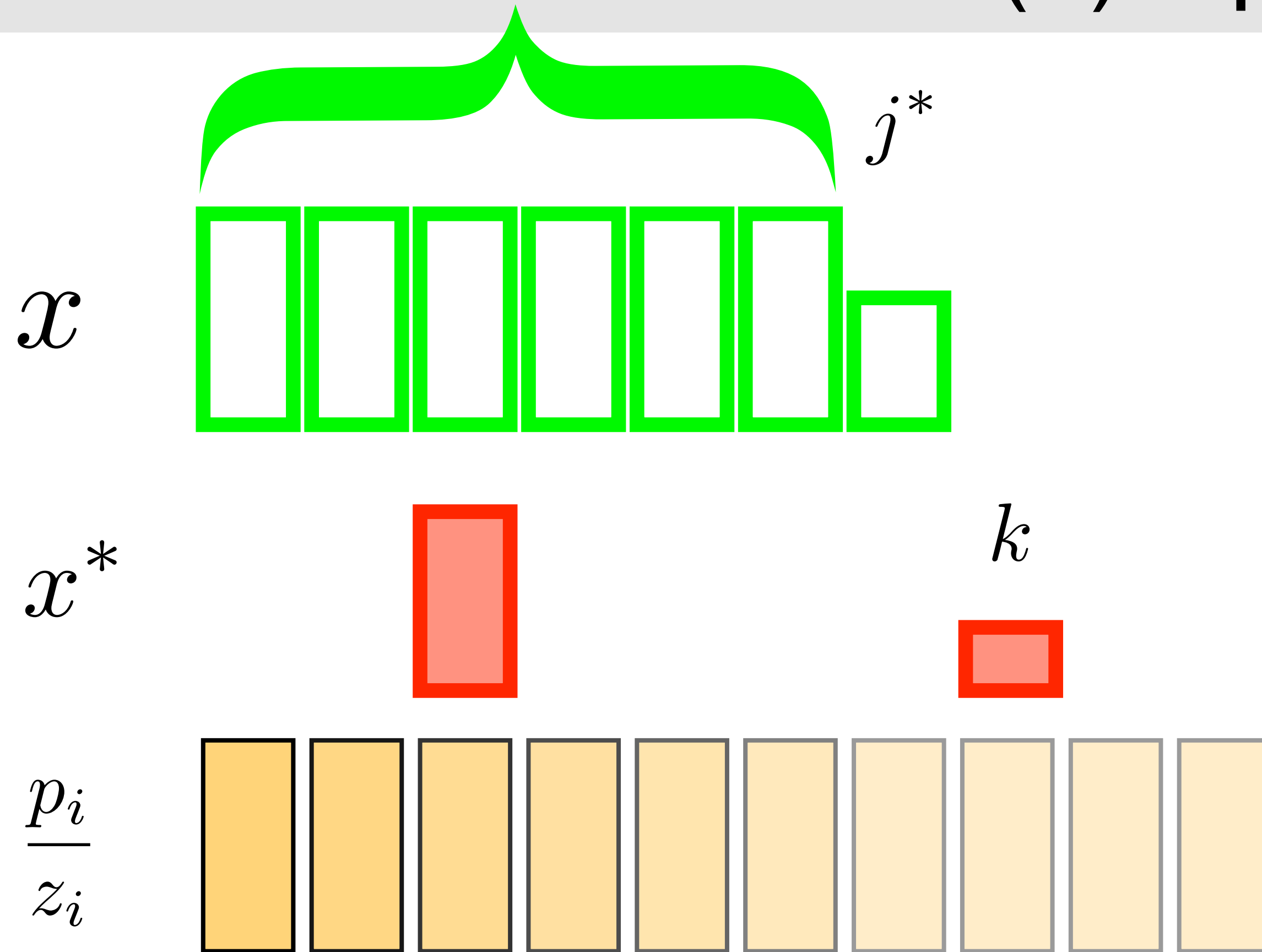
$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

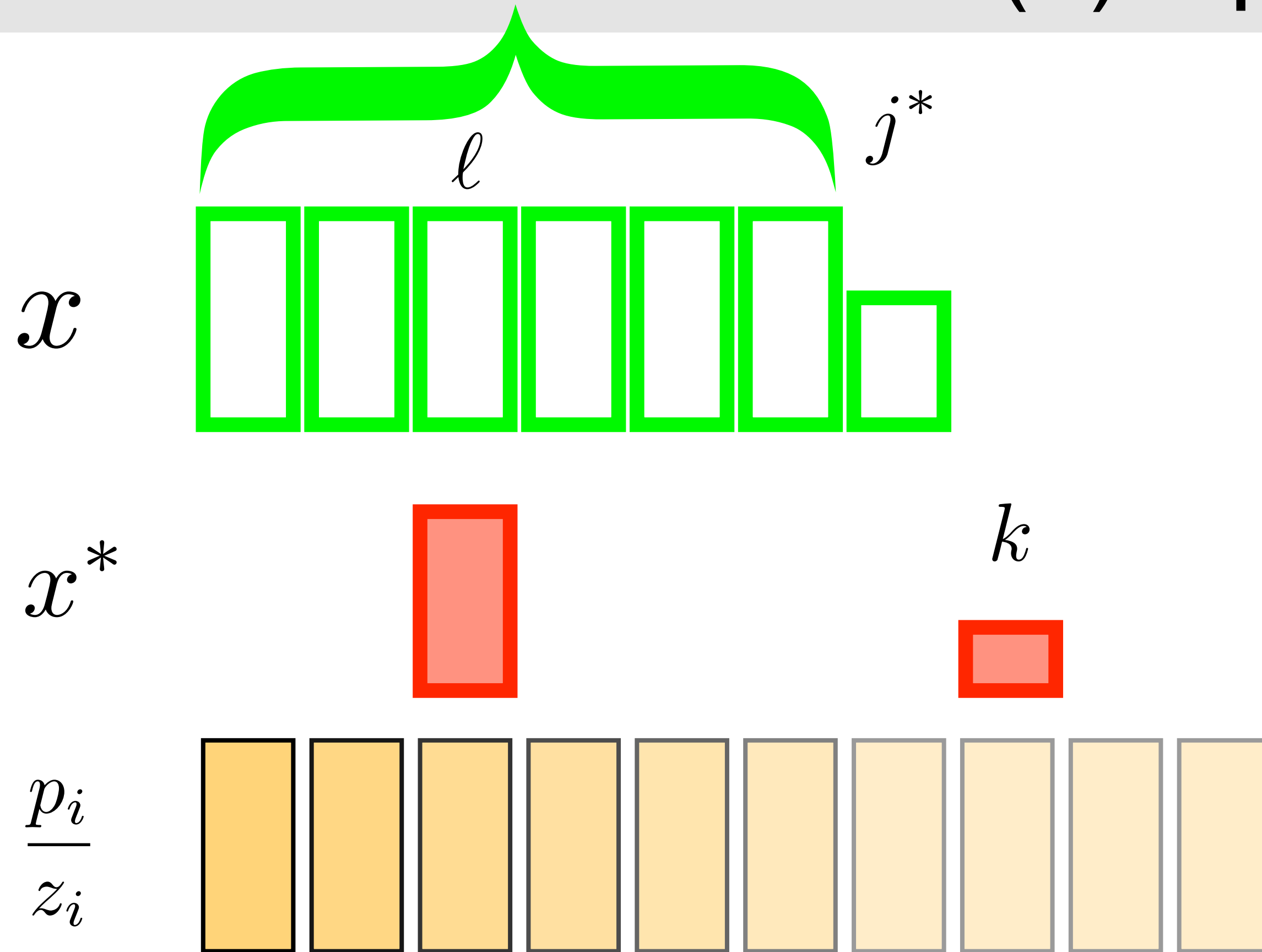
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists \ell \in \{1, \dots, j^*\} : x_{\pi(\ell)} > x_{\pi(\ell)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

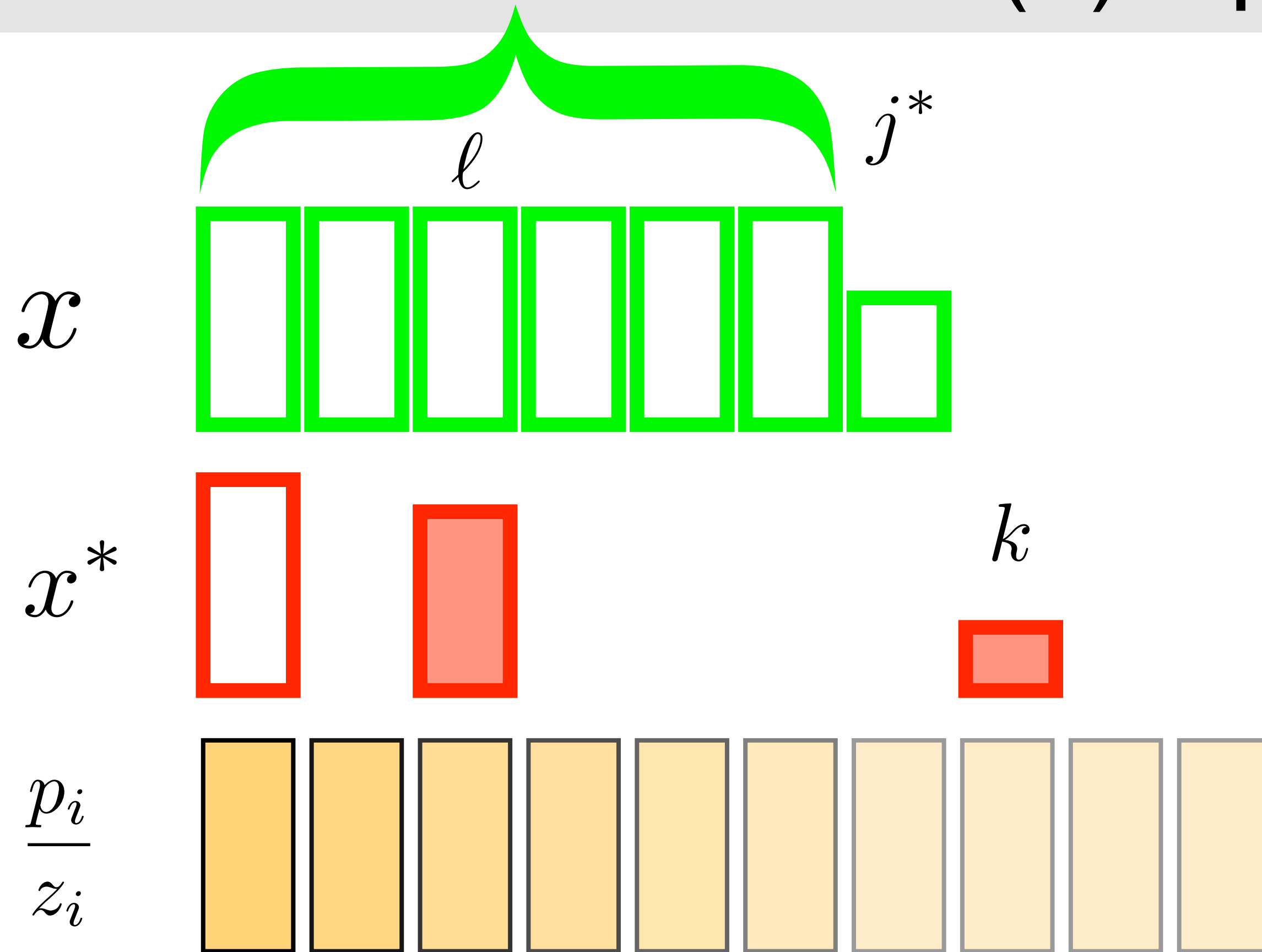
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists l \in \{1, \dots, j^*\} : x_{\pi(l)} > x_{\pi(l)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

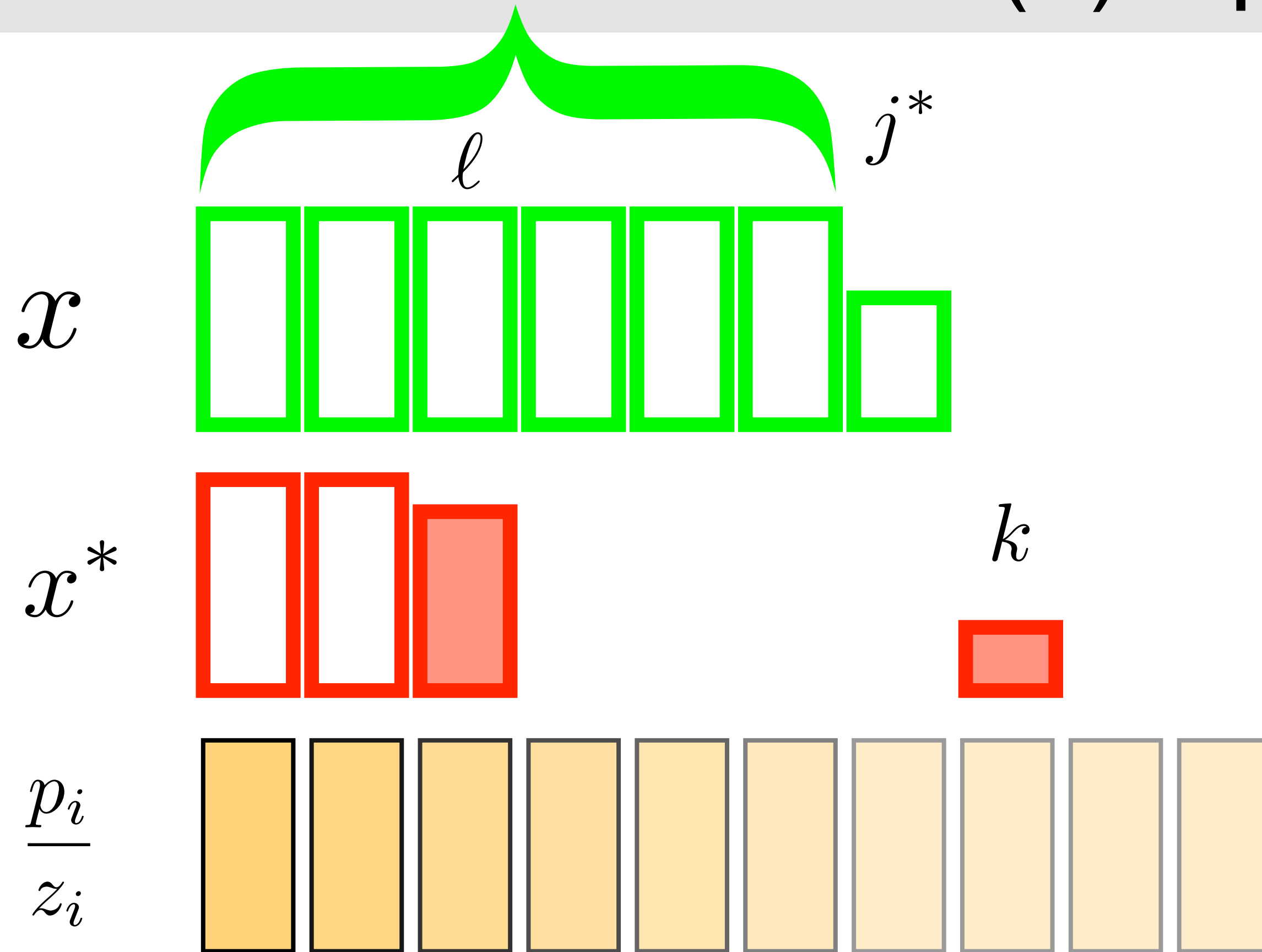
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists \ell \in \{1, \dots, j^*\} : x_{\pi(\ell)} > x_{\pi(\ell)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

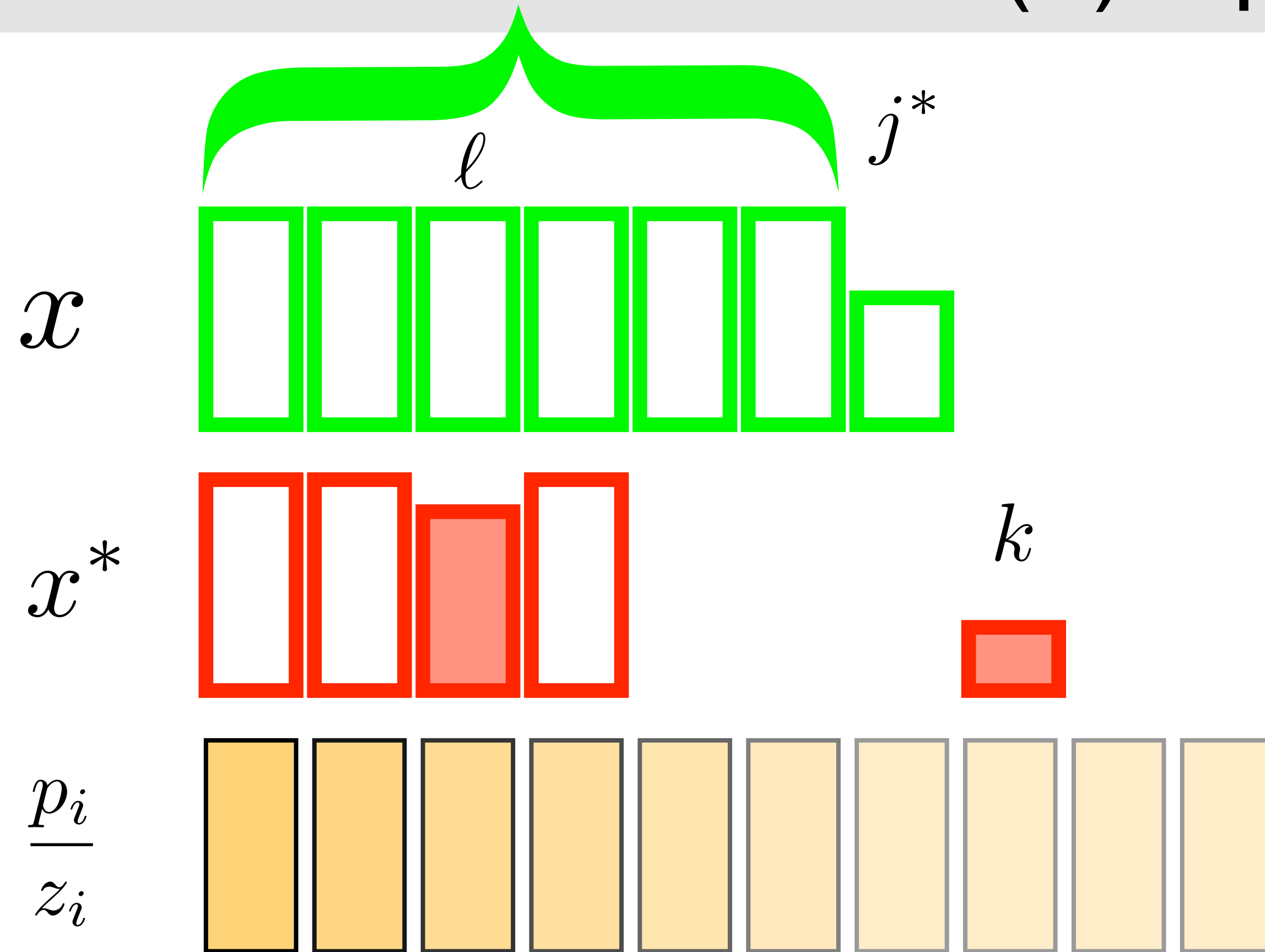
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists \ell \in \{1, \dots, j^*\} : x_{\pi(\ell)} > x_{\pi(\ell)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

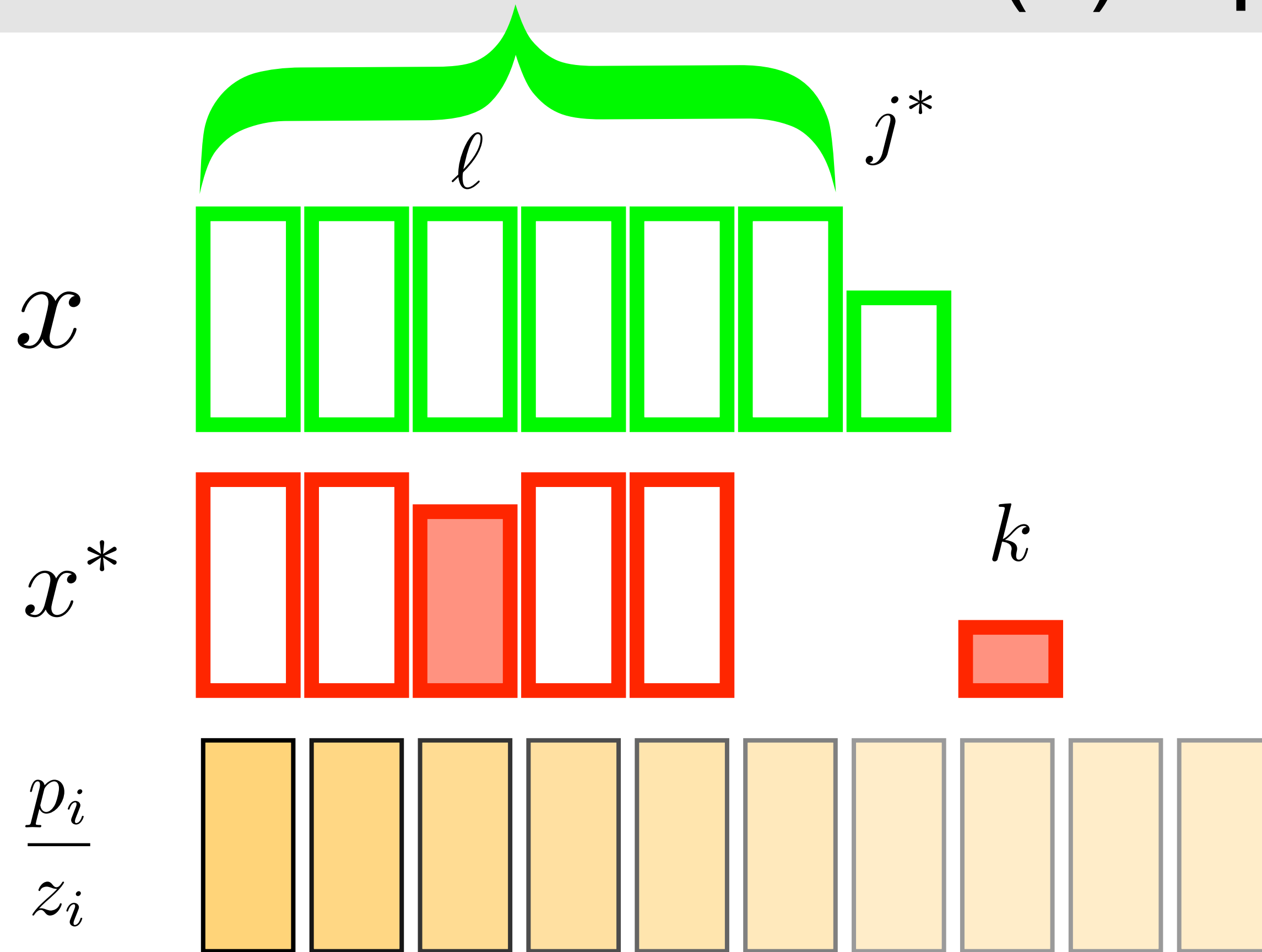
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists l \in \{1, \dots, j^*\} : x_{\pi(l)} > x_{\pi(l)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

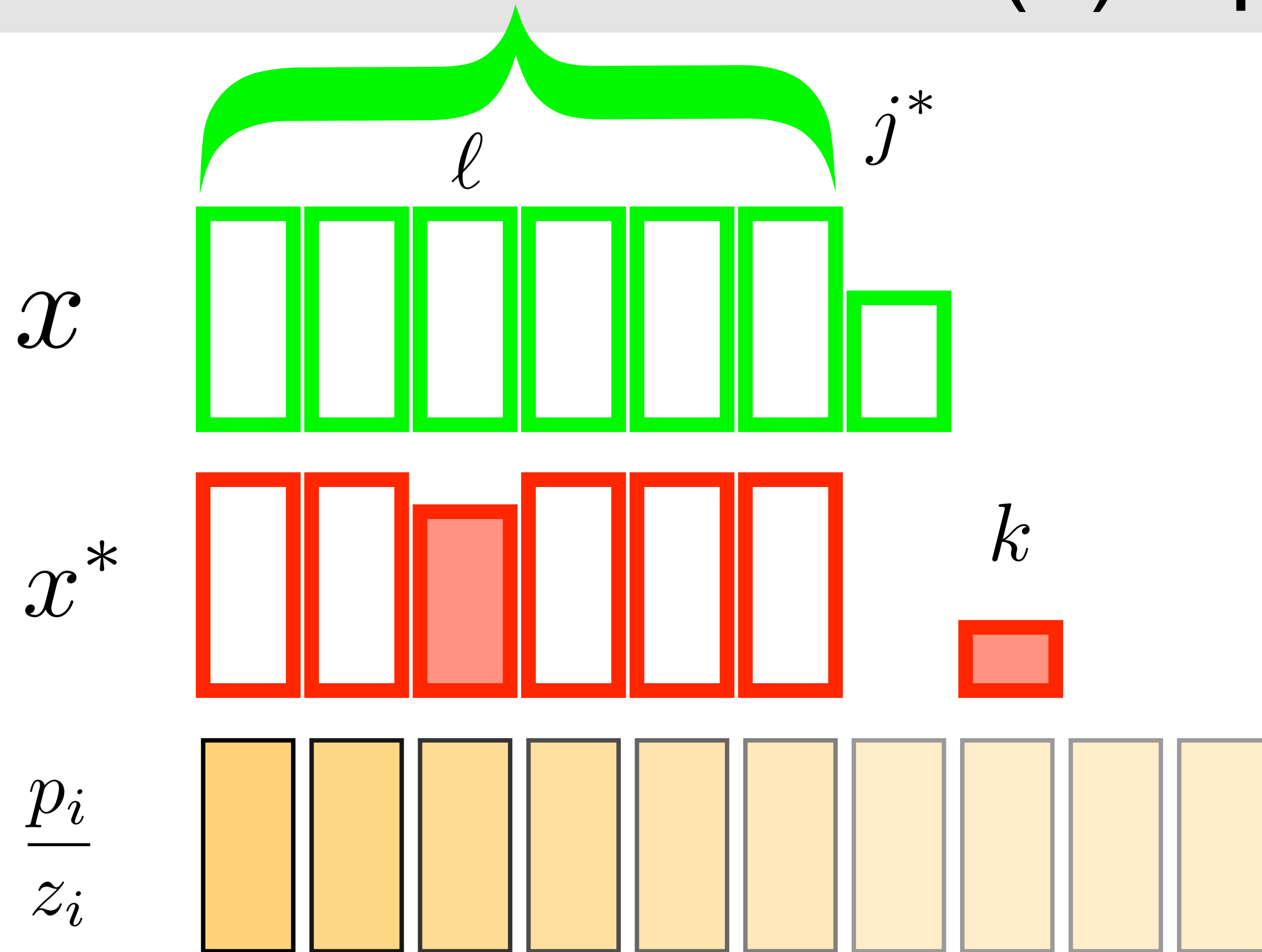
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists l \in \{1, \dots, j^*\} : x_{\pi(l)} > x_{\pi(l)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

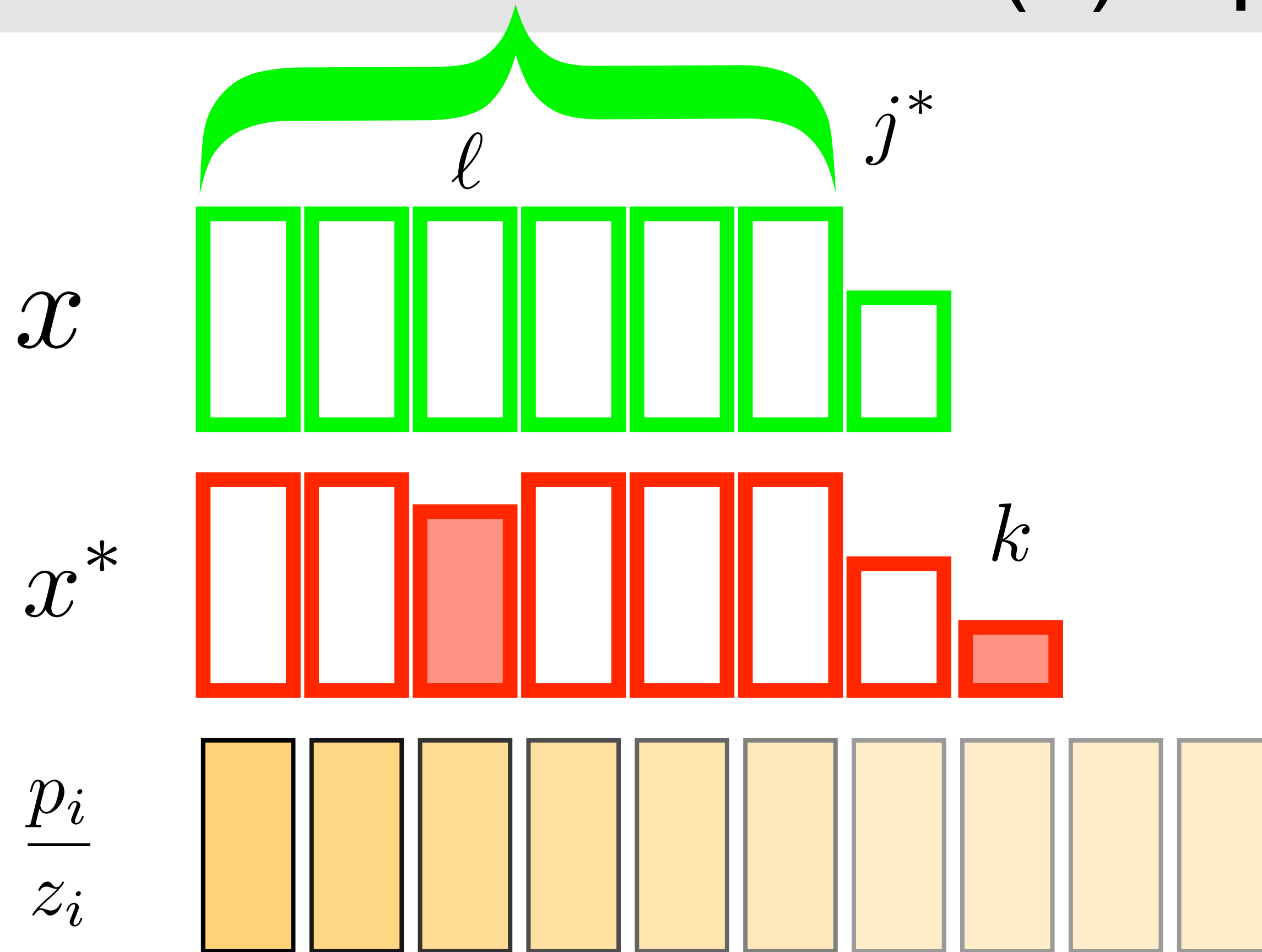
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists \ell \in \{1, \dots, j^*\} : x_{\pi(\ell)} > x_{\pi(\ell)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

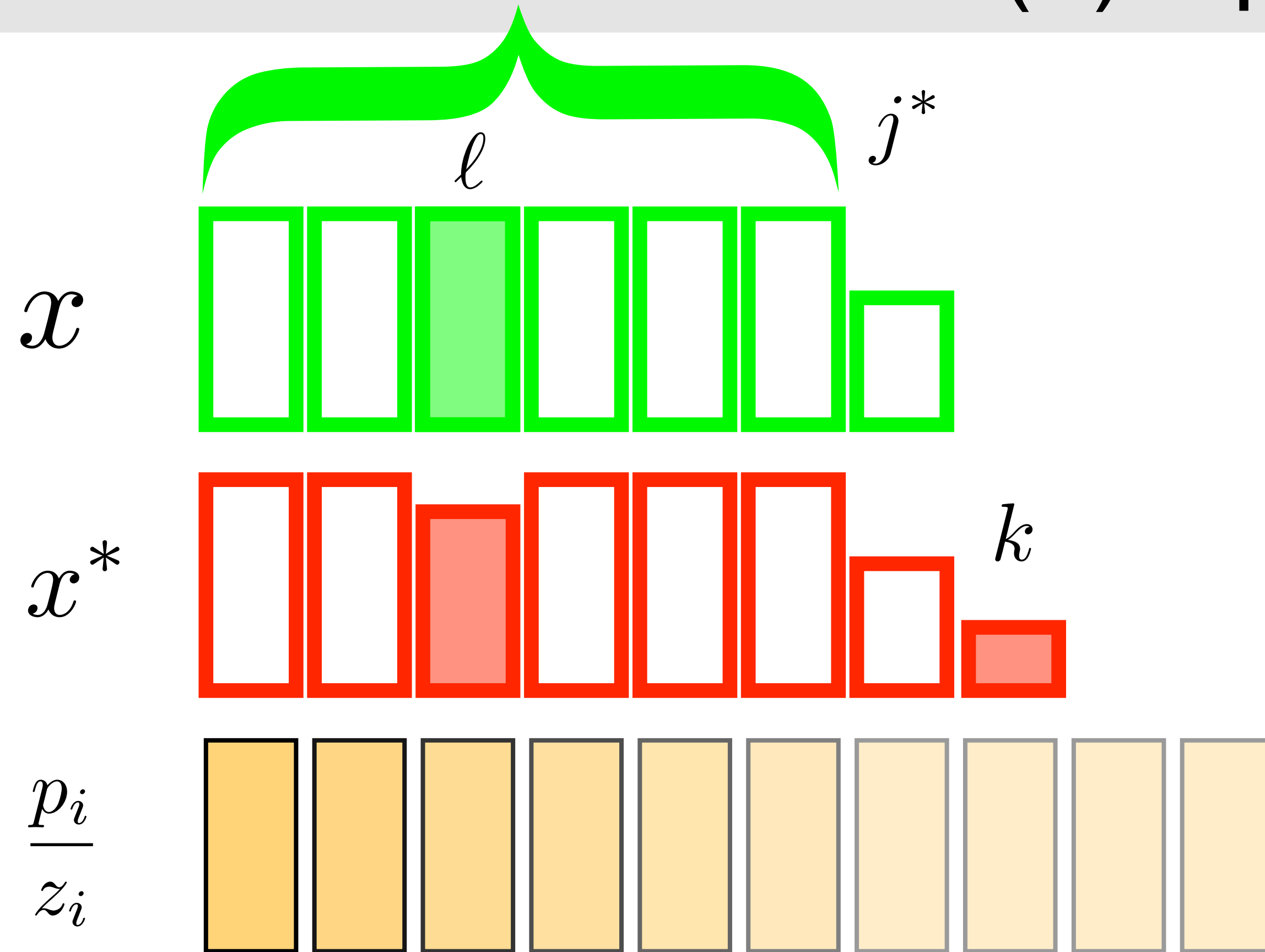
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists \ell \in \{1, \dots, j^*\} : x_{\pi(\ell)} > x_{\pi(\ell)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

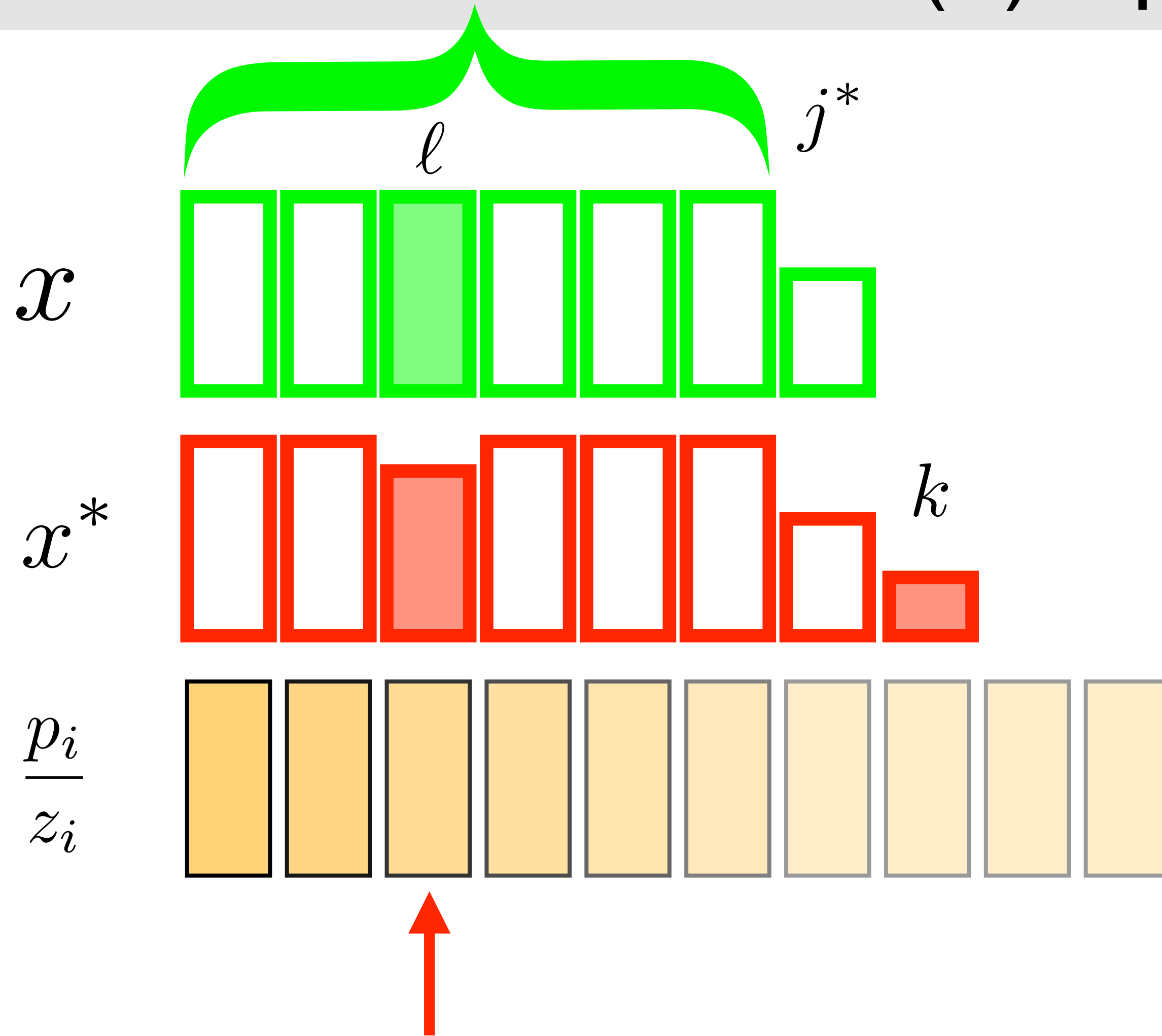
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists \ell \in \{1, \dots, j^*\} : x_{\pi(\ell)} > x_{\pi(\ell)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

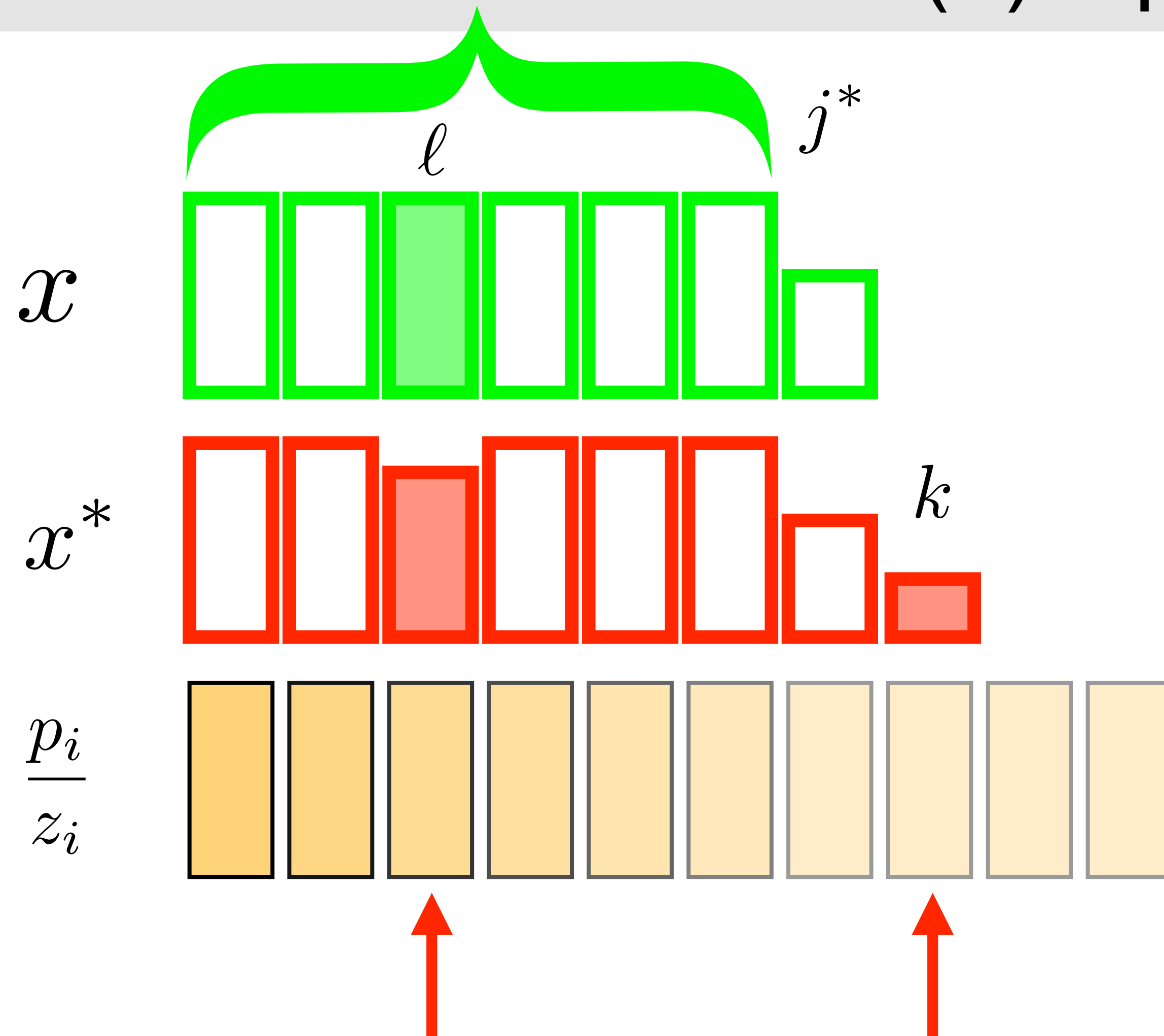
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists l \in \{1, \dots, j^*\} : x_{\pi(l)} > x_{\pi(l)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

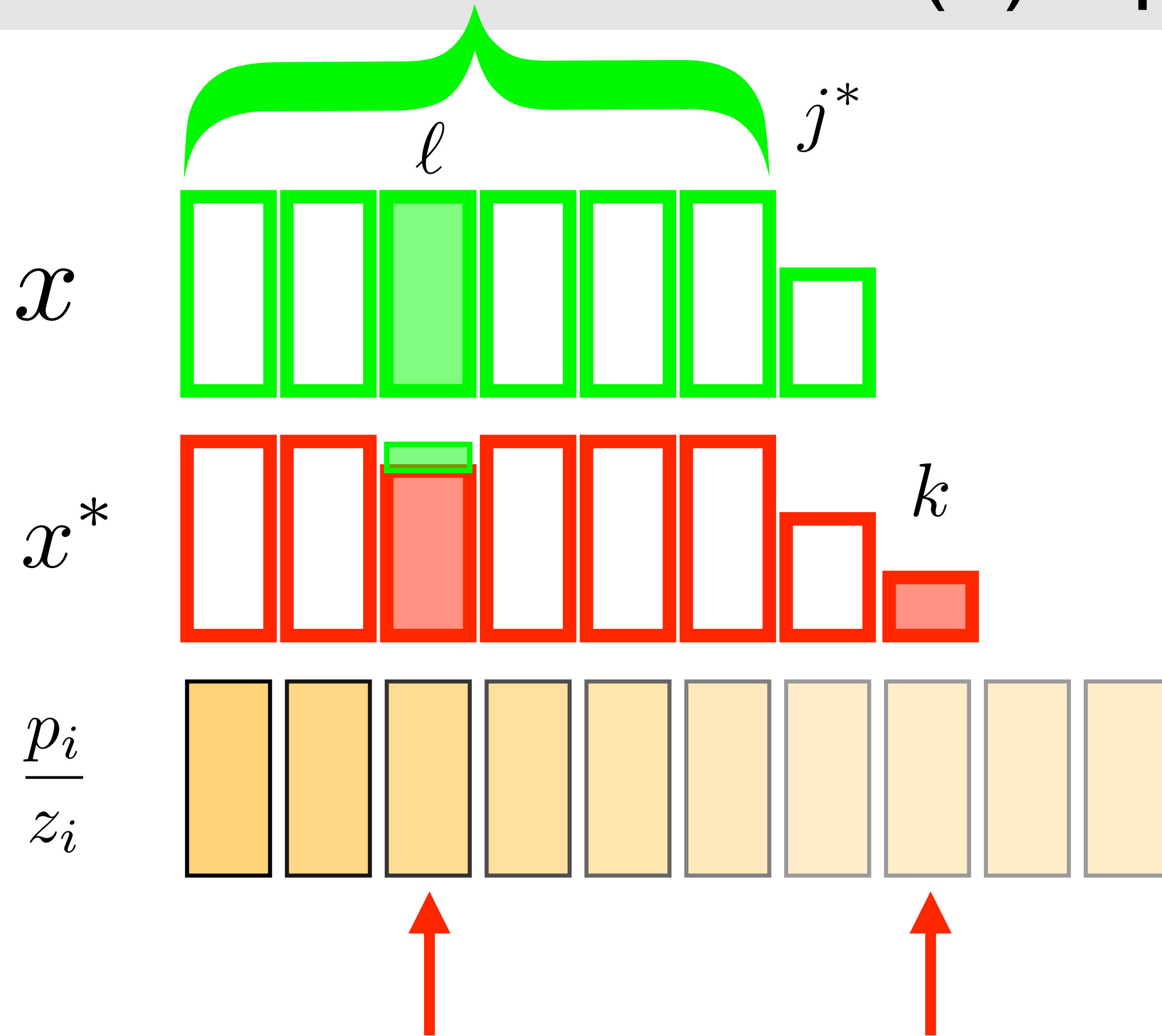
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists l \in \{1, \dots, j^*\} : x_{\pi(l)} > x_{\pi(l)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

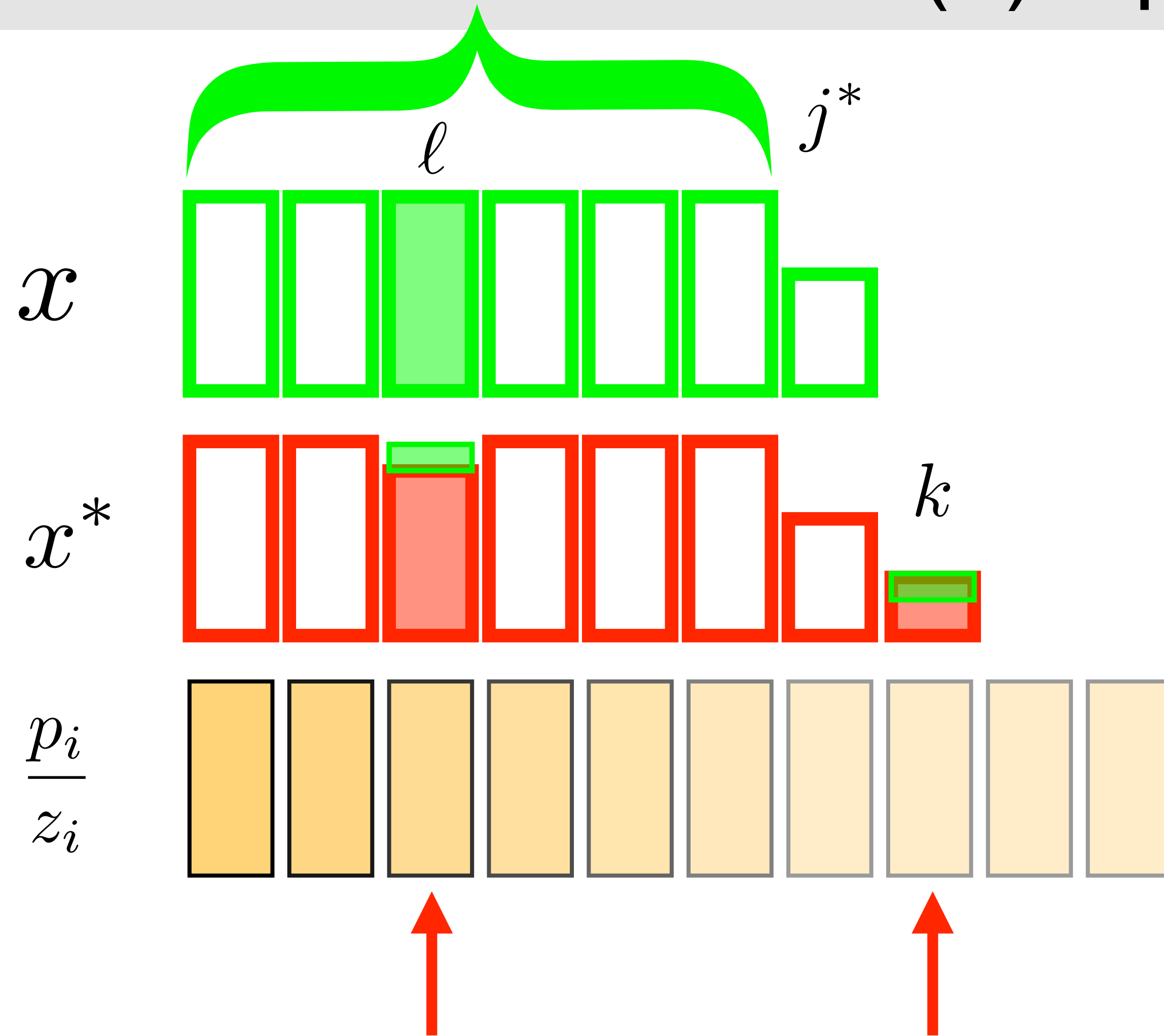
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists l \in \{1, \dots, j^*\} : x_{\pi(l)} > x_{\pi(l)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

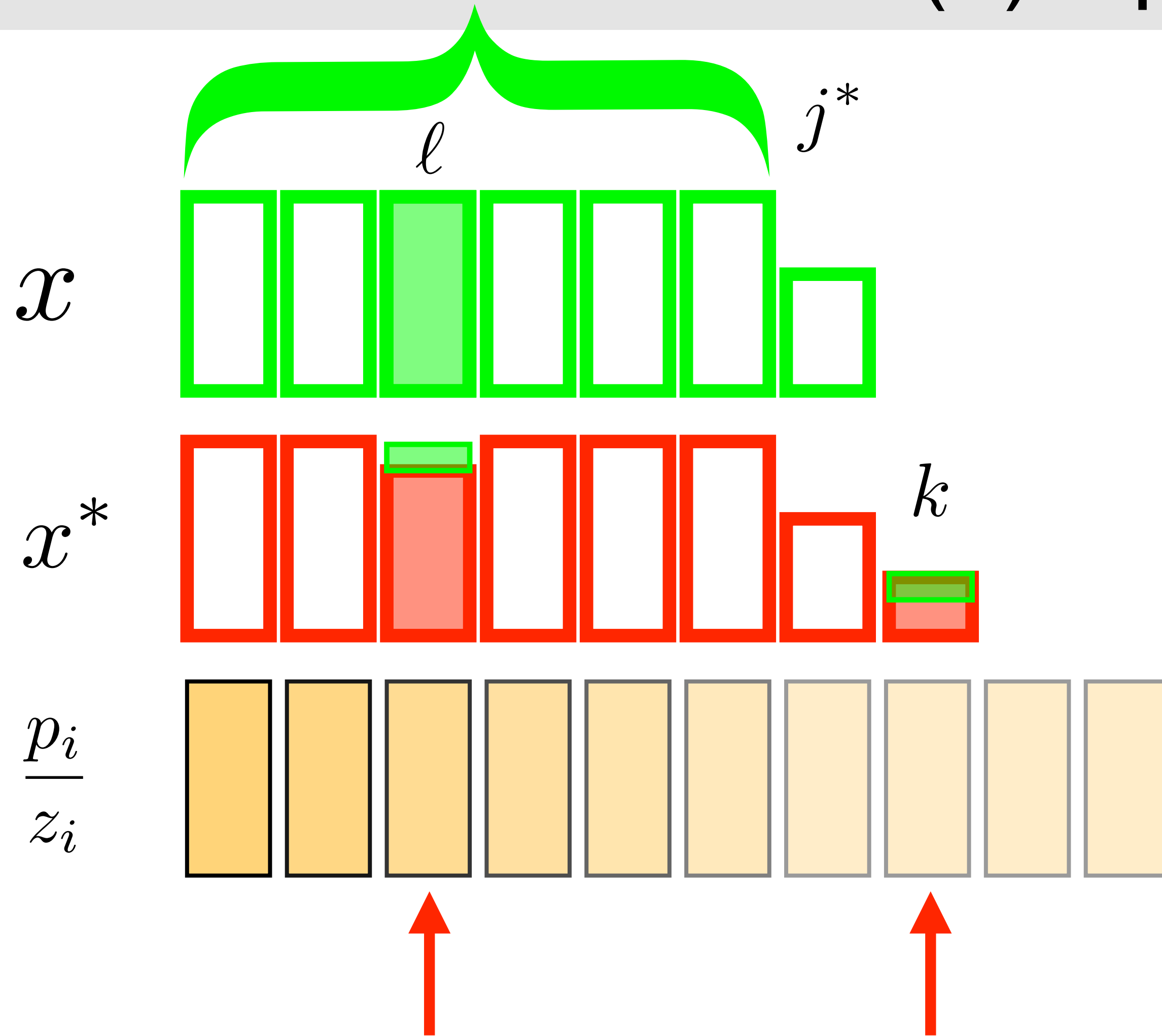
$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists l \in \{1, \dots, j^*\} : x_{\pi(l)} > x_{\pi(l)}^*$$

(2) Optimalität



$$\sum_{i=1}^n x_i p_i < \sum_{i=1}^n x_i^* p_i$$

$$p_i > 0 :$$

$$\implies \exists k \in \{1, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists k \in \{j^*, \dots, n\} : x_{\pi(k)} < x_{\pi(k)}^*$$

$$\implies \exists l \in \{1, \dots, j^*\} : x_{\pi(l)} > x_{\pi(l)}^*$$

x^* nicht optimal!

Verallgemeinerung: Mehrfachverwendung

Verallgemeinerung: Mehrfachverwendung

Problem 1.6 (INTEGER KNAPSACK).

Verallgemeinerung: Mehrfachverwendung

Problem 1.6 (INTEGER KNAPSACK).

Gegeben:

Verallgemeinerung: Mehrfachverwendung

Problem 1.6 (INTEGER KNAPSACK).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i Gewinn p_i

Verallgemeinerung: Mehrfachverwendung

Problem 1.6 (INTEGER KNAPSACK).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i Gewinn p_i
- Größenschranke Z

Verallgemeinerung: Mehrfachverwendung

Problem 1.6 (INTEGER KNAPSACK).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i Gewinn p_i
- Größenschranke Z

Gesucht:

Verallgemeinerung: Mehrfachverwendung

Problem 1.6 (INTEGER KNAPSACK).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i Gewinn p_i
- Größenschranke Z

Gesucht:

Für jedes Objekt ein Wert

Verallgemeinerung: Mehrfachverwendung

Problem 1.6 (INTEGER KNAPSACK).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i Gewinn p_i
- Größenschranke Z

Gesucht:

Für jedes Objekt ein Wert

$$x_i \in \mathbb{N}$$

Verallgemeinerung: Mehrfachverwendung

Problem 1.6 (INTEGER KNAPSACK).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i Gewinn p_i
- Größenschranke Z

Gesucht:

Für jedes Objekt ein Wert

$$x_i \in \mathbb{N}$$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

Verallgemeinerung: Mehrfachverwendung

Problem 1.6 (INTEGER KNAPSACK).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i Gewinn p_i
- Größenschranke Z

Gesucht:

Für jedes Objekt ein Wert

$$x_i \in \mathbb{N}$$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

Verallgemeinerung: Mehrfachverwendung

Problem 1.6 (INTEGER KNAPSACK).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i Gewinn p_i
- Größenschranke Z

Gesucht:

Für jedes Objekt ein Wert

$$x_i \in \mathbb{N}$$

mit

$$\sum_{i=1}^n z_i x_i \leq Z$$

und

$$\sum_{i=1}^n p_i x_i = \text{Maximal}$$

Spezialfall: Alle Gewinndichten gleich

Spezialfall: Alle Gewinndichten gleich

Problem 1.7.

Spezialfall: Alle Gewinndichten gleich

Problem 1.7.

Gegeben:

Spezialfall: Alle Gewinndichten gleich

Problem 1.7.

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i

Spezialfall: Alle Gewinndichten gleich

Problem 1.7.

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i
- Größenschränke Z

Spezialfall: Alle Gewinndichten gleich

Problem 1.7.

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i
- Größenschranke Z

Spezialfall: Alle Gewinndichten gleich

Problem 1.7.

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i
- Größenschranke Z

Gesucht:

Spezialfall: Alle Gewinndichten gleich

Problem 1.7.

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i
- Größenschranke Z

Gesucht:

Eine Menge

$$S \subseteq \{1, \dots, n\}$$

Spezialfall: Alle Gewinndichten gleich

Problem 1.7.

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i
- Größenschranke Z

Gesucht:

Eine Menge

$$S \subseteq \{1, \dots, n\}$$

mit

$$\sum_{i \in S} z_i \leq Z$$

Spezialfall: Alle Gewinndichten gleich

Problem 1.7.

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i
- Größenschranke Z

Gesucht:

Eine Menge

$$S \subseteq \{1, \dots, n\}$$

mit

$$\sum_{i \in S} z_i \leq Z$$

und

$$\sum_{i \in S} z_i = \textit{Maximal}$$

Spezialfall des Spezialfalls: Größenschränke treffen

Spezialfall des Spezialfalls: Größenschränke treffen

Problem 1.8 (SUBSET SUM).

Spezialfall des Spezialfalls: Größenschränke treffen

Problem 1.8 (SUBSET SUM).

Gegeben:

Spezialfall des Spezialfalls: Größenschränke treffen

Problem 1.8 (SUBSET SUM).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i

Spezialfall des Spezialfalls: Größenschränke treffen

Problem 1.8 (SUBSET SUM).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i
- Zielgröße Z

Spezialfall des Spezialfalls: Größenschränke treffen

Problem 1.8 (SUBSET SUM).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i
- Zielgröße Z

Spezialfall des Spezialfalls: Größenschränke treffen

Problem 1.8 (SUBSET SUM).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i
- Zielgröße Z

Gesucht:

Spezialfall des Spezialfalls: Größenschränke treffen

Problem 1.8 (SUBSET SUM).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i
- Zielgröße Z

Gesucht:

Eine Menge

$$S \subseteq \{1, \dots, n\}$$

Spezialfall des Spezialfalls: Größenschränke treffen

Problem 1.8 (SUBSET SUM).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i
- Zielgröße Z

Gesucht:

Eine Menge

$$S \subseteq \{1, \dots, n\}$$

mit

$$\sum_{i \in S} z_i = Z$$

Spezialfall³: Partition

Spezialfall³: Partition

Problem 1.9 (PARTITION).

Spezialfall³: Partition

Problem 1.9 (PARTITION).

Gegeben:

Spezialfall³: Partition

Problem 1.9 (PARTITION).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i

Spezialfall³: Partition

Problem 1.9 (PARTITION).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i

Gesucht:

Spezialfall³: Partition

Problem 1.9 (PARTITION).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i

Gesucht:

Eine Menge

$$S \subseteq \{1, \dots, n\}$$

Spezialfall³: Partition

Problem 1.9 (PARTITION).

Gegeben:

- n Objekte $1, \dots, n$ mit jeweils Größe z_i

Gesucht:

Eine Menge

$$S \subseteq \{1, \dots, n\}$$

mit

$$\sum_{i \in S} z_i = \sum_{i \notin S} z_i$$

Beispiel

Beispiel

Beispiel 1.10.

Partition für $\{z_1, \dots, z_9\} = \{7, 13, 17, 20, 29, 31, 31, 35, 57\}$

Gesamtsumme: 240

Beispiel

Beispiel 1.10.

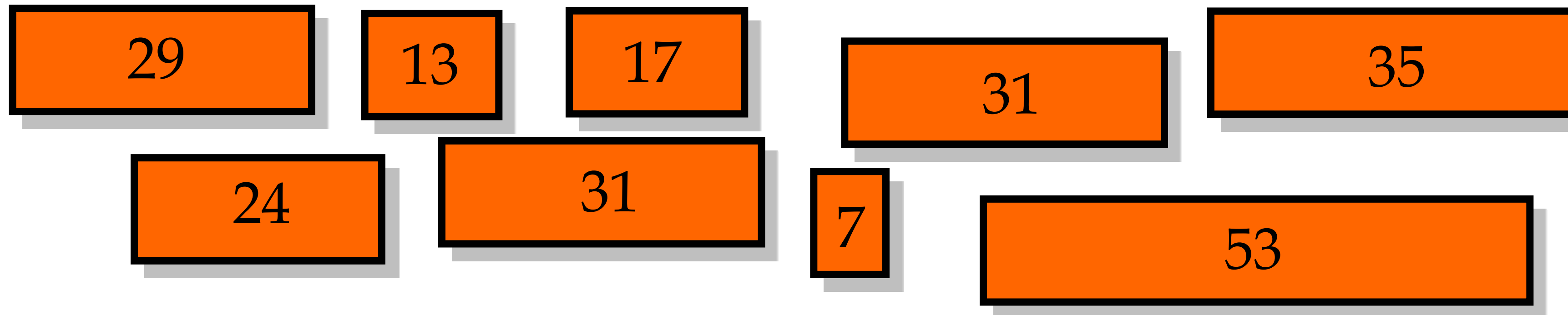
Partition für $\{z_1, \dots, z_9\} = \{7, 13, 17, 20, 29, 31, 31, 35, 57\}$

Gesamtsumme: 240

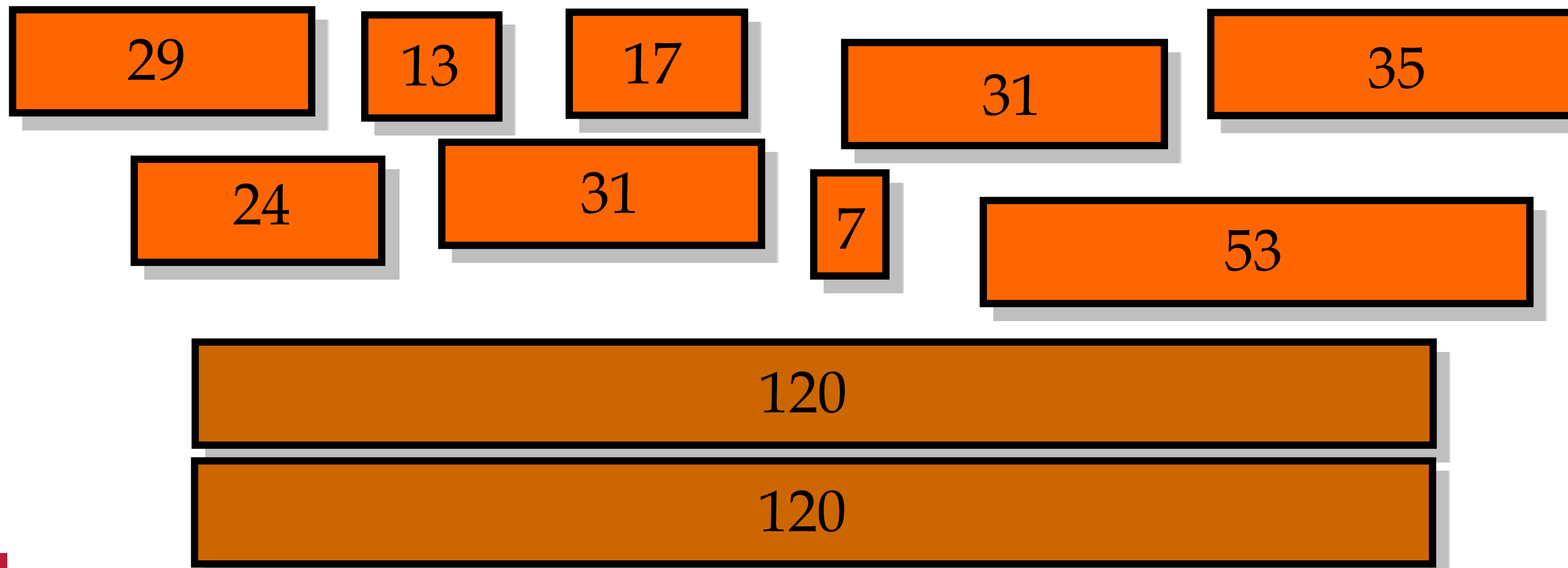
Finde Aufteilung in zwei gleiche Teilmengen!

Ein Bild hilft

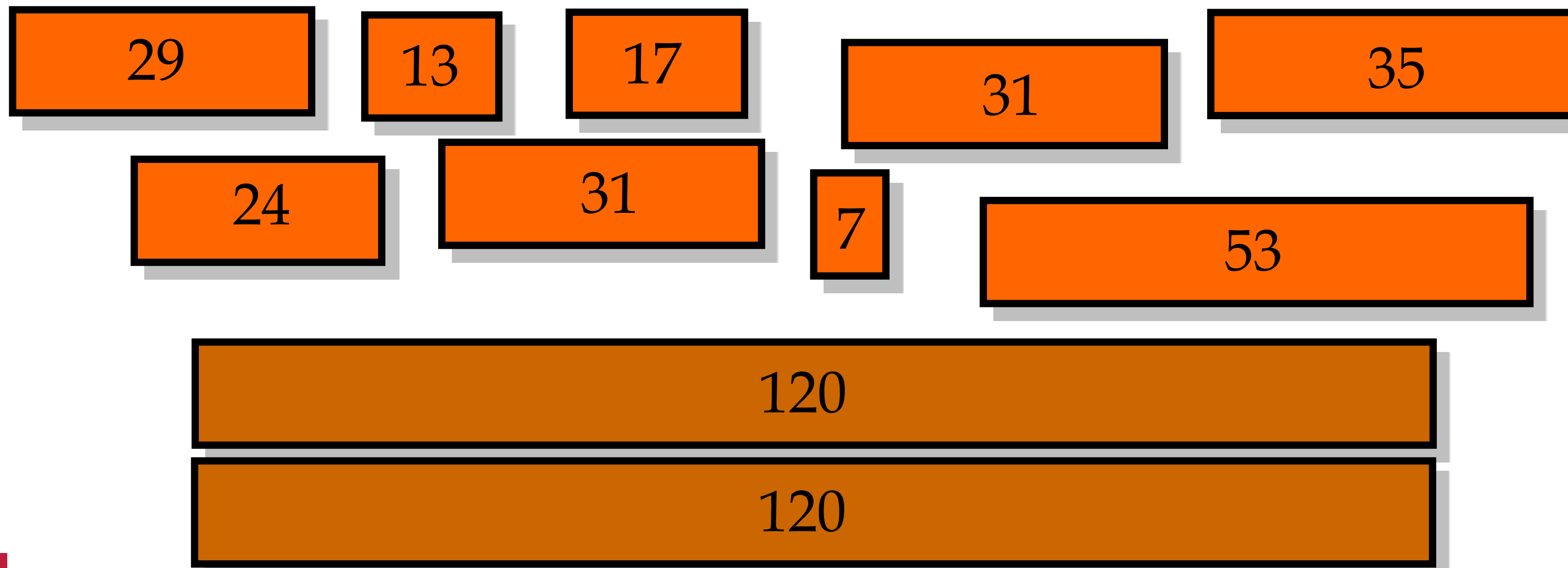
Ein Bild hilft



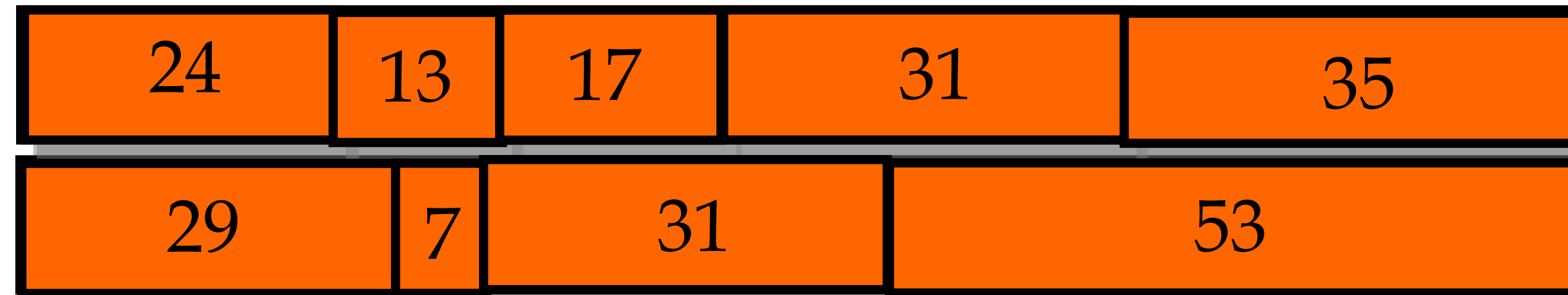
Ein Bild hilft



Ein Bild hilft



Ein Bild hilft

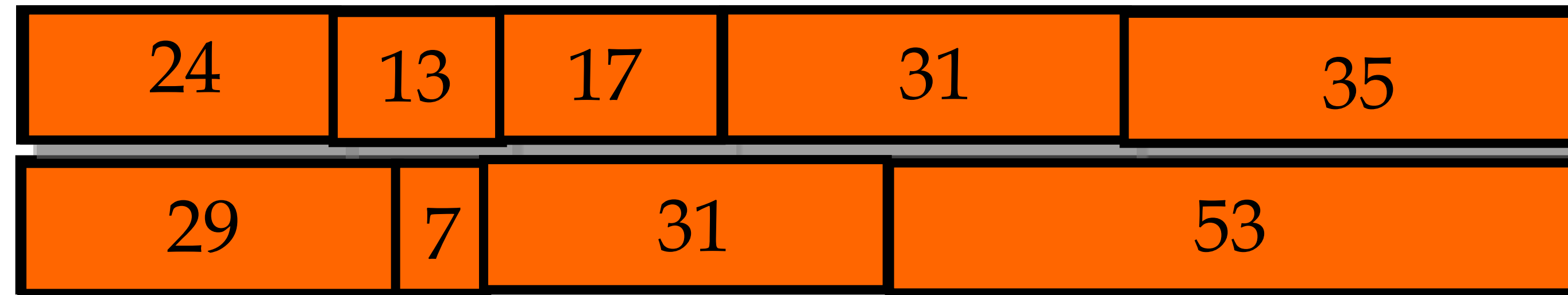


Ein Bild hilft

Beispiel 1.10.

Partition für $\{z_1, \dots, z_9\} = \{7, 13, 17, 20, 29, 31, 31, 35, 57\}$

Gesamtsumme: 240

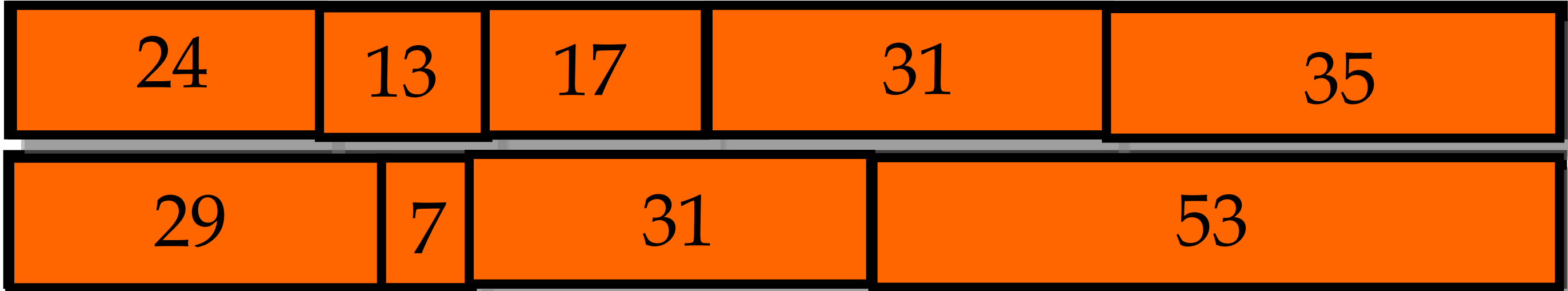


Ein Bild hilft

Beispiel 1.10.

Partition für $\{z_1, \dots, z_9\} = \{7, 13, 17, 20, 29, 31, 31, 35, 57\}$

Gesamtsumme: 240

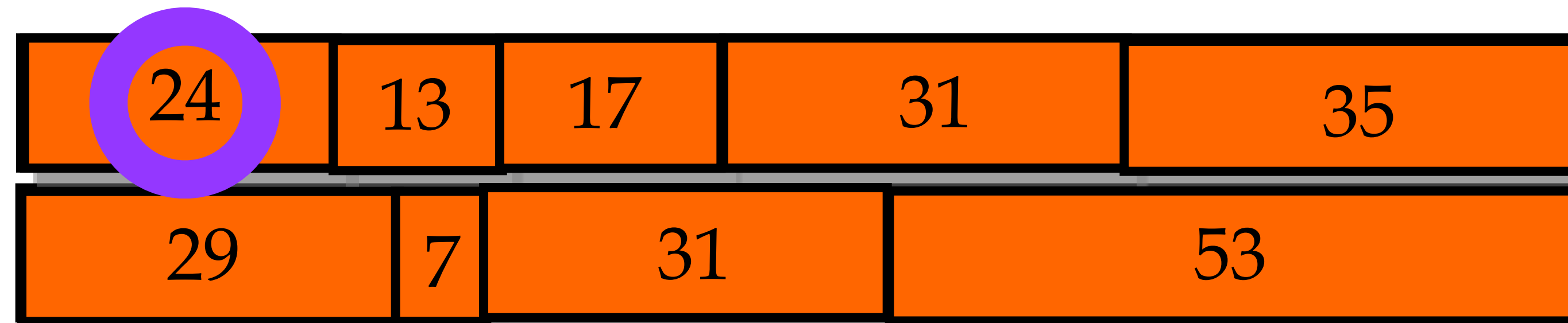


Ein Bild hilft

Beispiel 1.10.

Partition für $\{z_1, \dots, z_9\} = \{7, 13, 17, 20, 29, 31, 31, 35, 57\}$

Gesamtsumme: 240

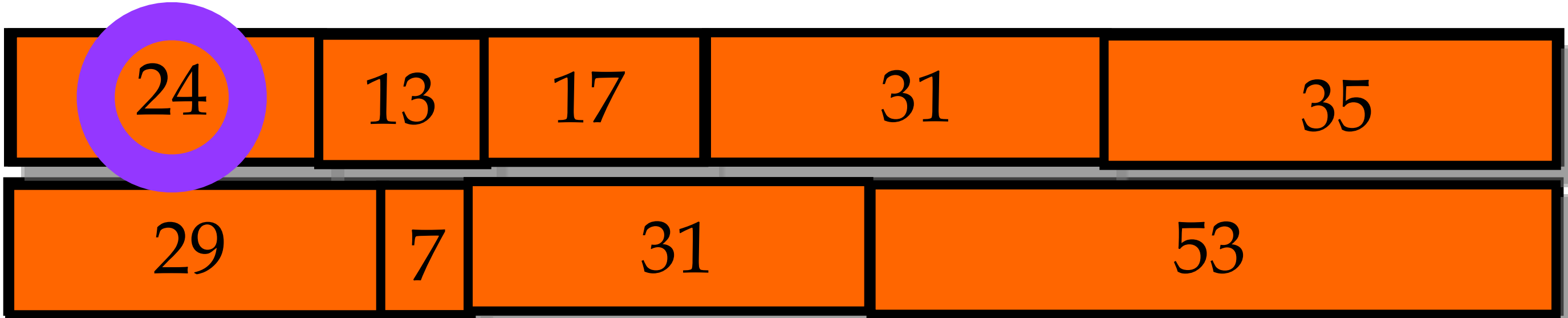


Ein Bild hilft

Beispiel 1.10.

Partition für $\{z_1, \dots, z_9\} = \{7, 13, 17, 20, 29, 31, 31, 35, 57\}$

Gesamtsumme: 240

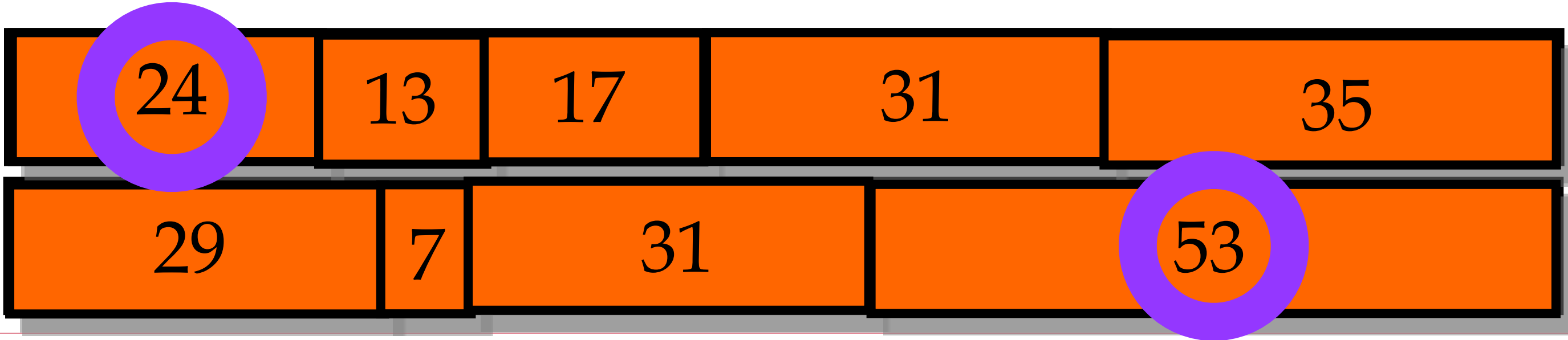


Ein Bild hilft

Beispiel 1.10.

Partition für $\{z_1, \dots, z_9\} = \{7, 13, 17, 20, 29, 31, 31, 35, 57\}$

Gesamtsumme: 240



Keine Lösung?!

Keine Lösung?!

120

120

Keine Lösung?!

120

57

120

Keine Lösung?!

120

57

35

120

Keine Lösung?!

120

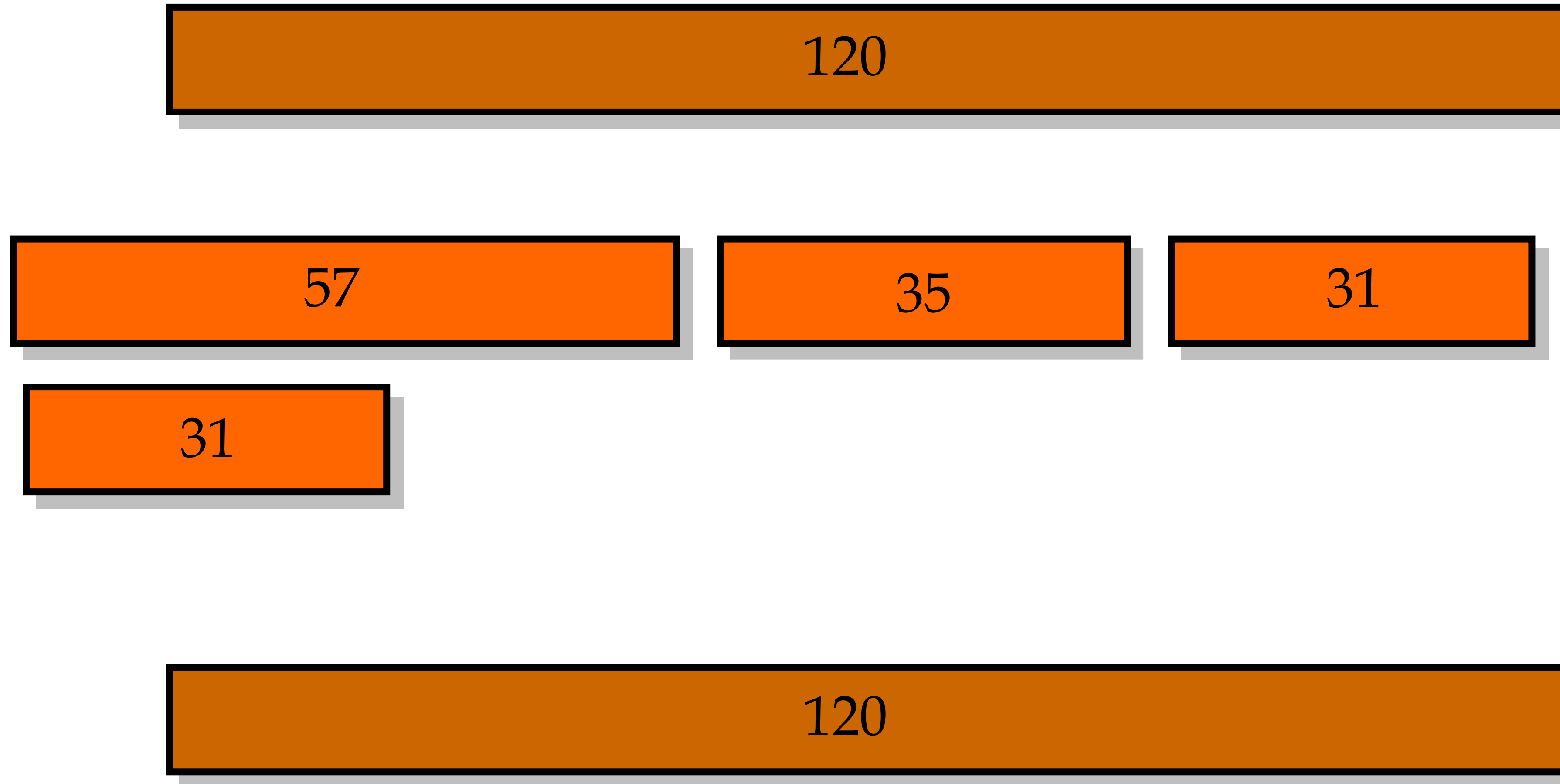
57

35

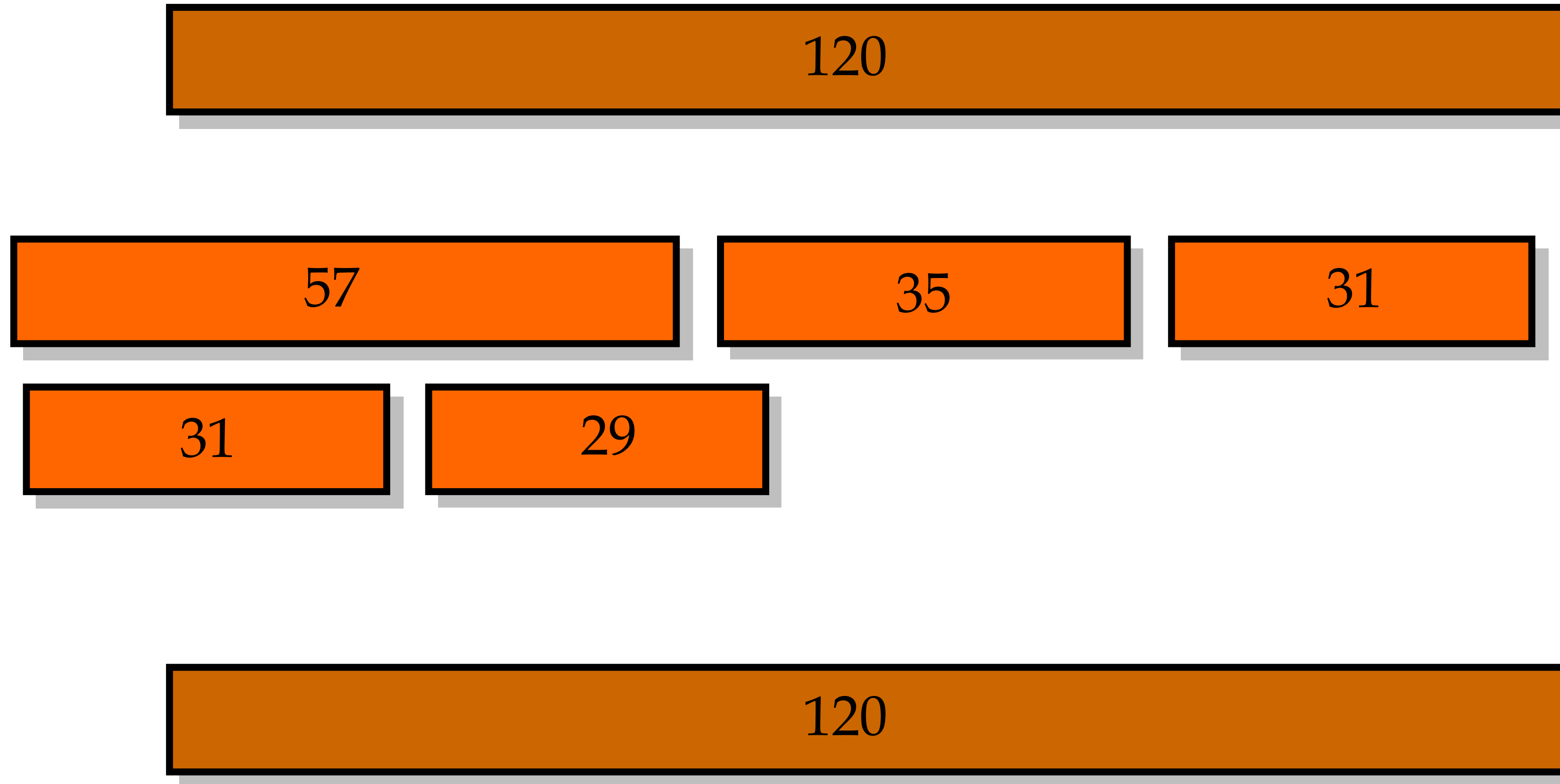
31

120

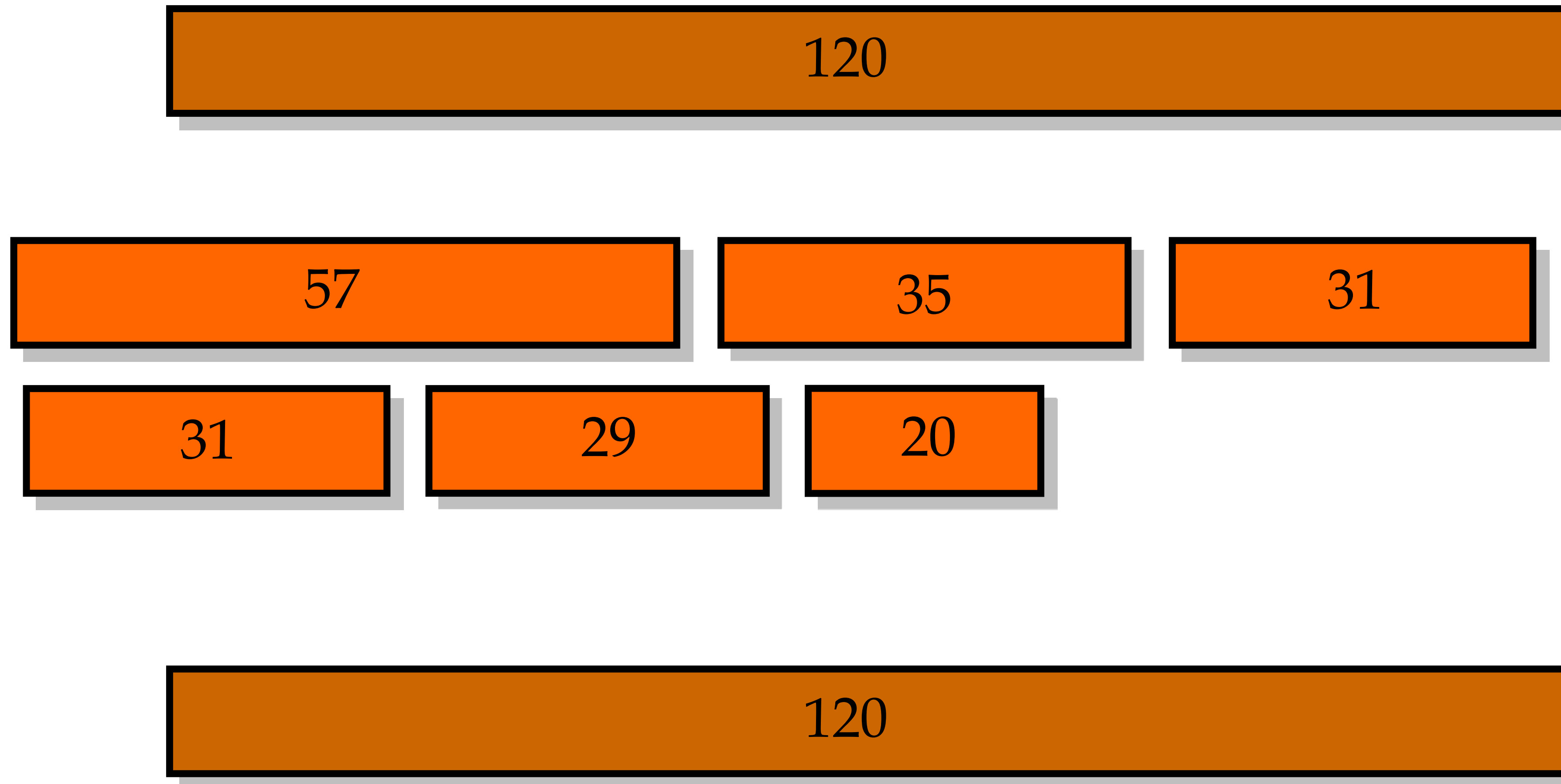
Keine Lösung?!



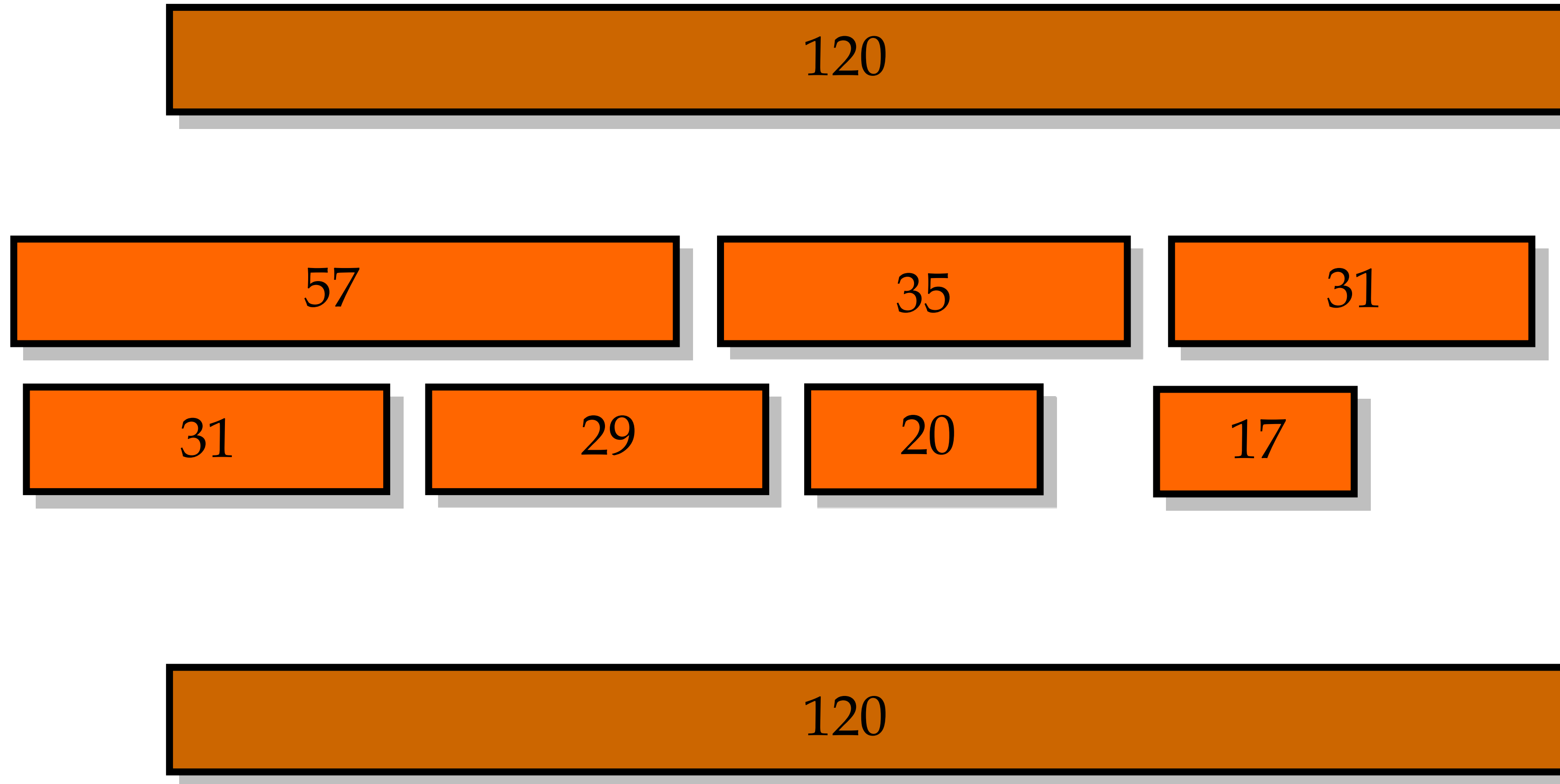
Keine Lösung?!



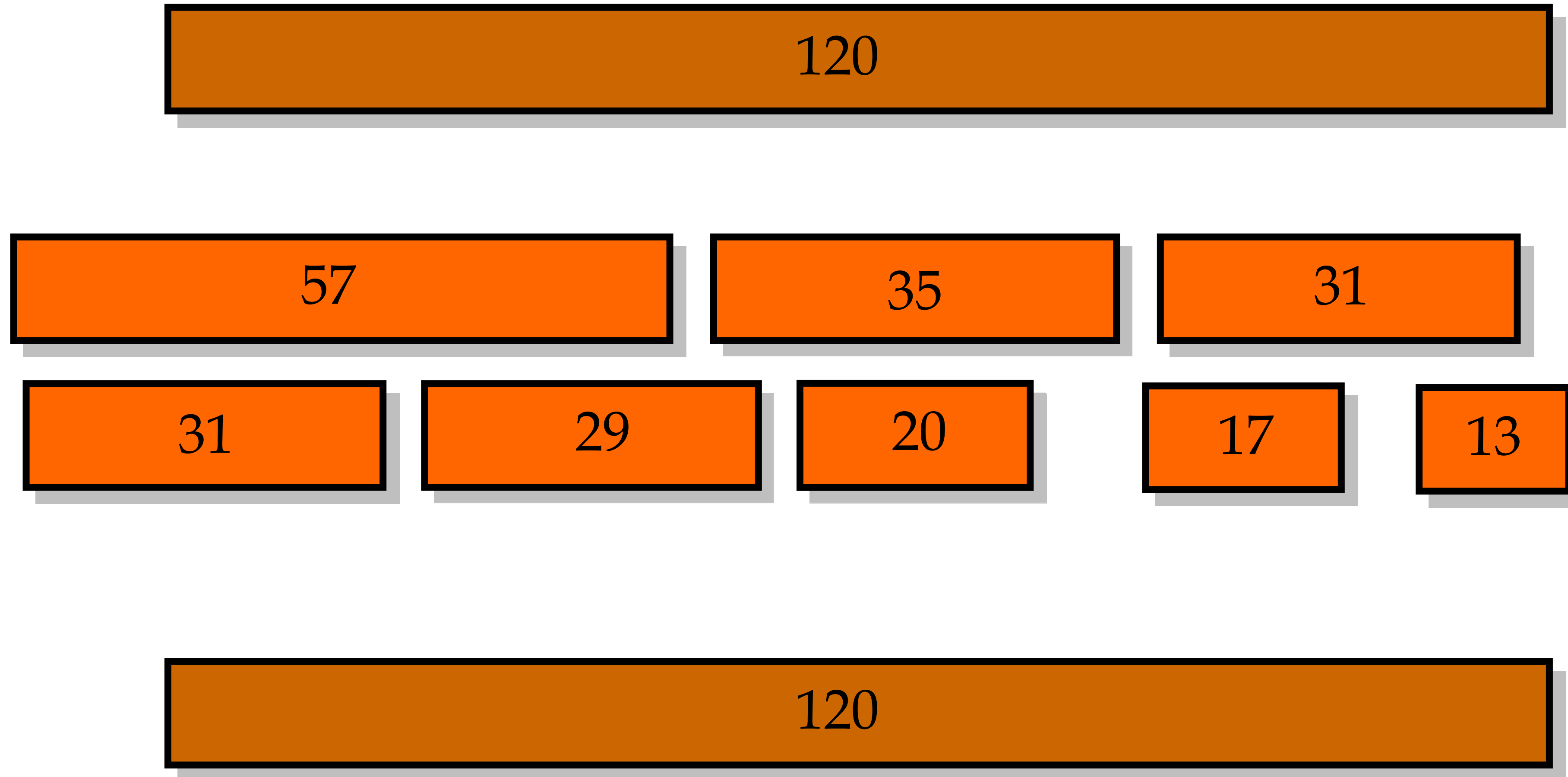
Keine Lösung?!



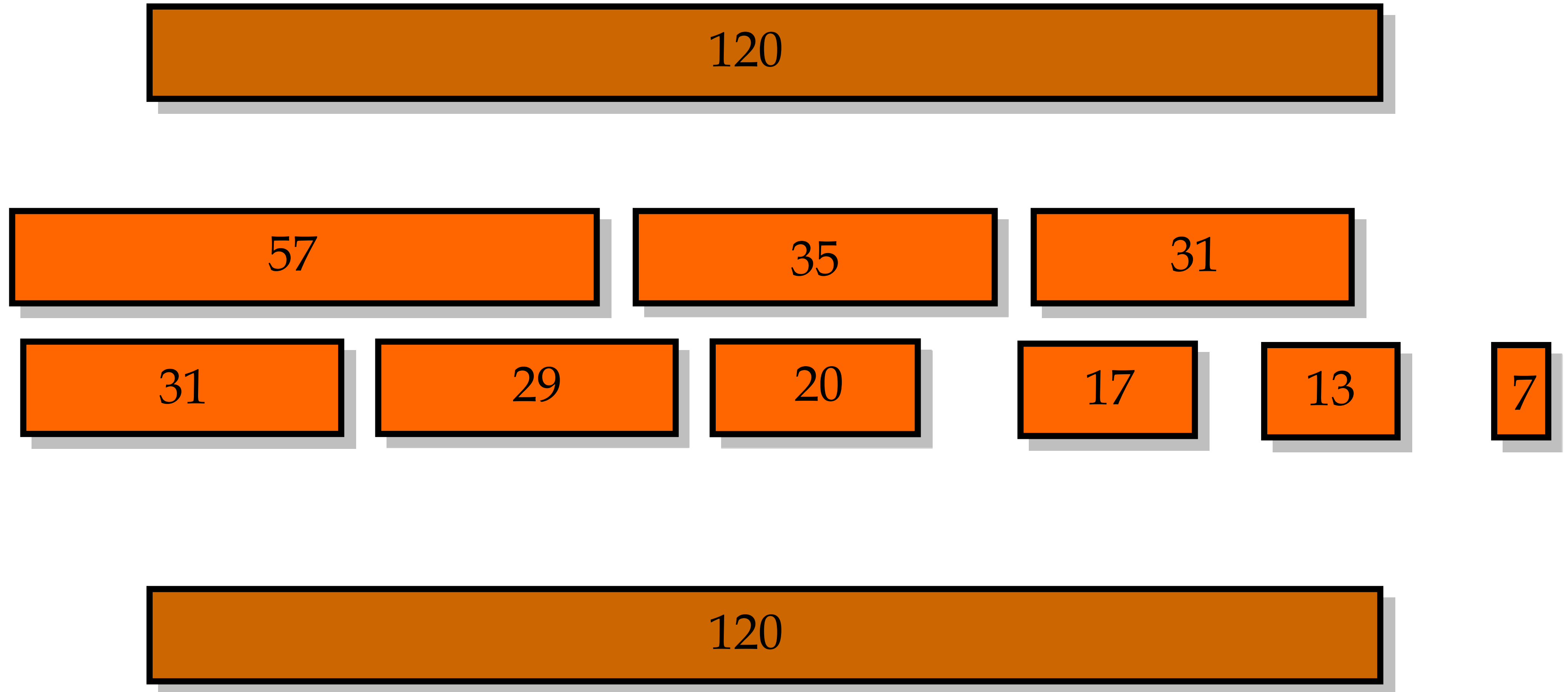
Keine Lösung?!



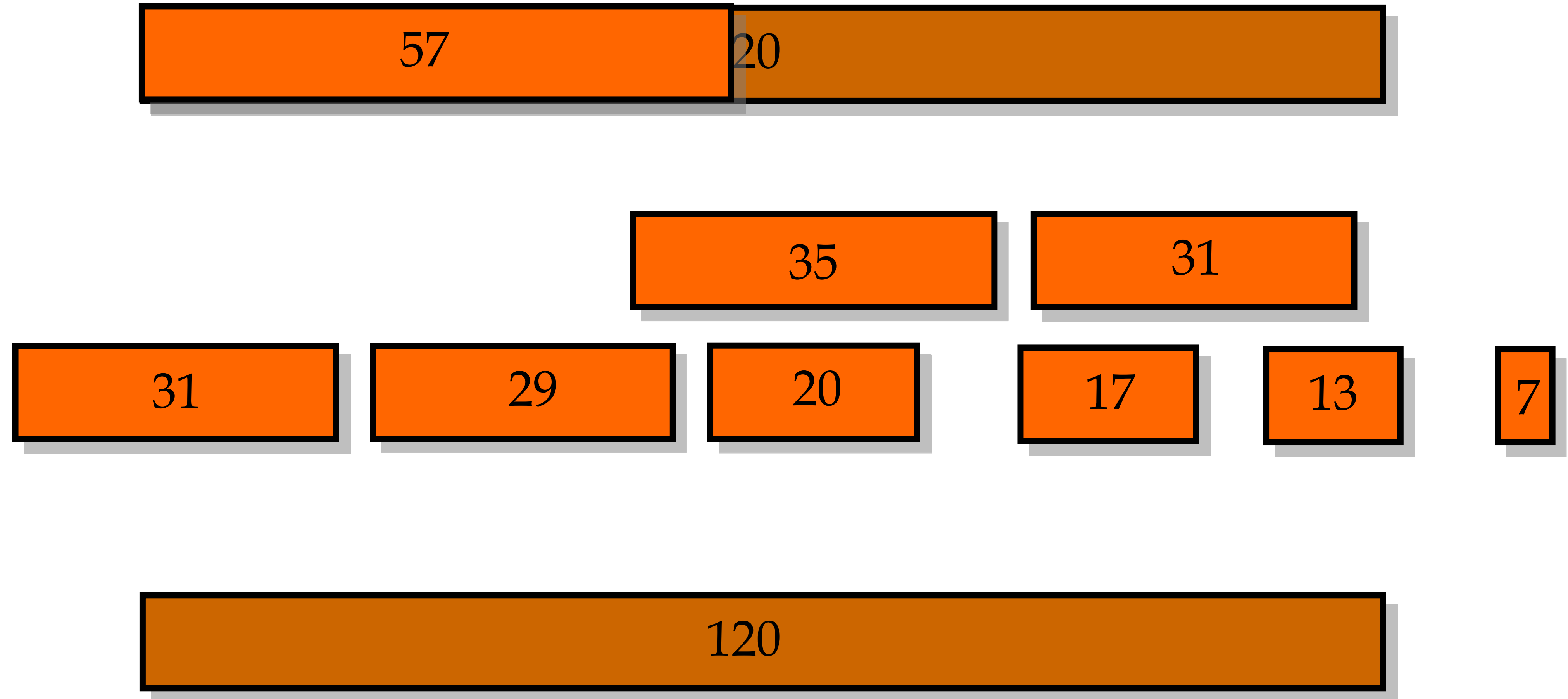
Keine Lösung?!



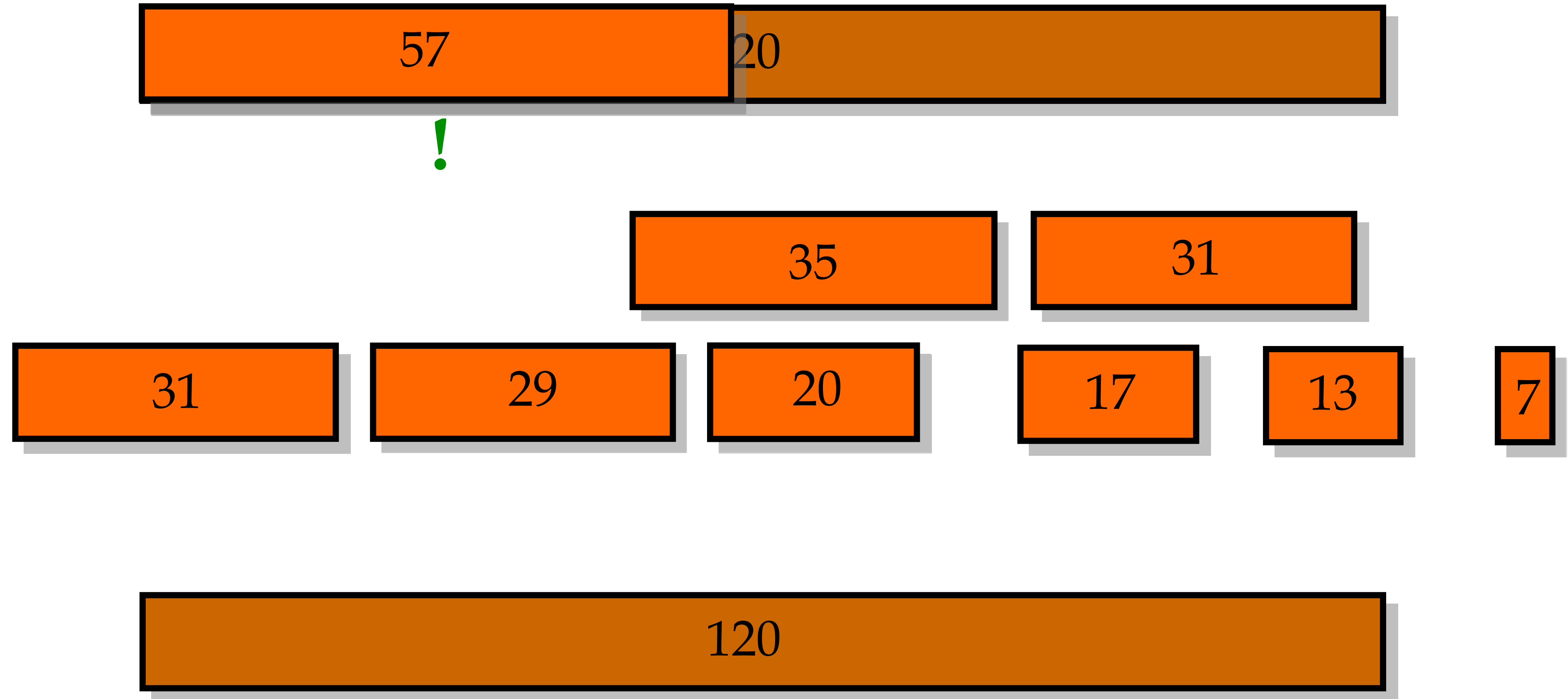
Keine Lösung?!



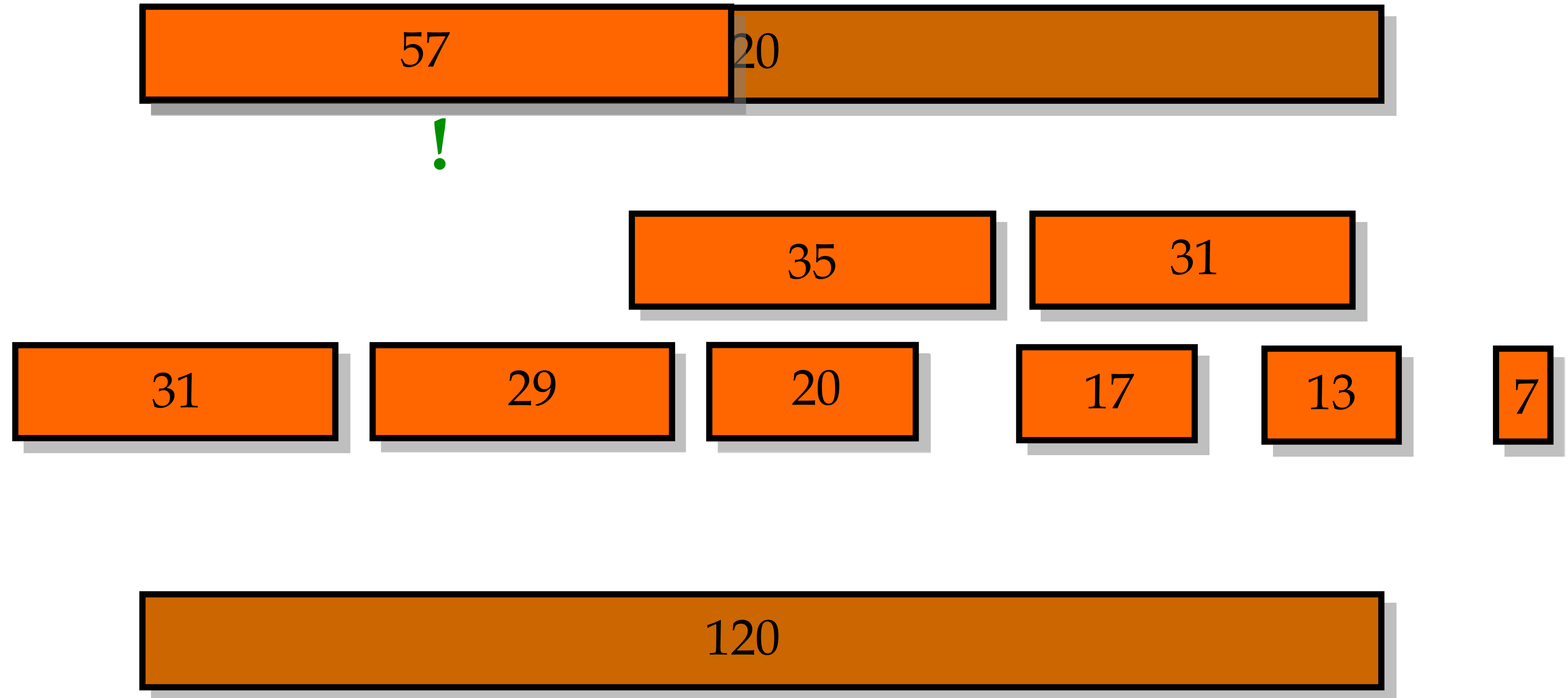
Keine Lösung?!



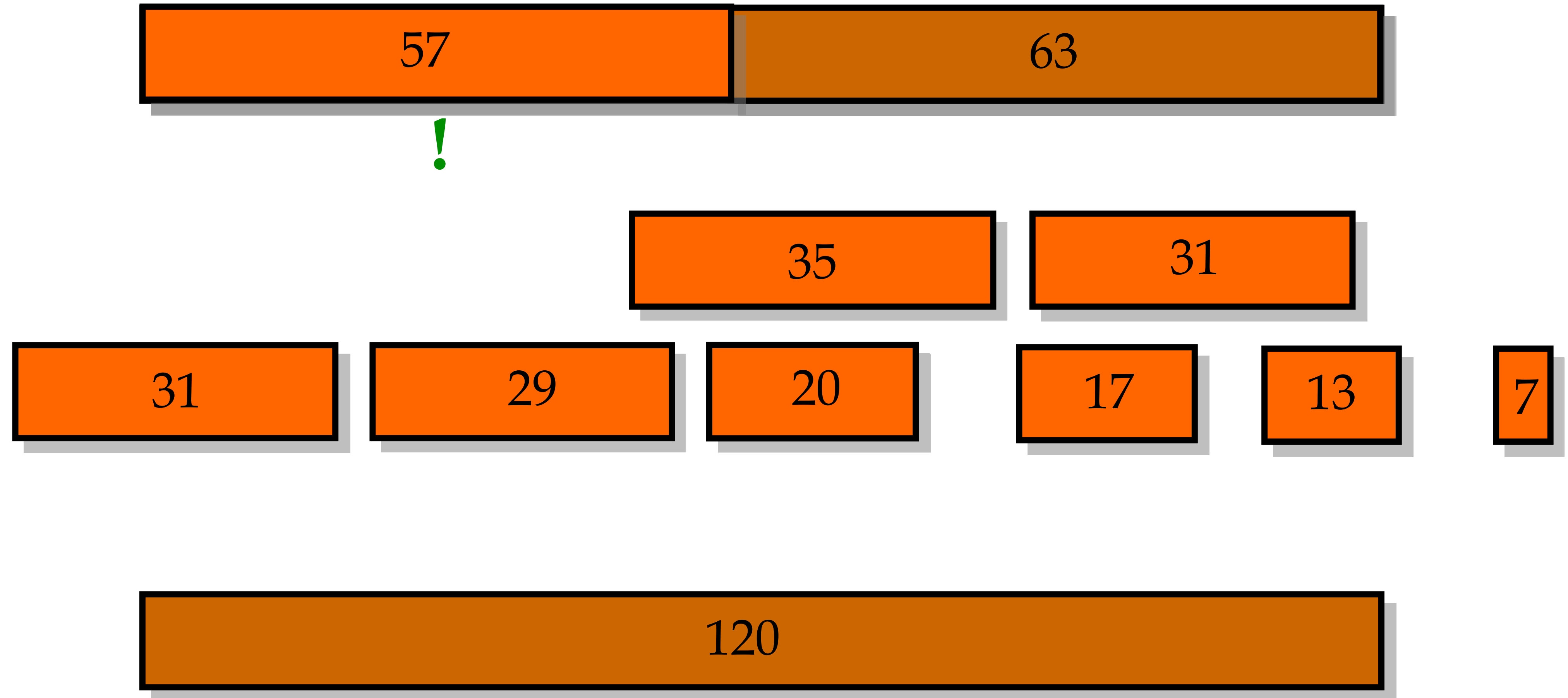
Keine Lösung?!



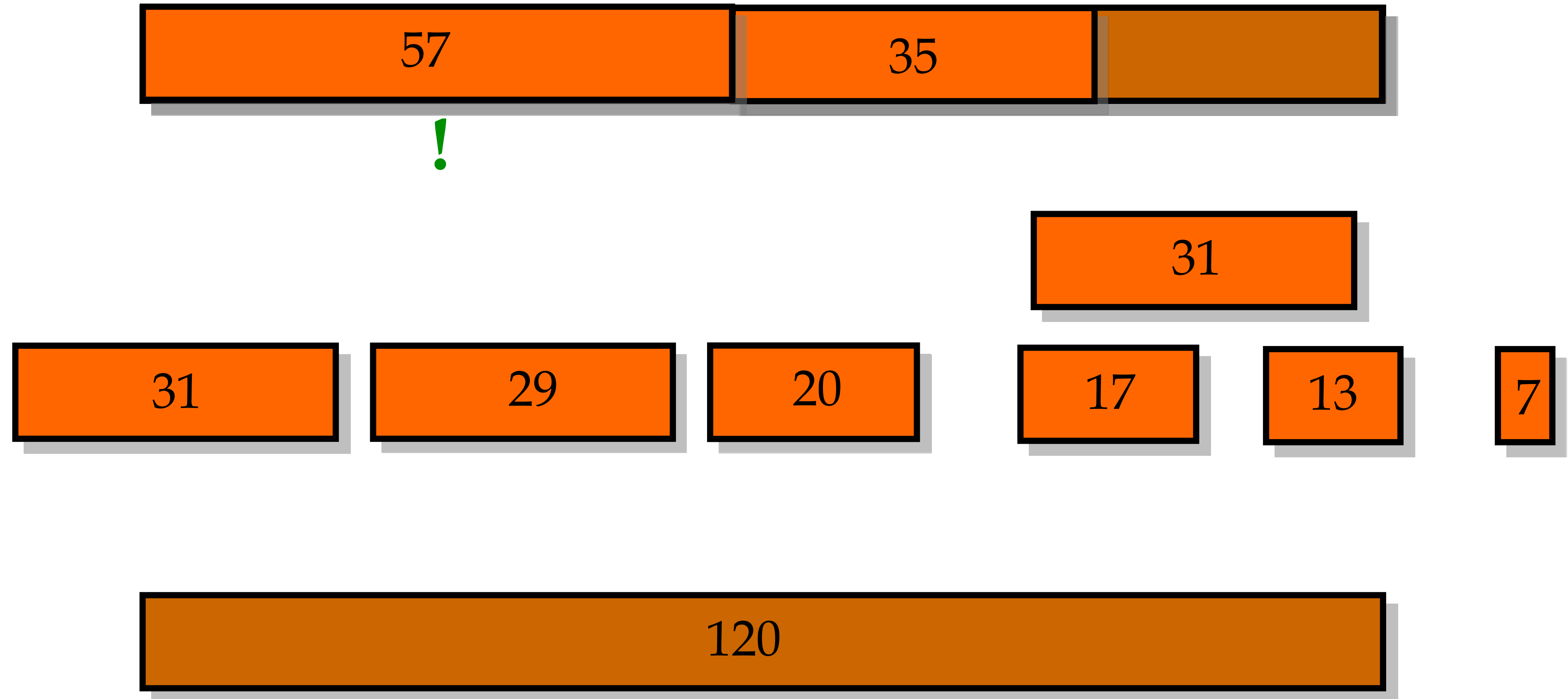
Keine Lösung?!



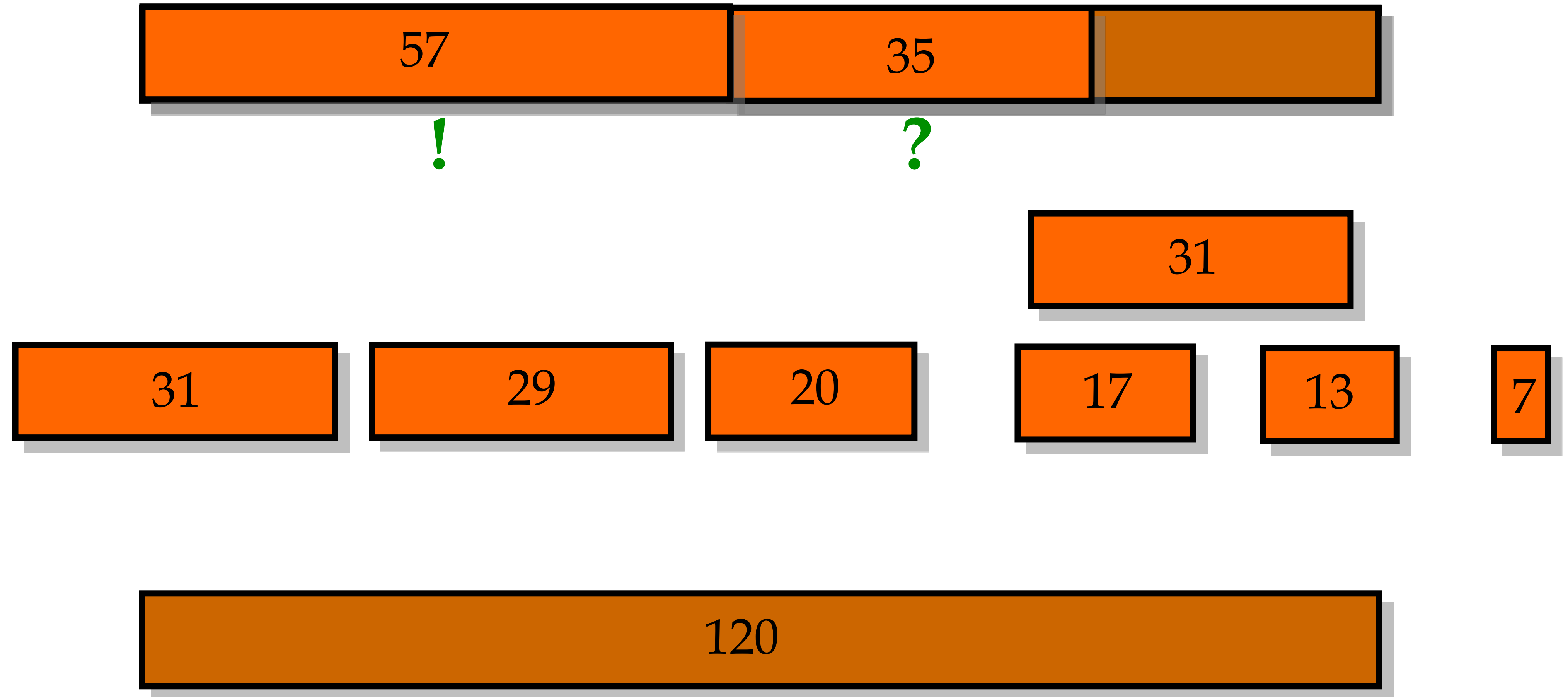
Keine Lösung?!



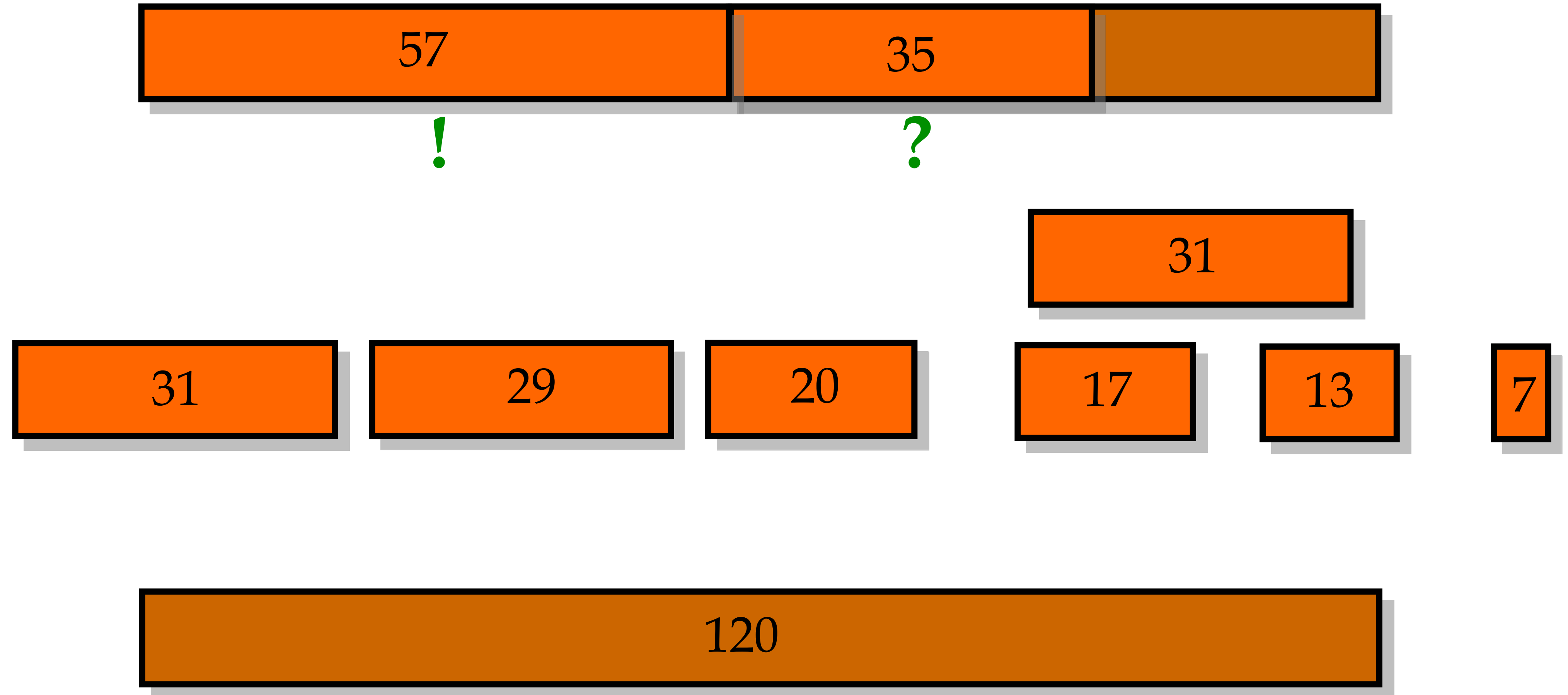
Keine Lösung?!



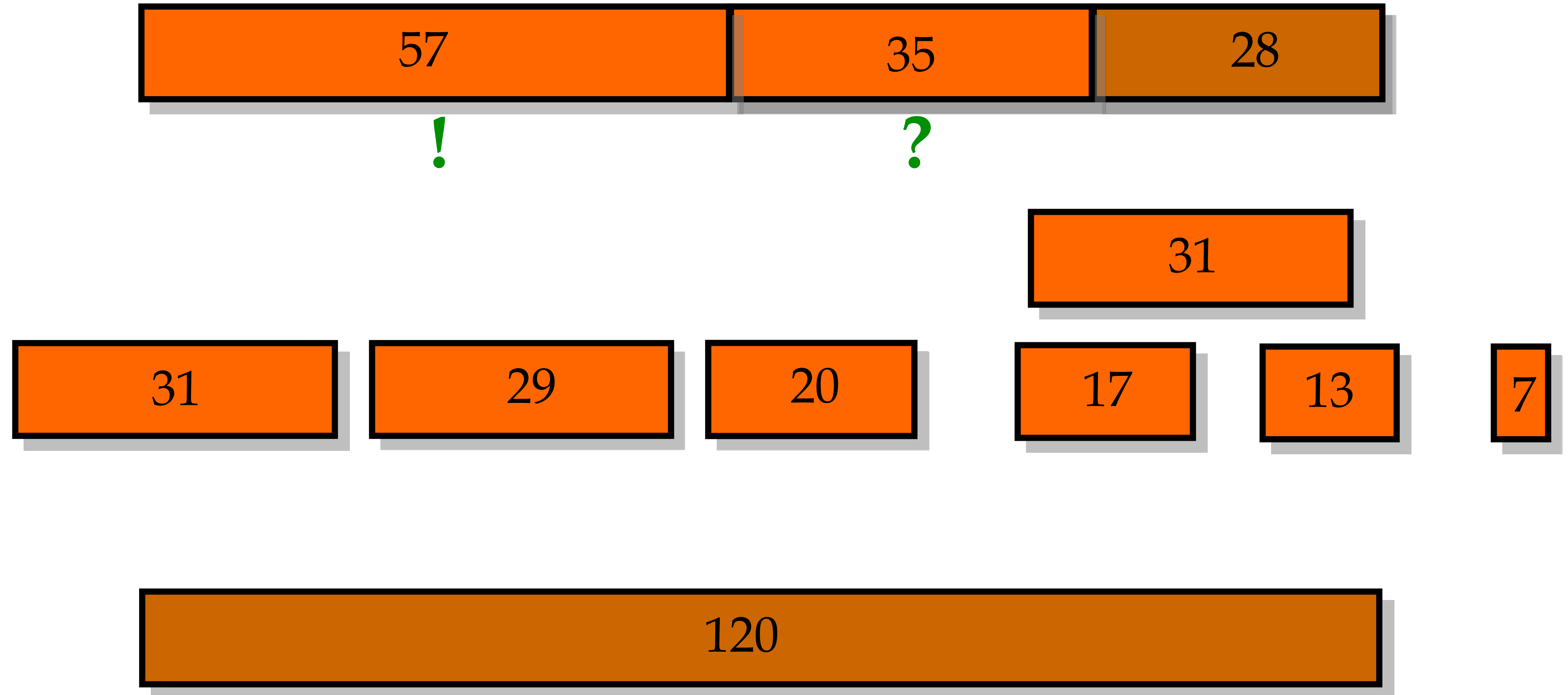
Keine Lösung?!



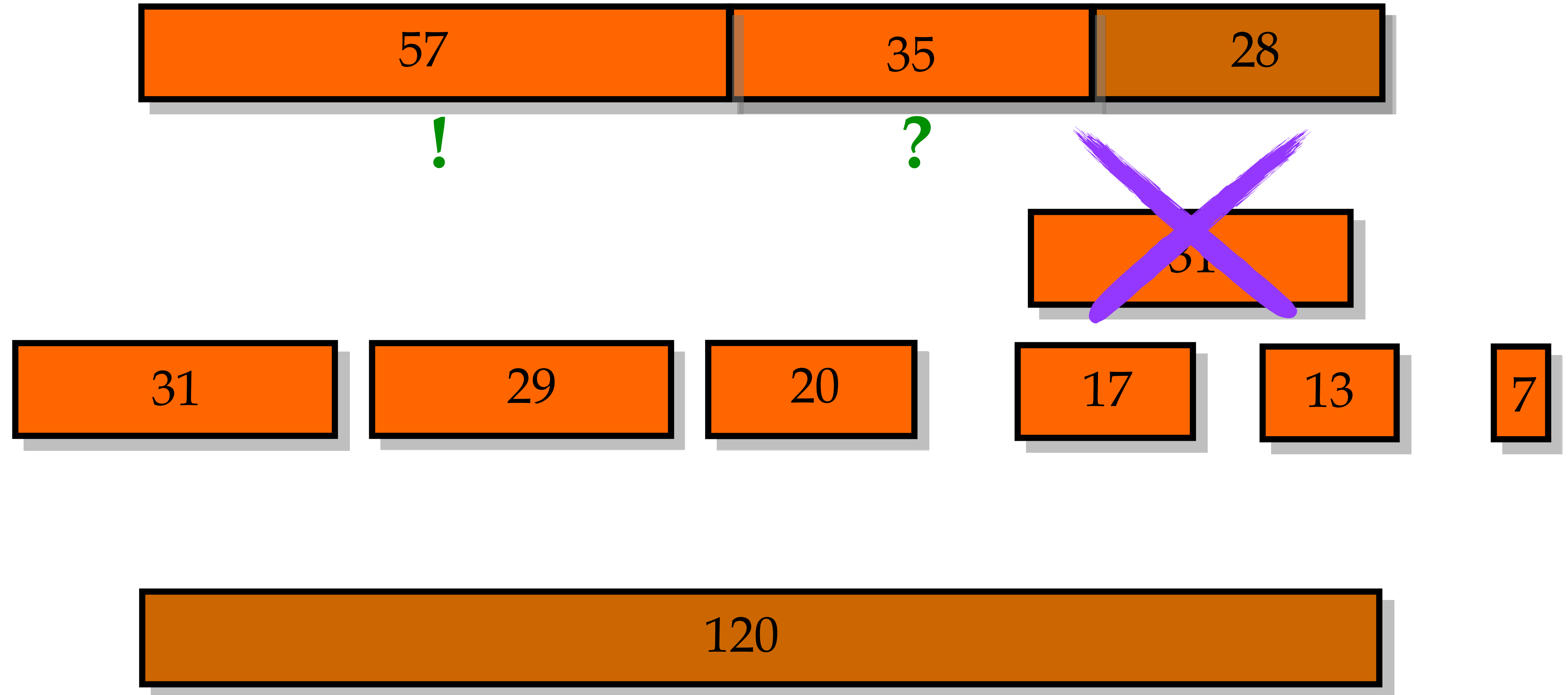
Keine Lösung?!



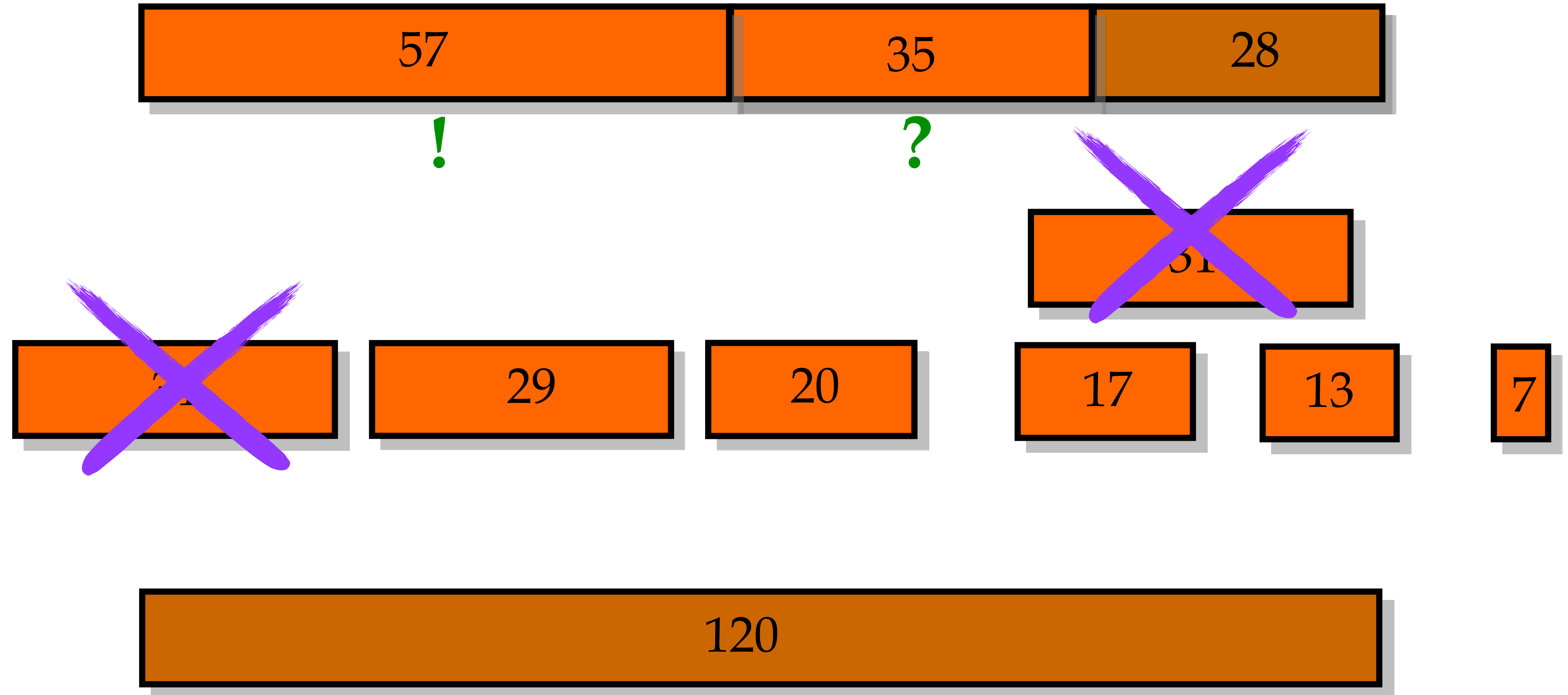
Keine Lösung?!



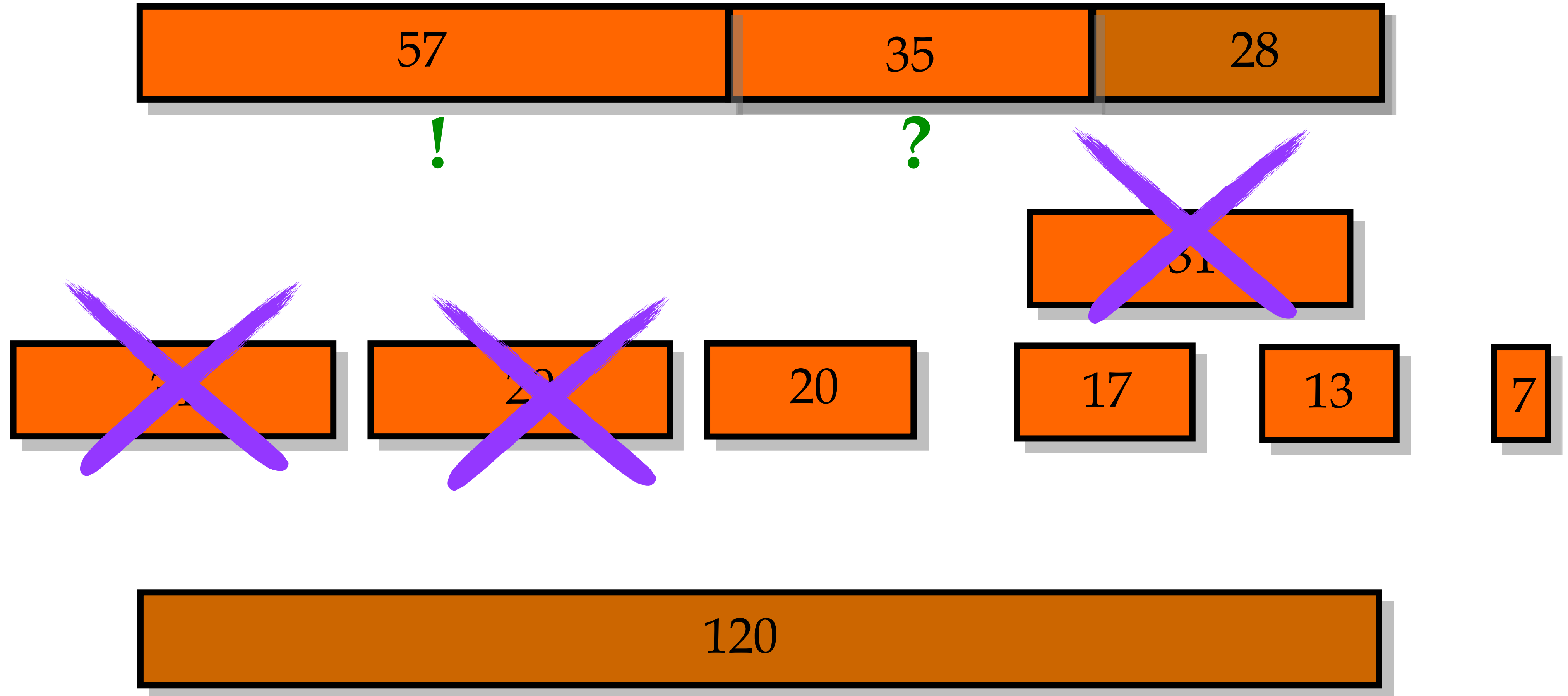
Keine Lösung?!



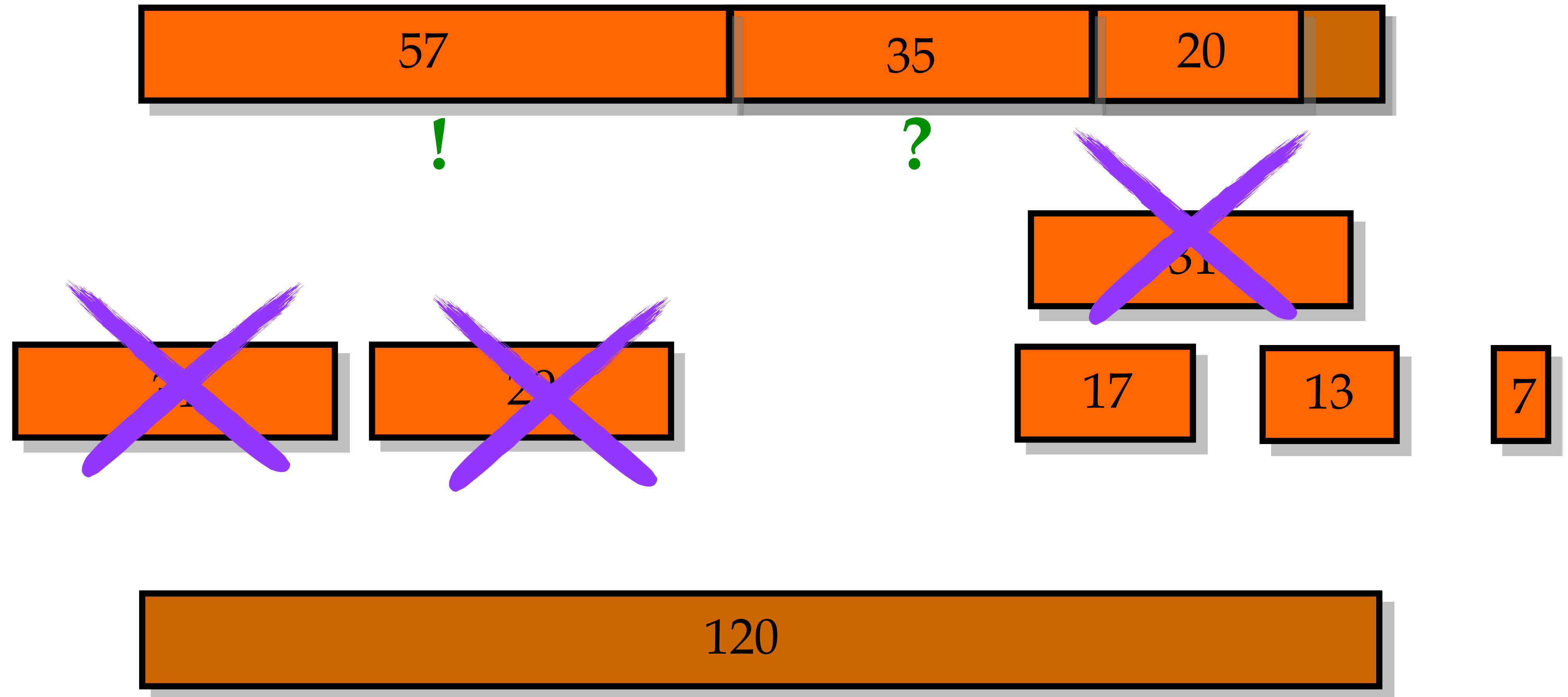
Keine Lösung?!



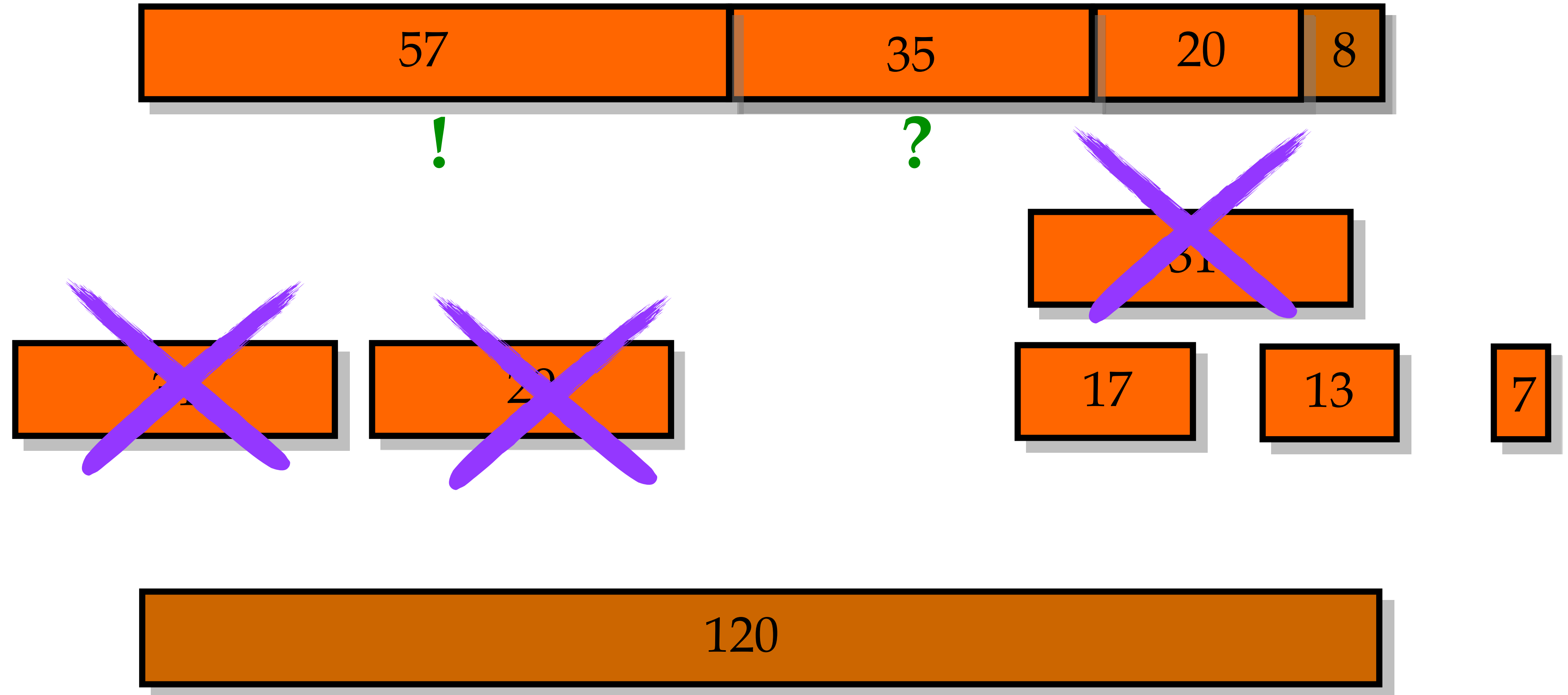
Keine Lösung?!



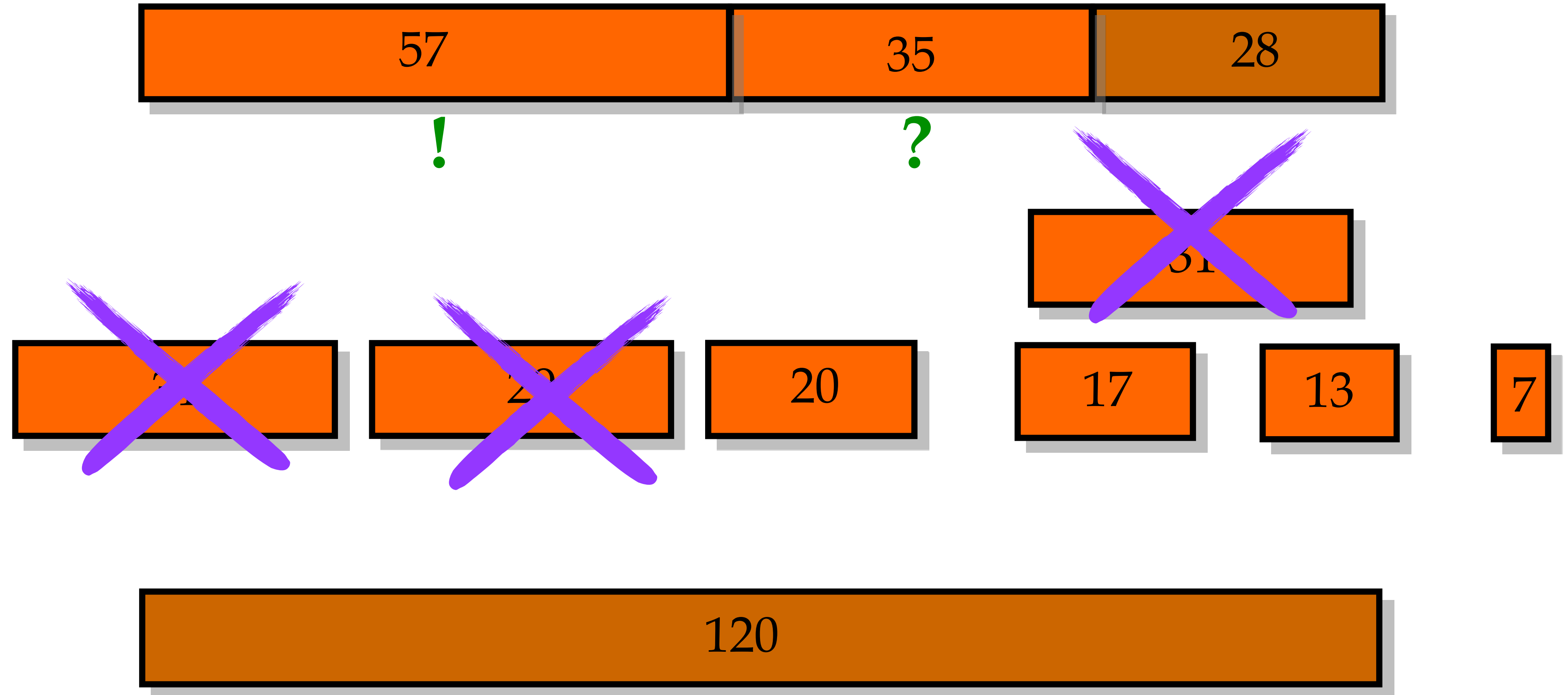
Keine Lösung?!



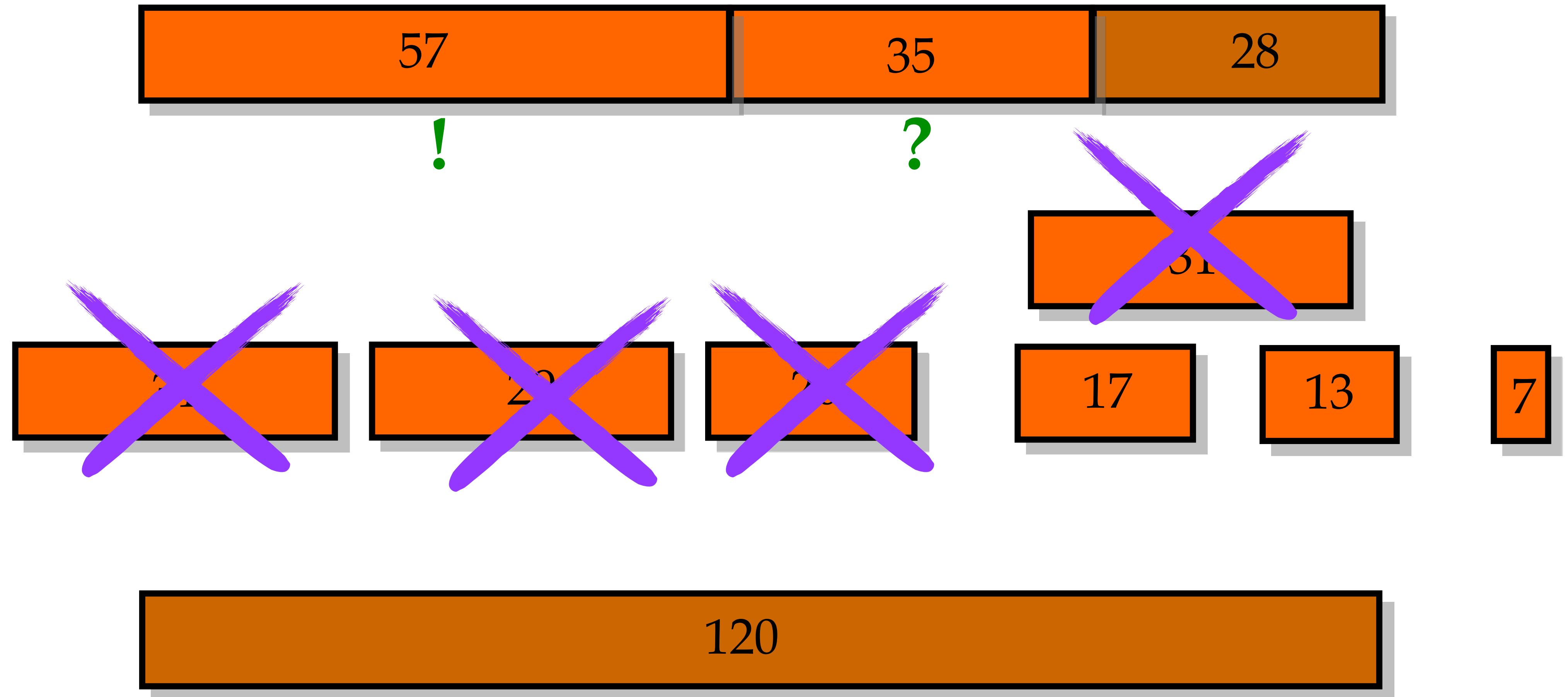
Keine Lösung?!



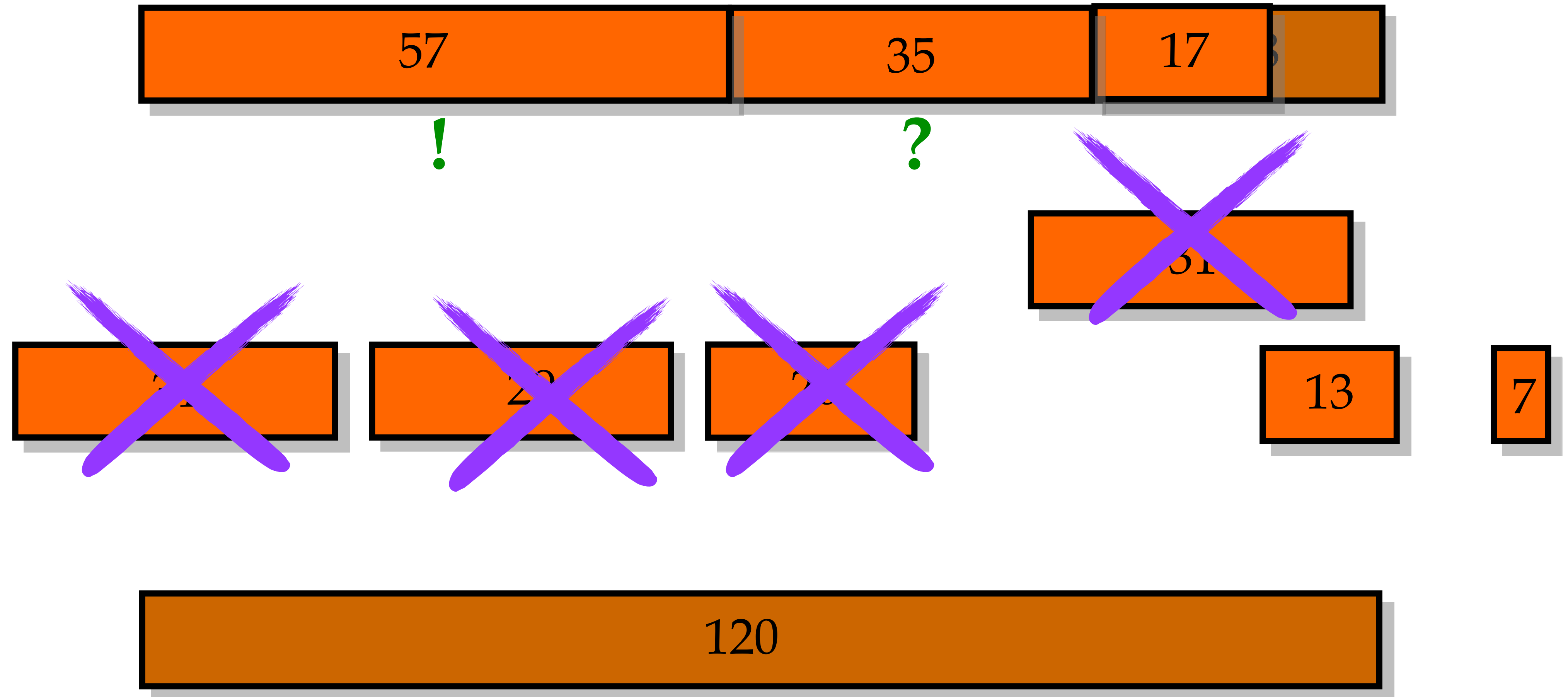
Keine Lösung?!



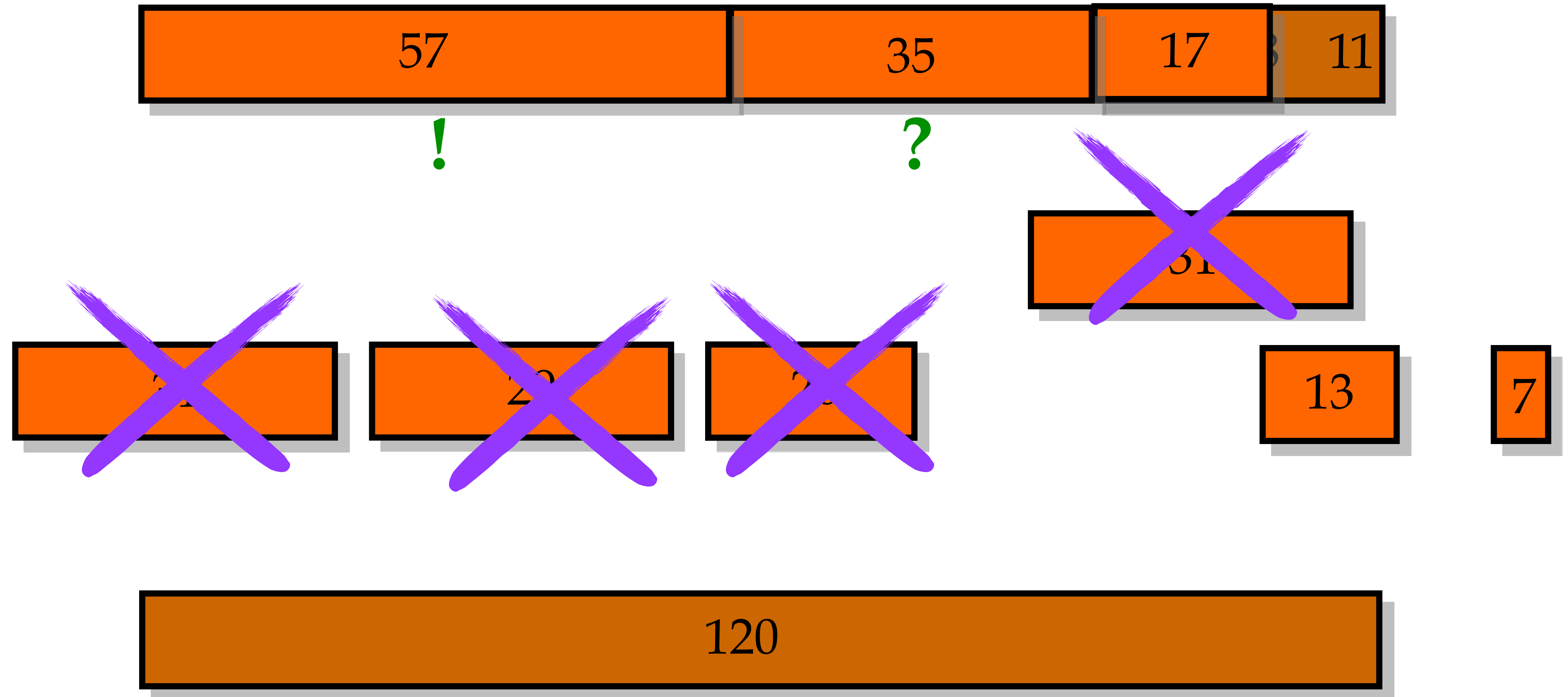
Keine Lösung?!



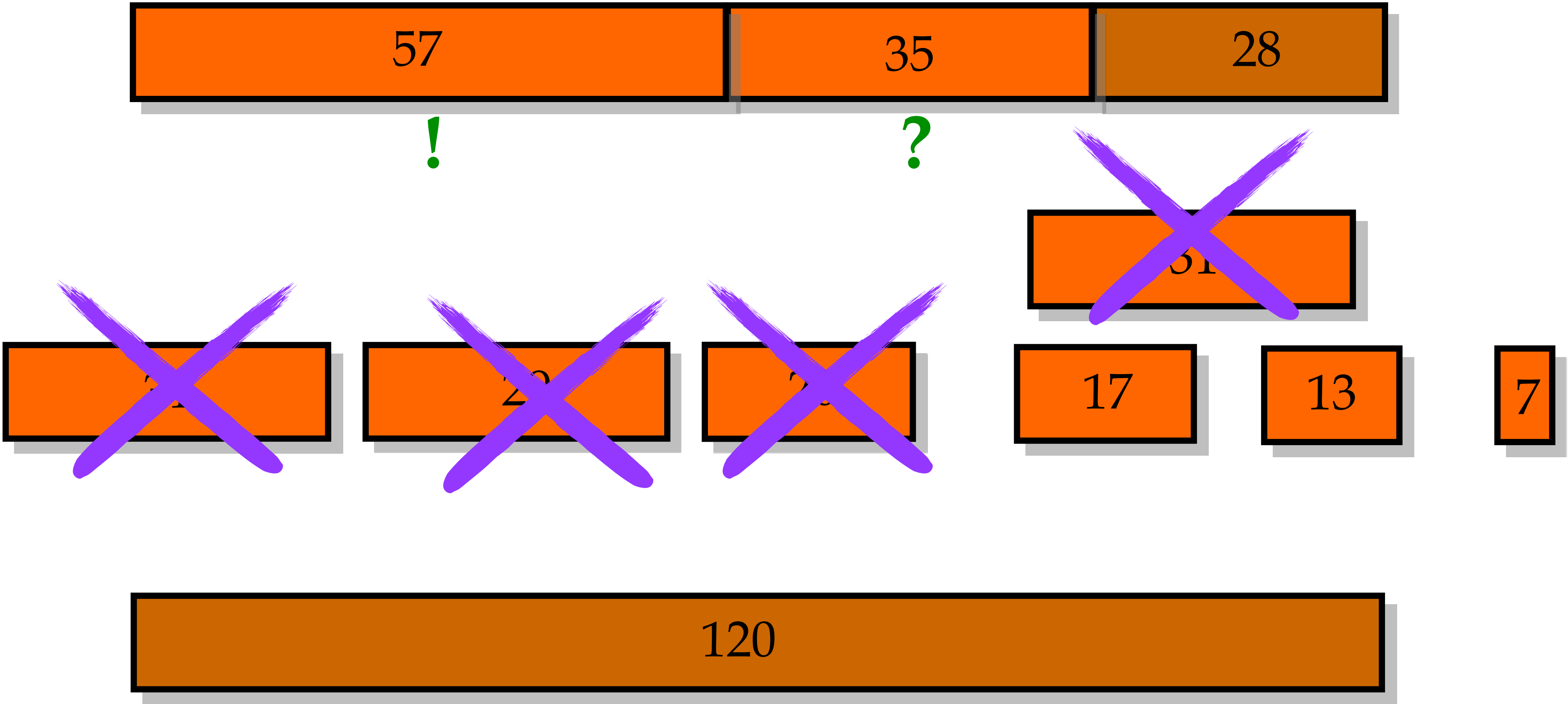
Keine Lösung?!



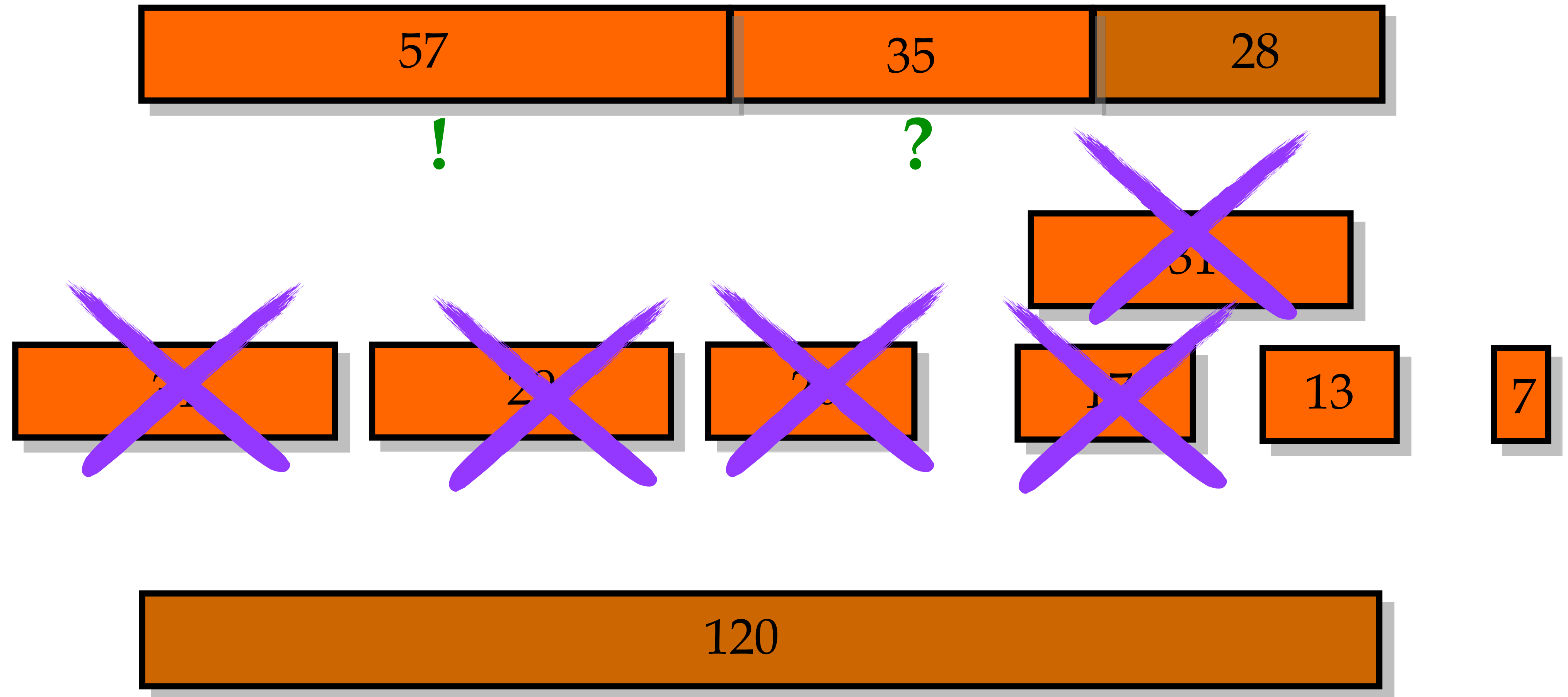
Keine Lösung?!



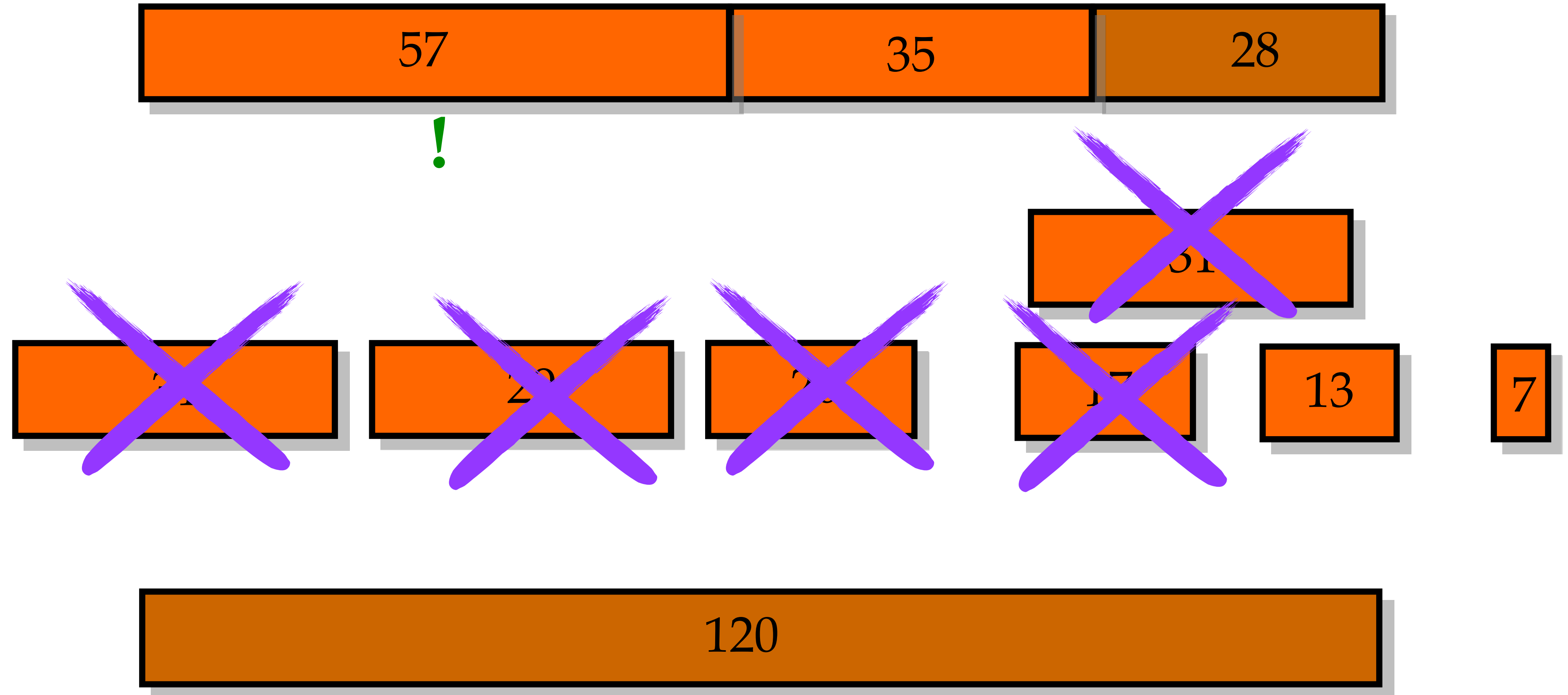
Keine Lösung?!



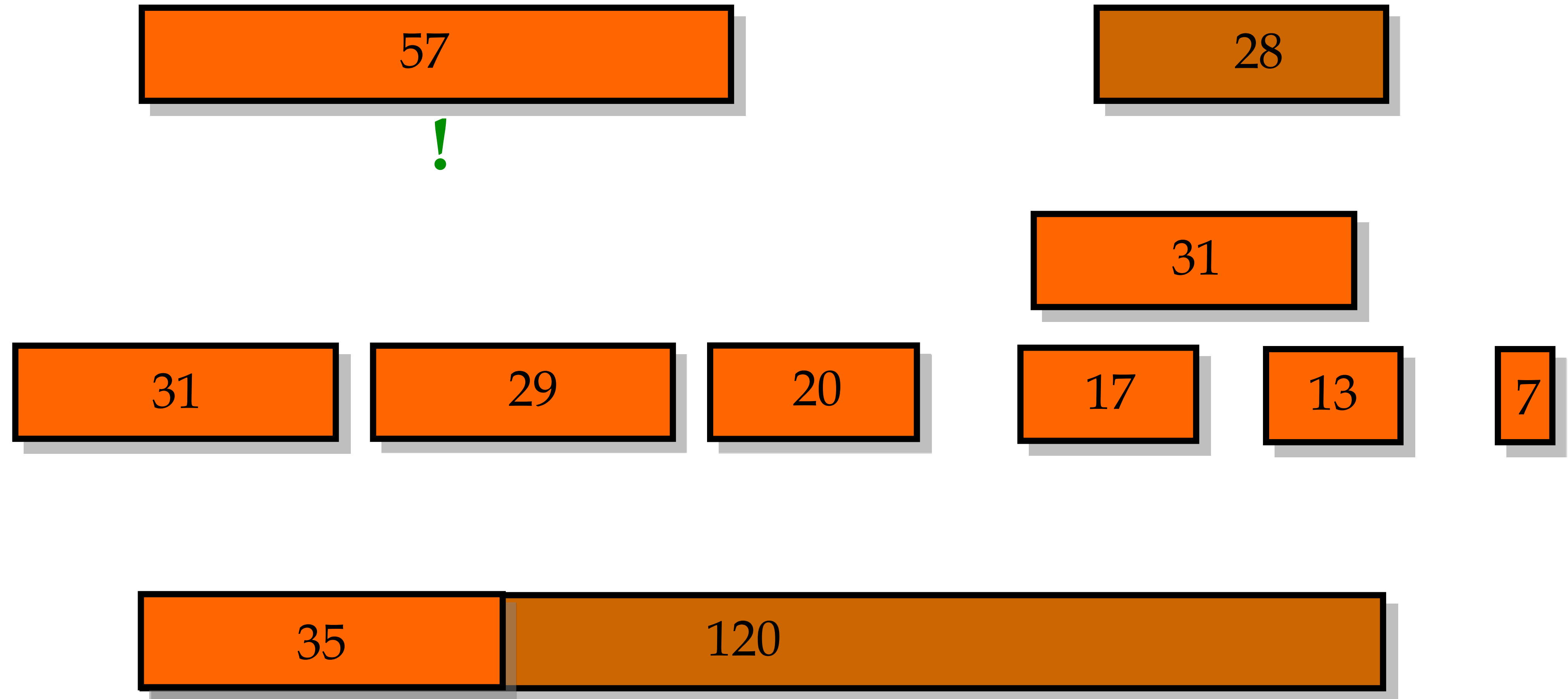
Keine Lösung?!



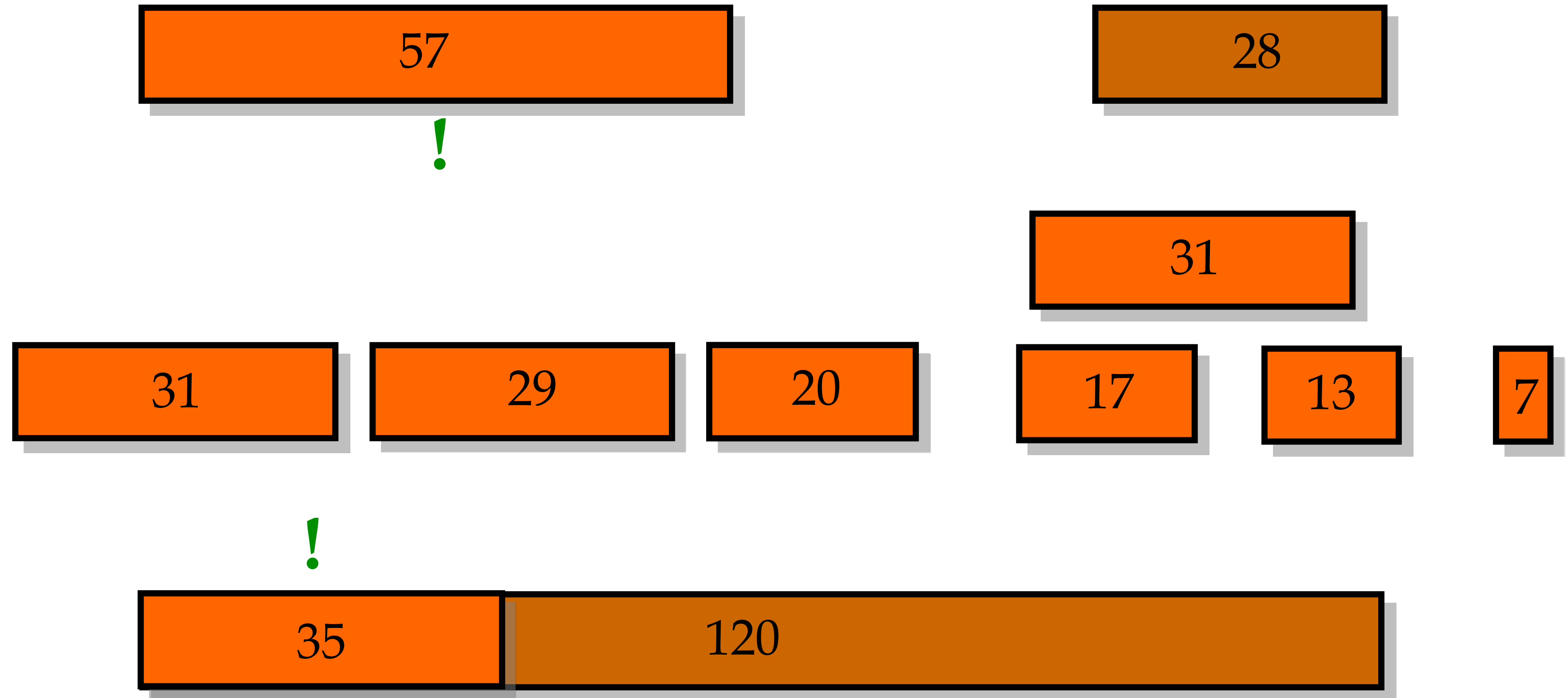
Keine Lösung?!



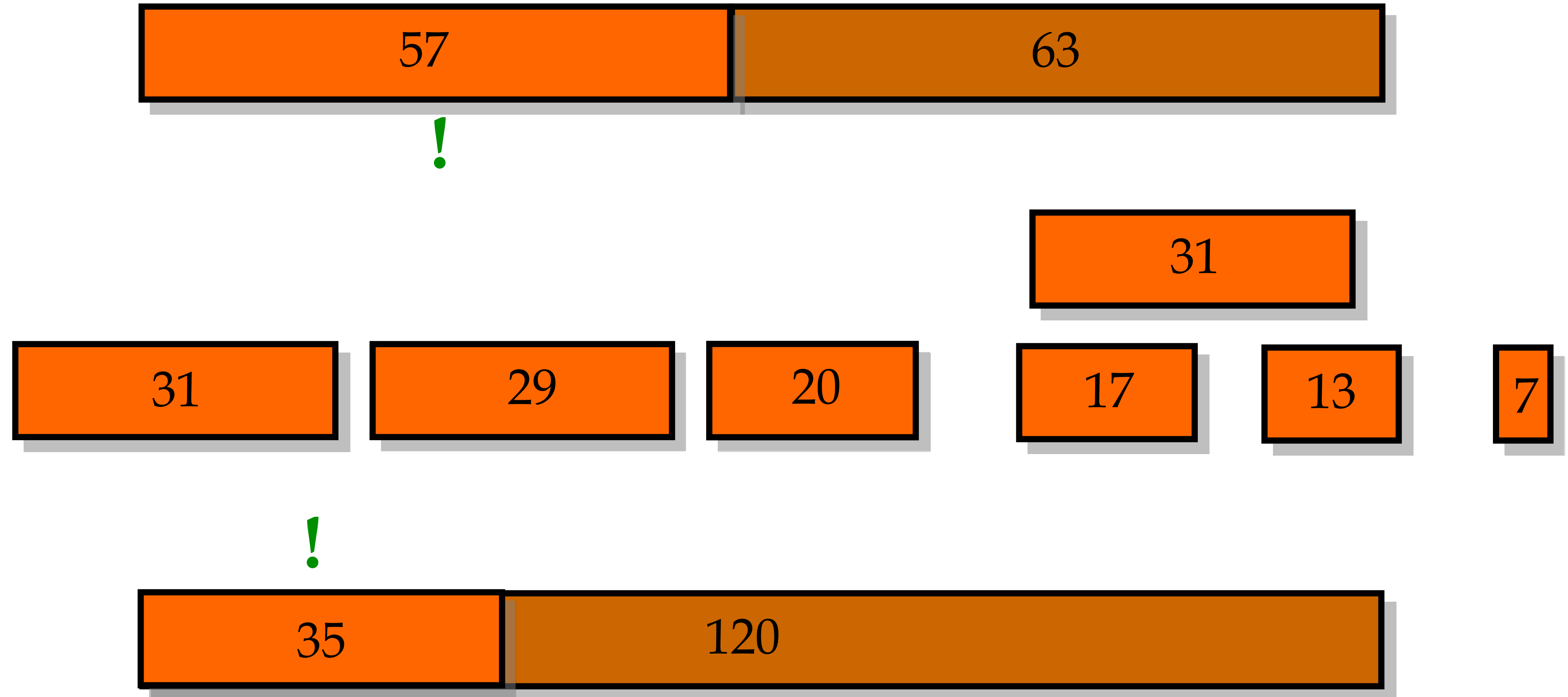
Keine Lösung?!



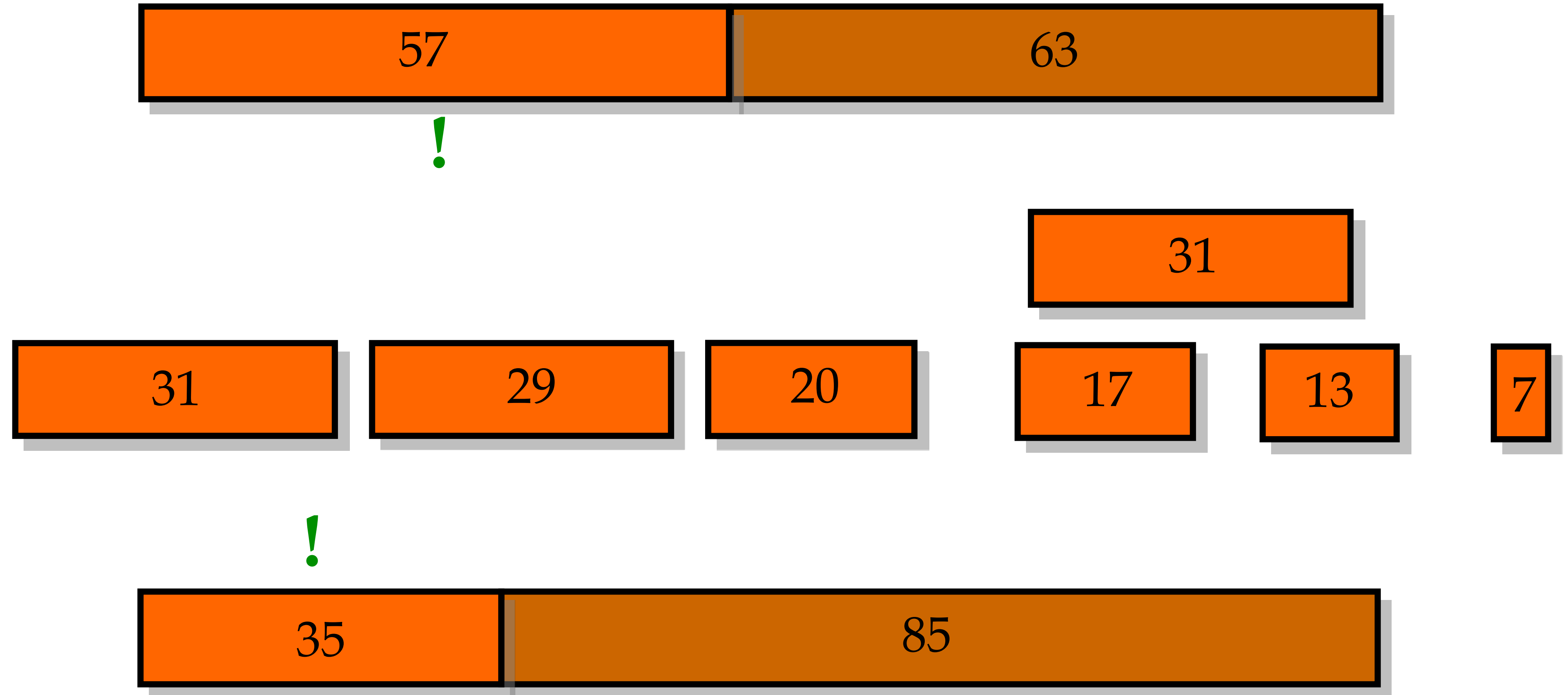
Keine Lösung?!



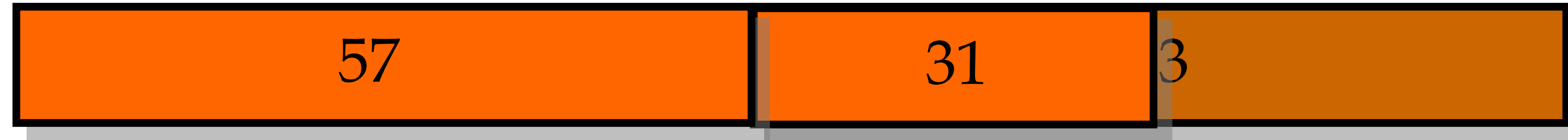
Keine Lösung?!



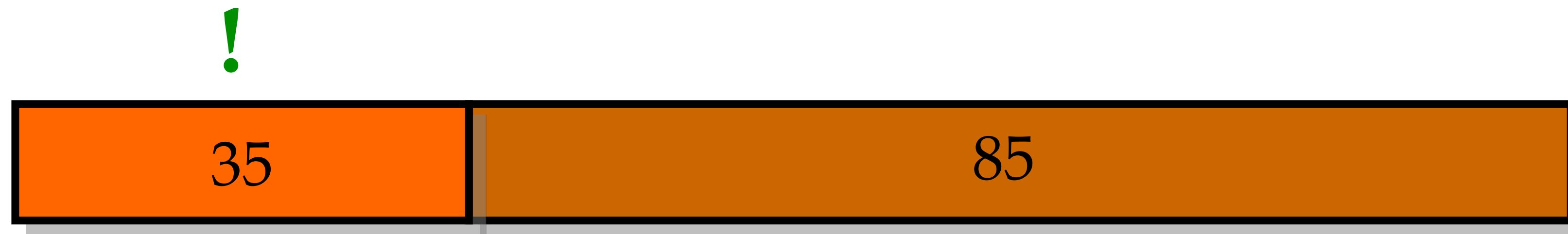
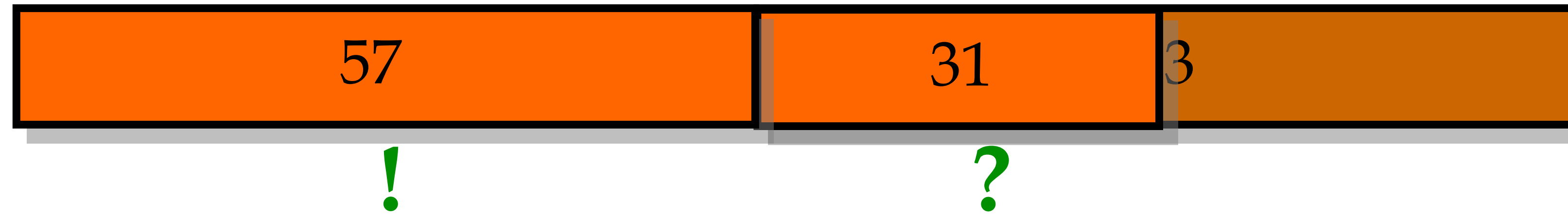
Keine Lösung?!



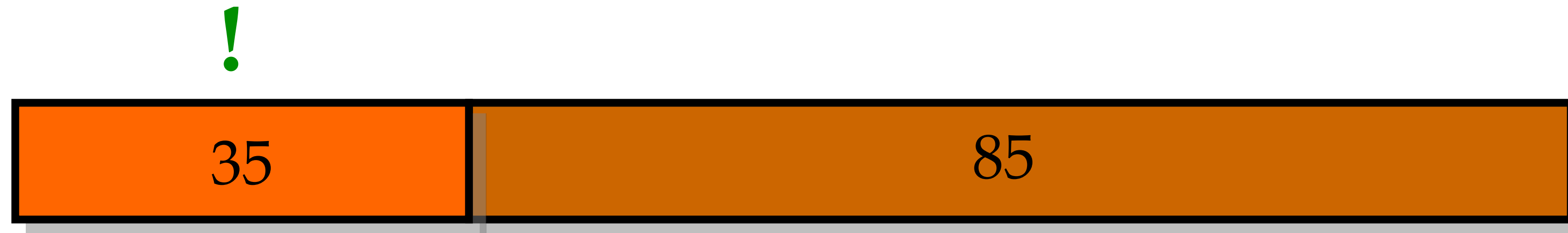
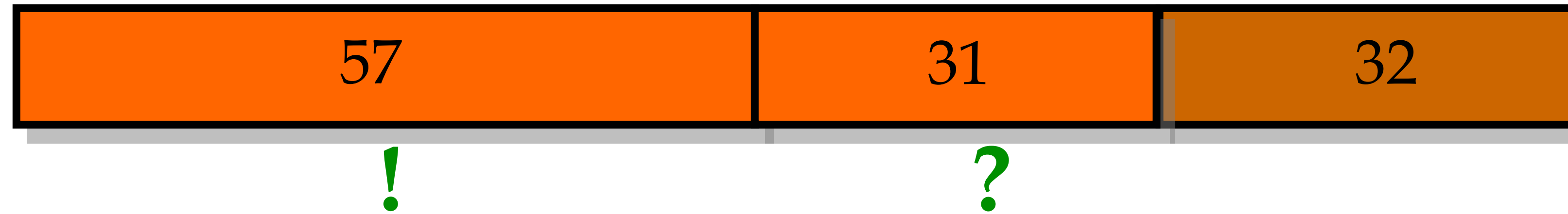
Keine Lösung?!



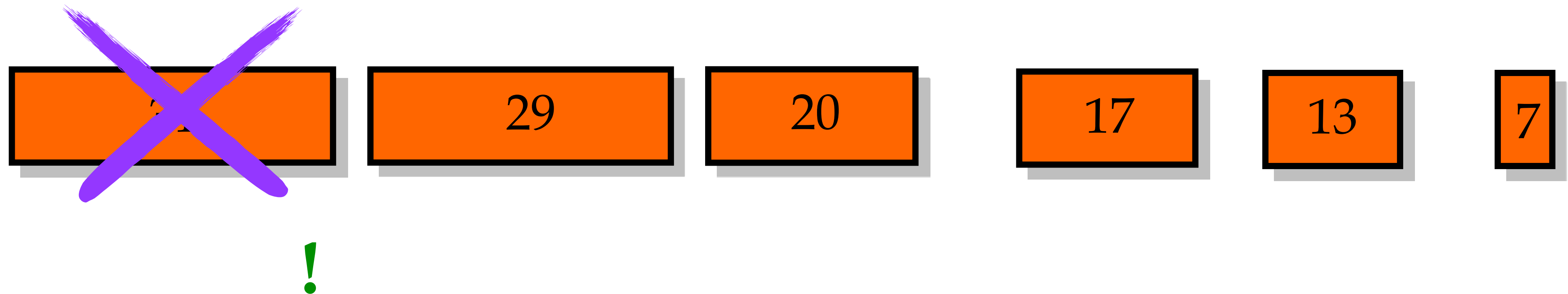
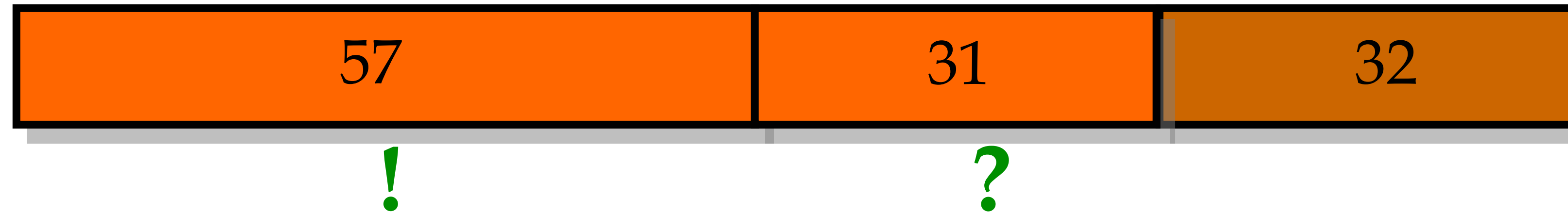
Keine Lösung?!



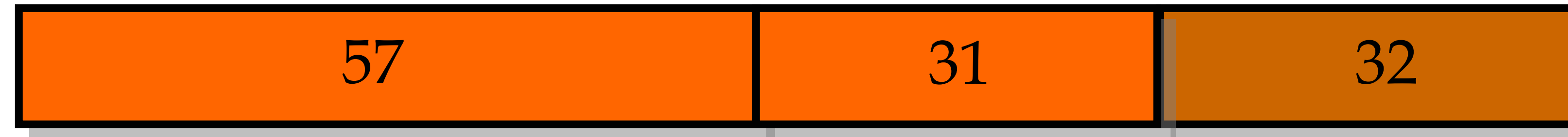
Keine Lösung?!



Keine Lösung?!

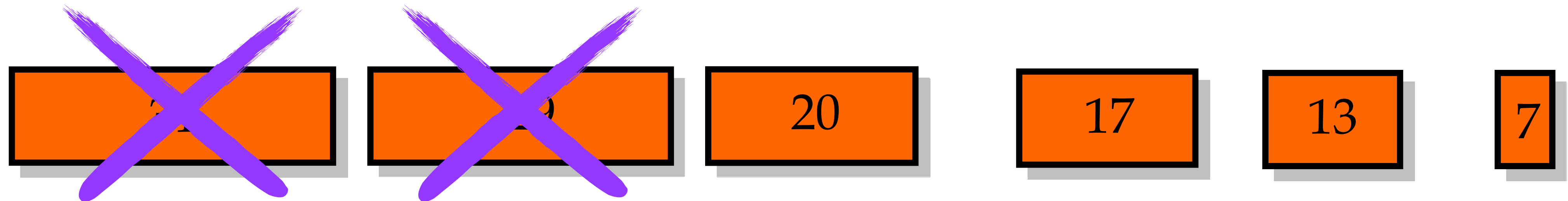


Keine Lösung?!



!

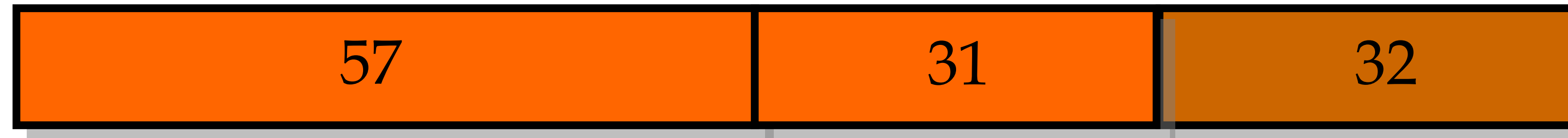
?



!

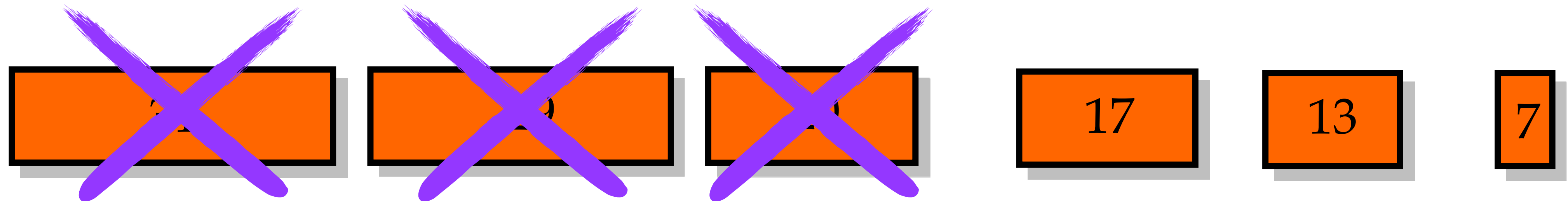


Keine Lösung?!



!

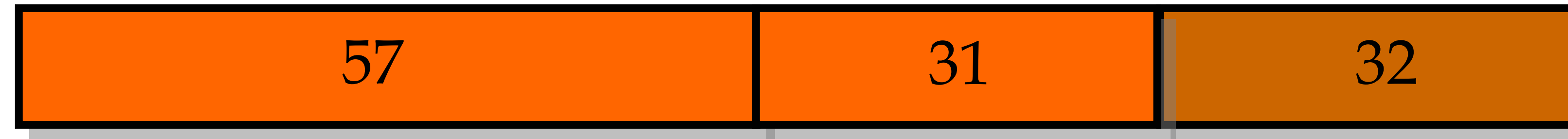
?



!

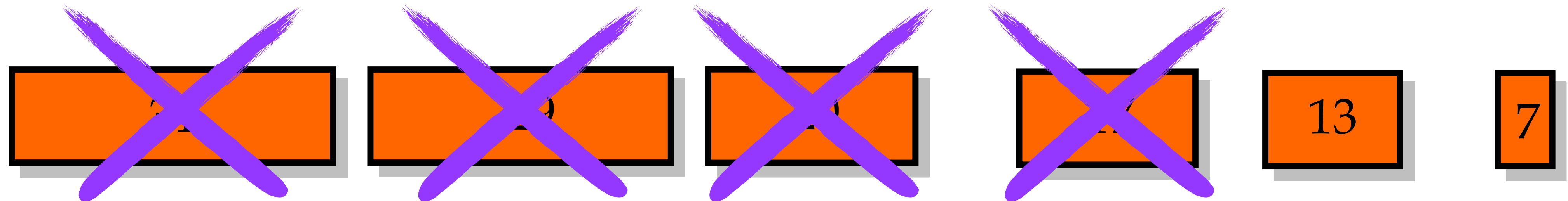


Keine Lösung?!



!

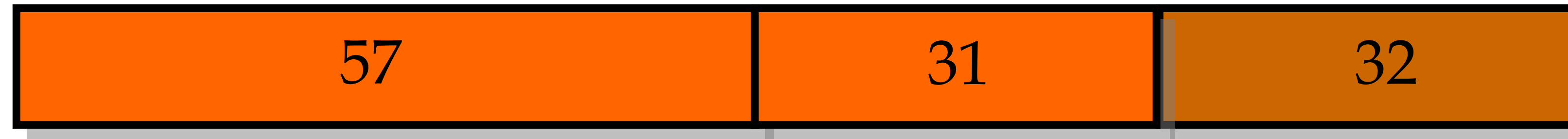
?



!

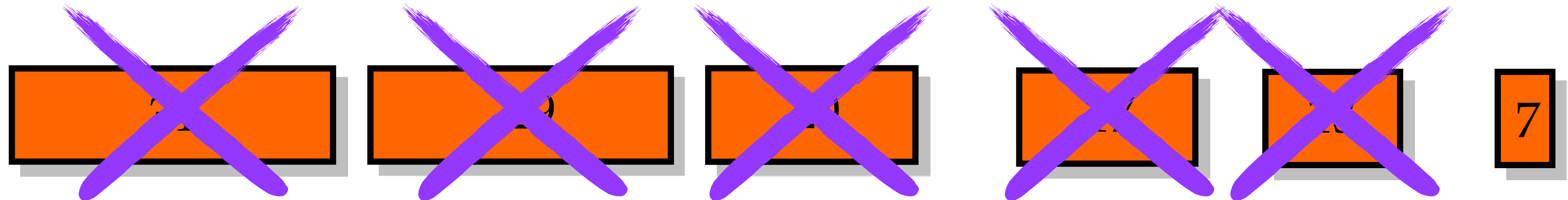


Keine Lösung?!



!

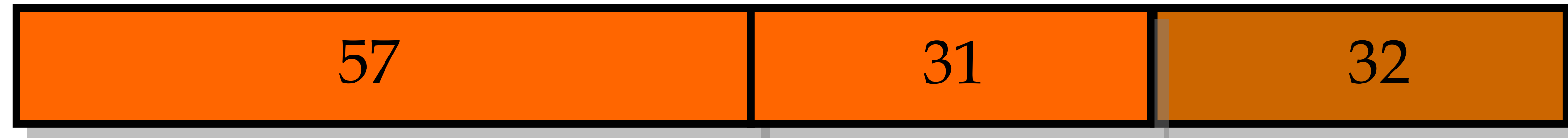
?



!

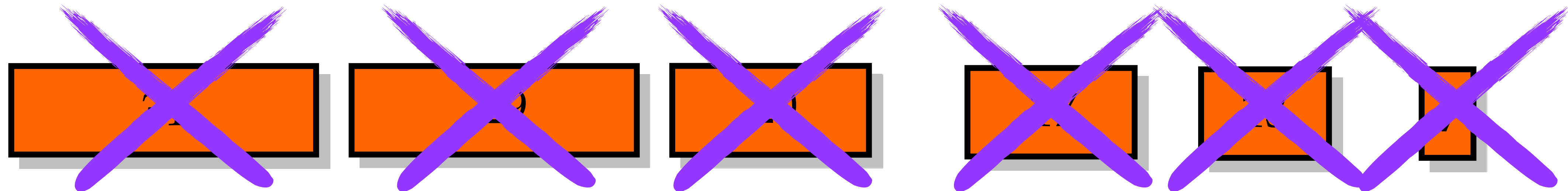


Keine Lösung?!



!

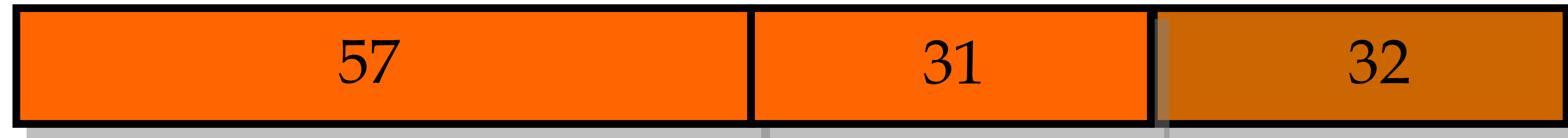
?



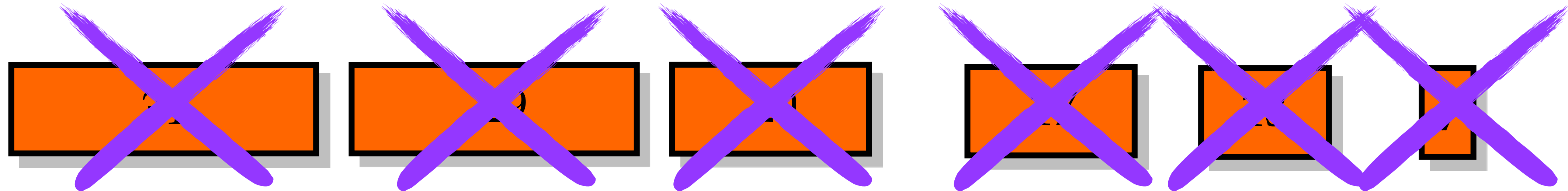
!



Keine Lösung?!



!



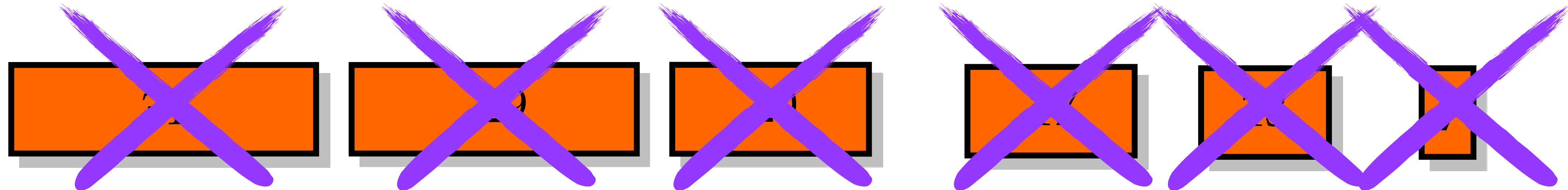
!



Keine Lösung?!

57

32



35 31 85

Keine Lösung?!

57



32

31

29

20

17

13

7



35

31

85

Keine Lösung?!

57



32

31

29

20

17

13

7

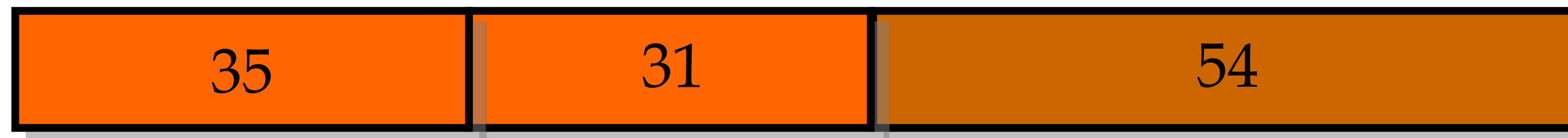
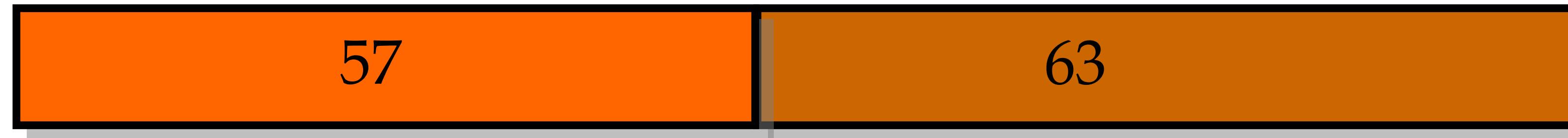


35

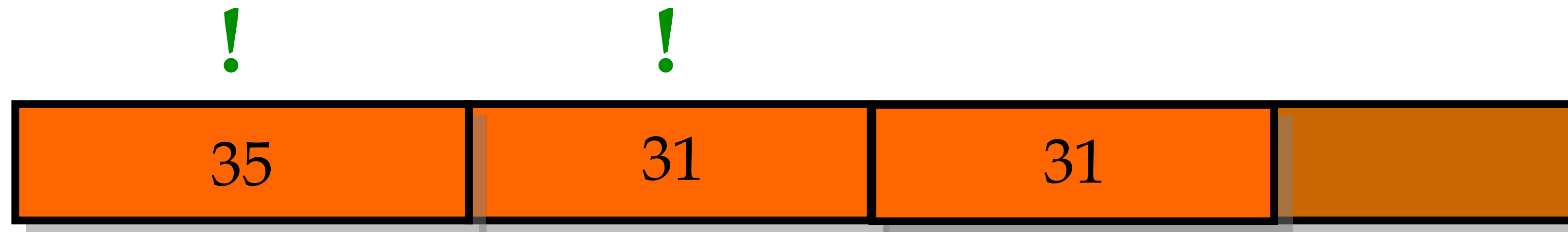
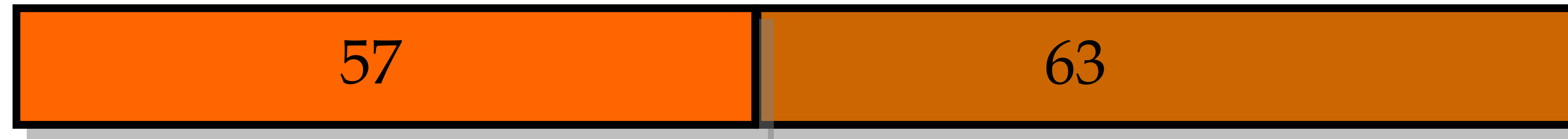
31

85

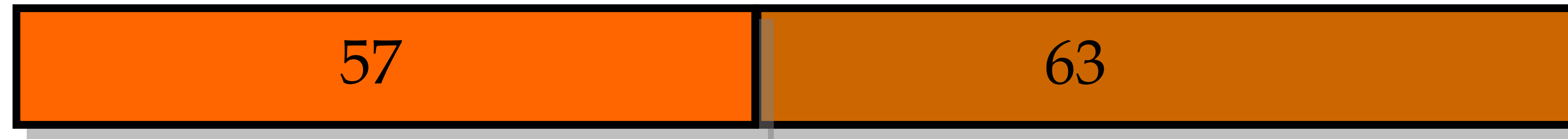
Keine Lösung?!



Keine Lösung?!



Keine Lösung?!



!



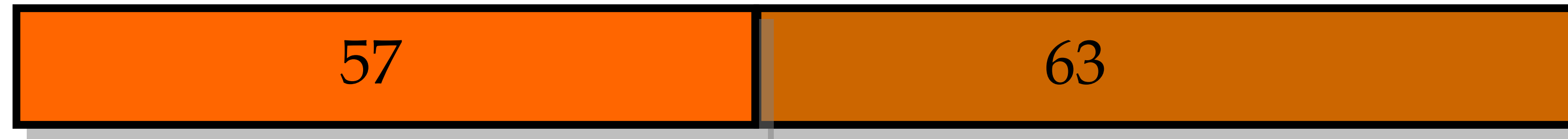
!

!

!



Keine Lösung?!



!



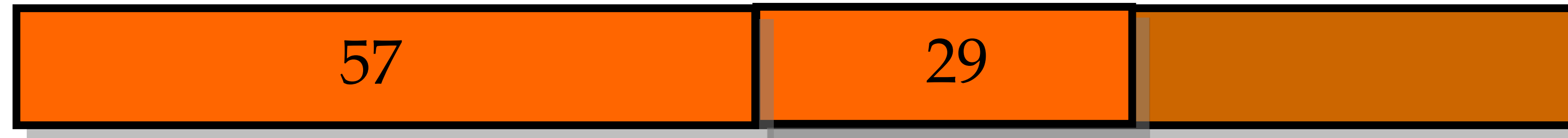
!

!

!



Keine Lösung?!



!

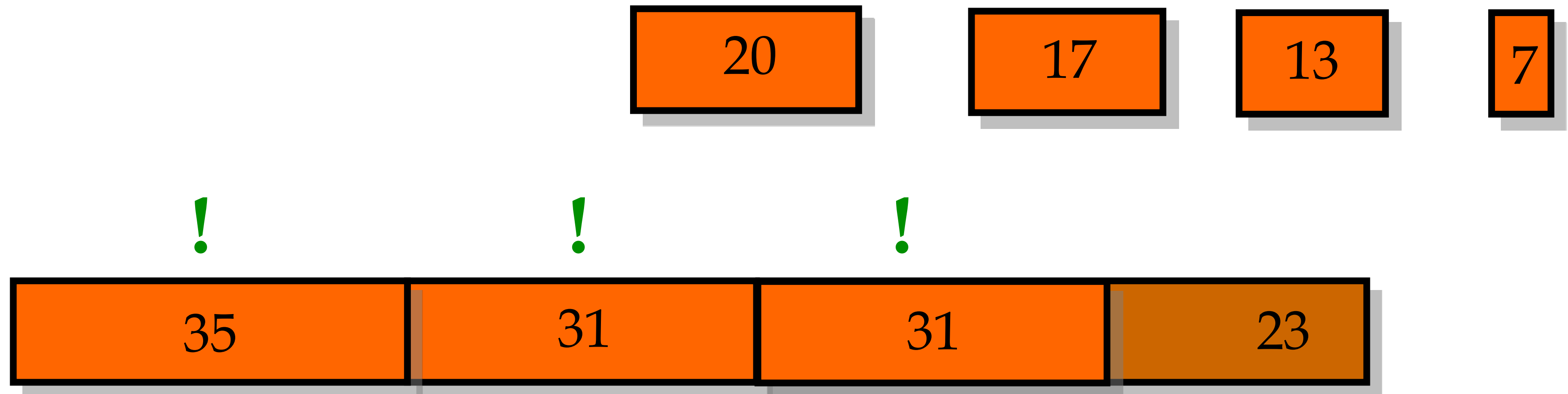
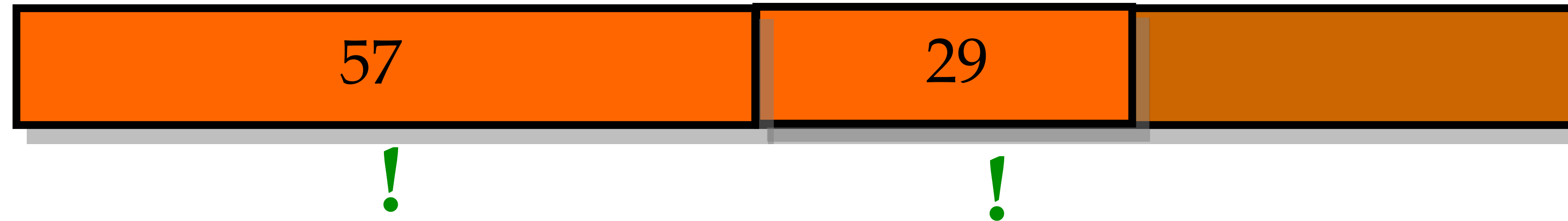


!

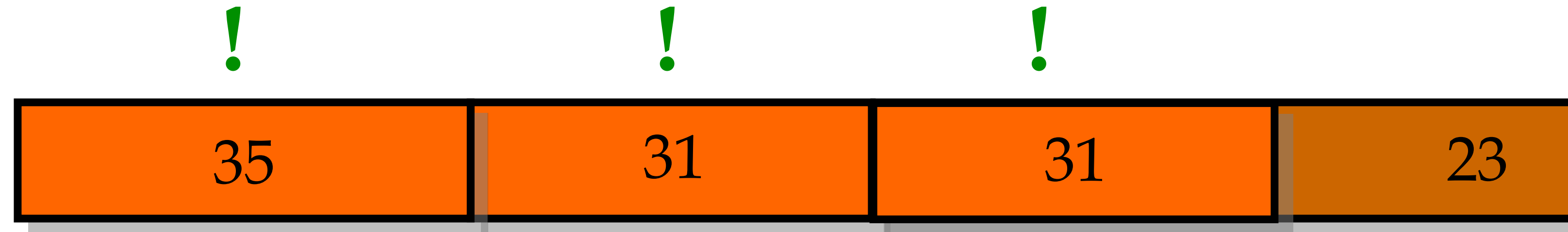
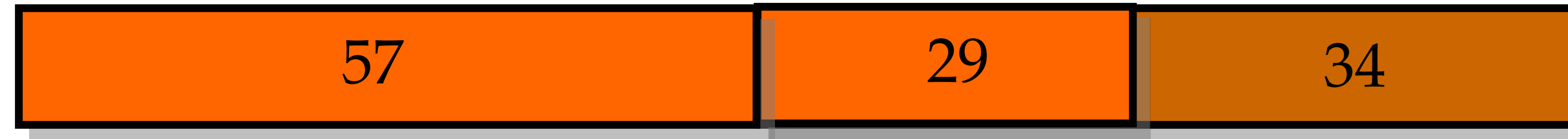
!

!

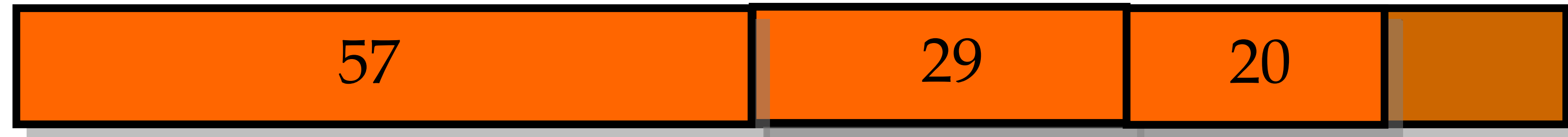
Keine Lösung?!



Keine Lösung?!

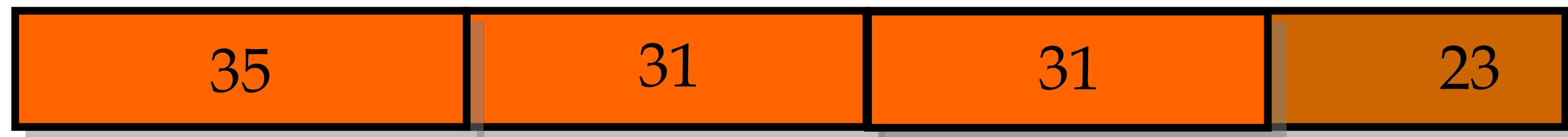


Keine Lösung?!



!

!

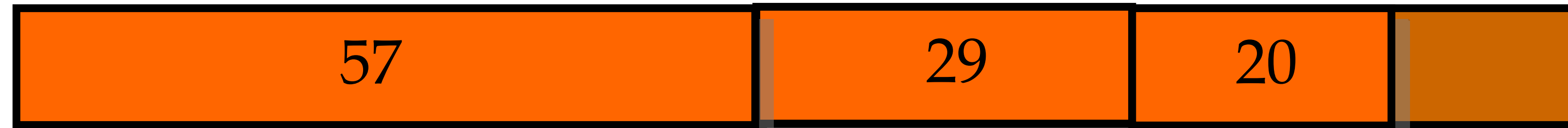


!

!

!

Keine Lösung?!



!

!

!



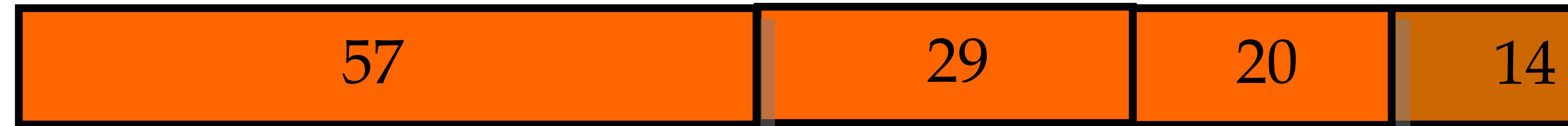
!

!

!



Keine Lösung?!



!

!

!



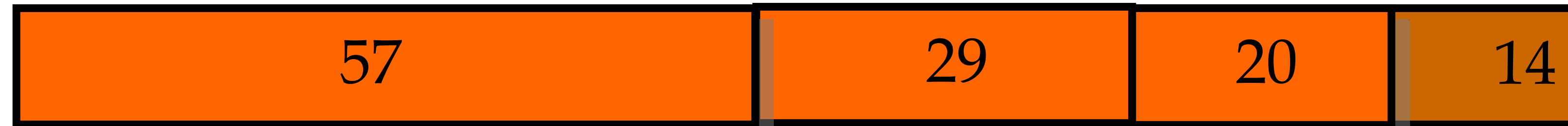
!

!

!



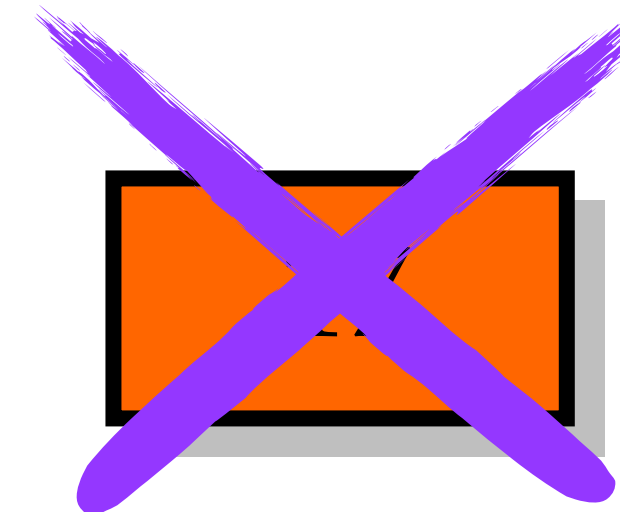
Keine Lösung?!



!

!

!



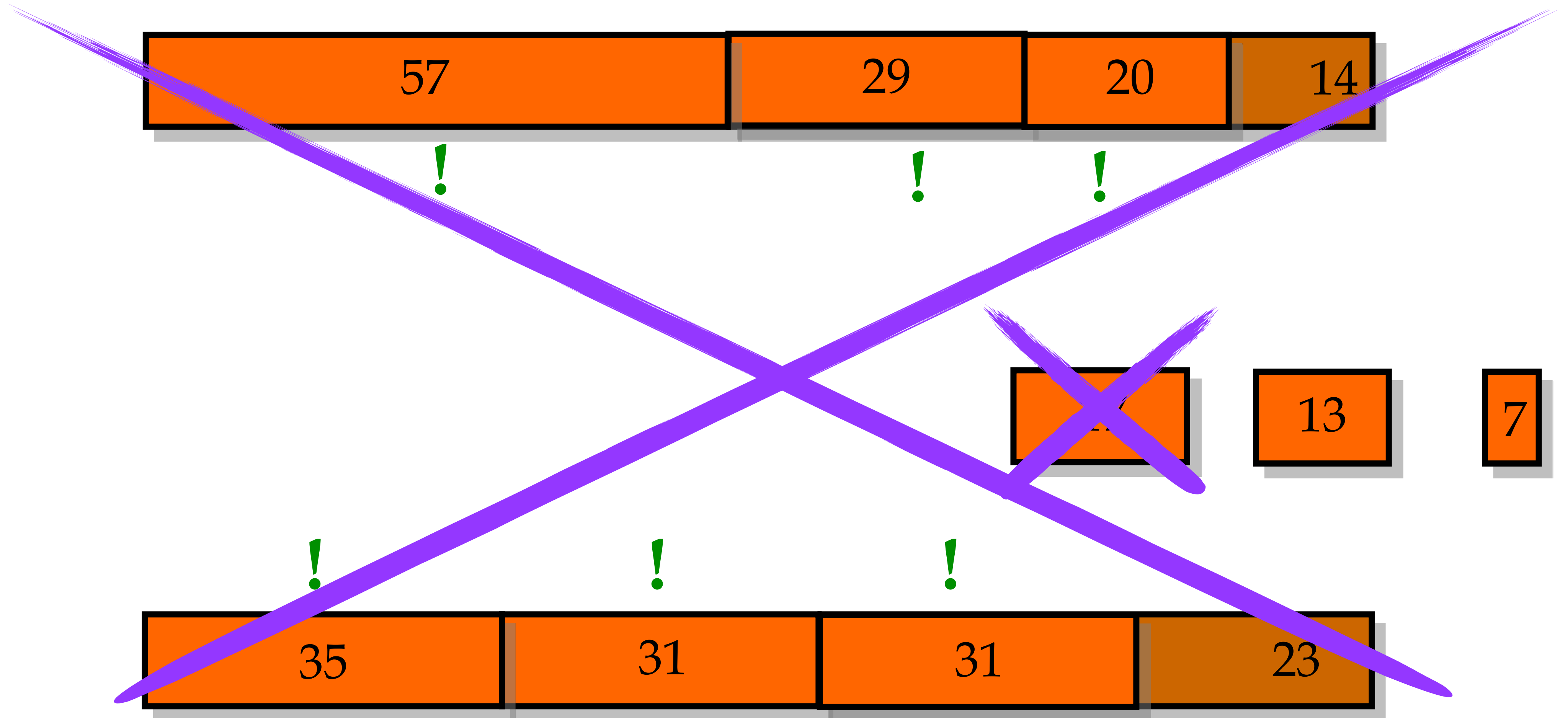
!

!

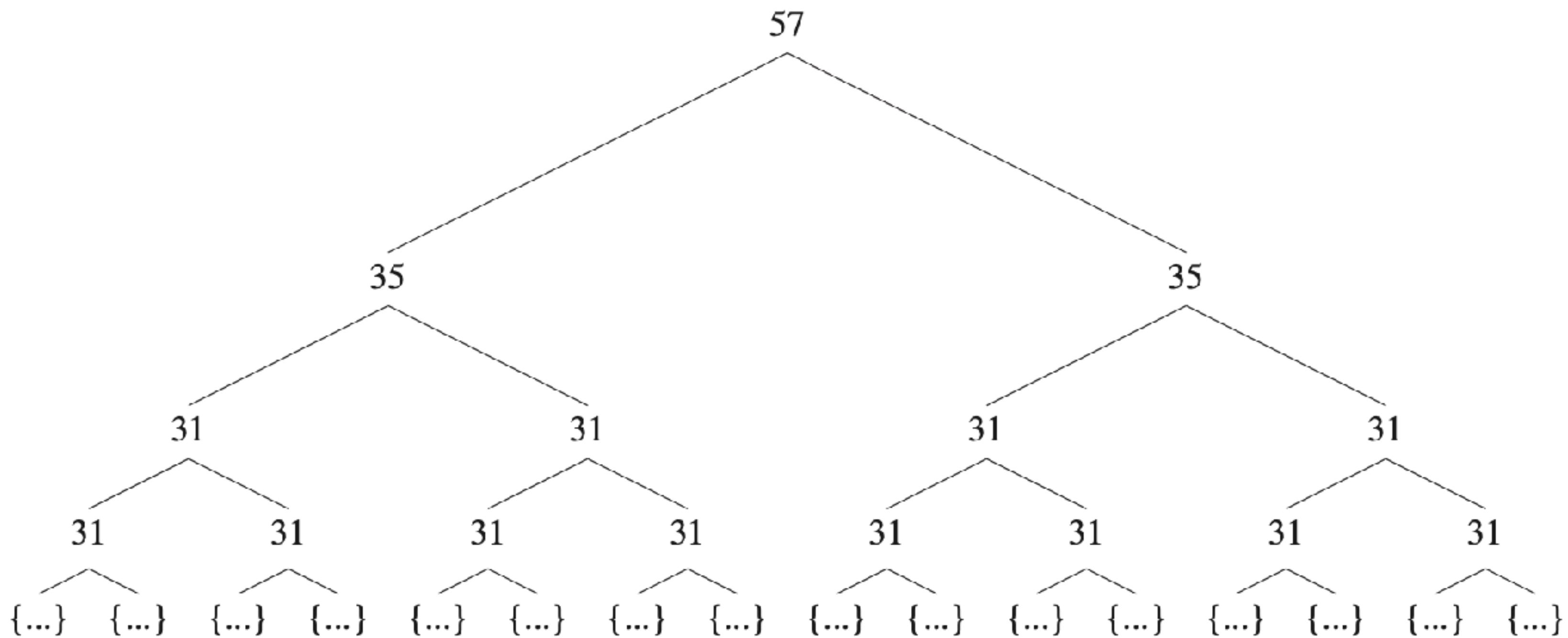
!



Keine Lösung?!

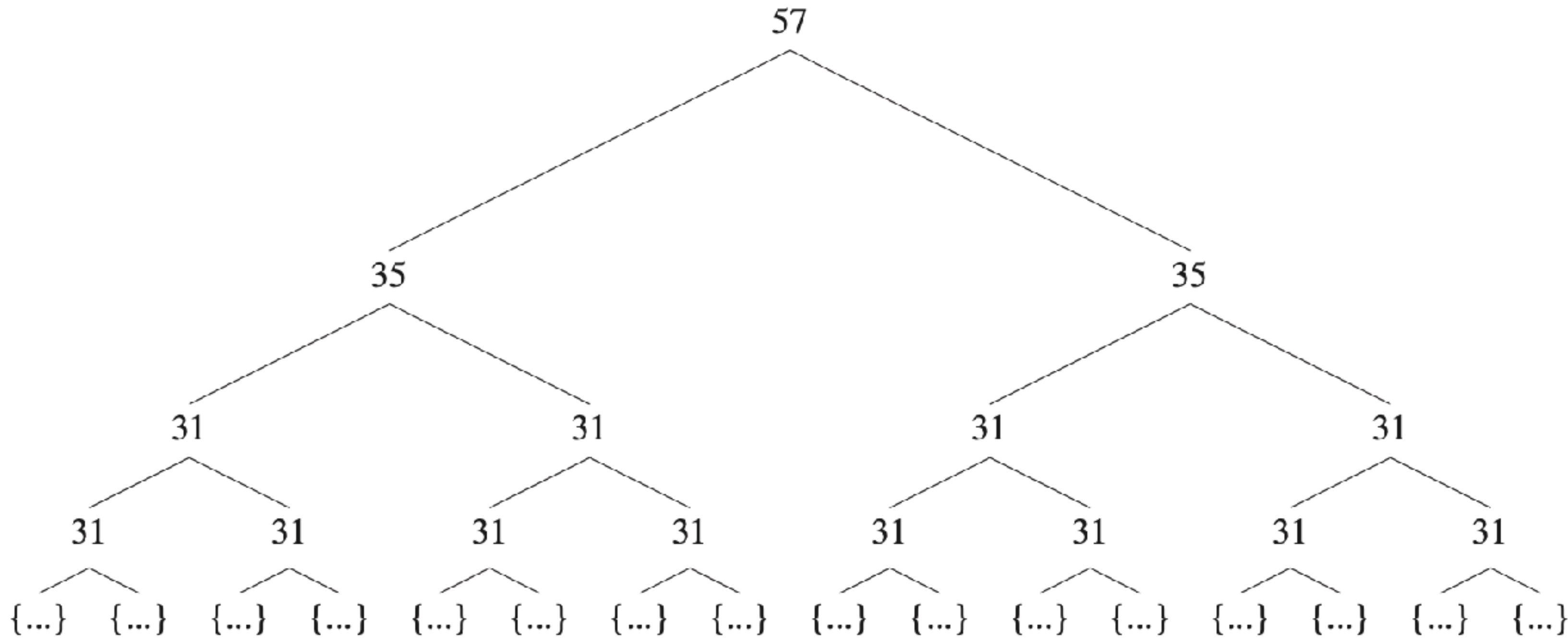


Enumerationsprinzip



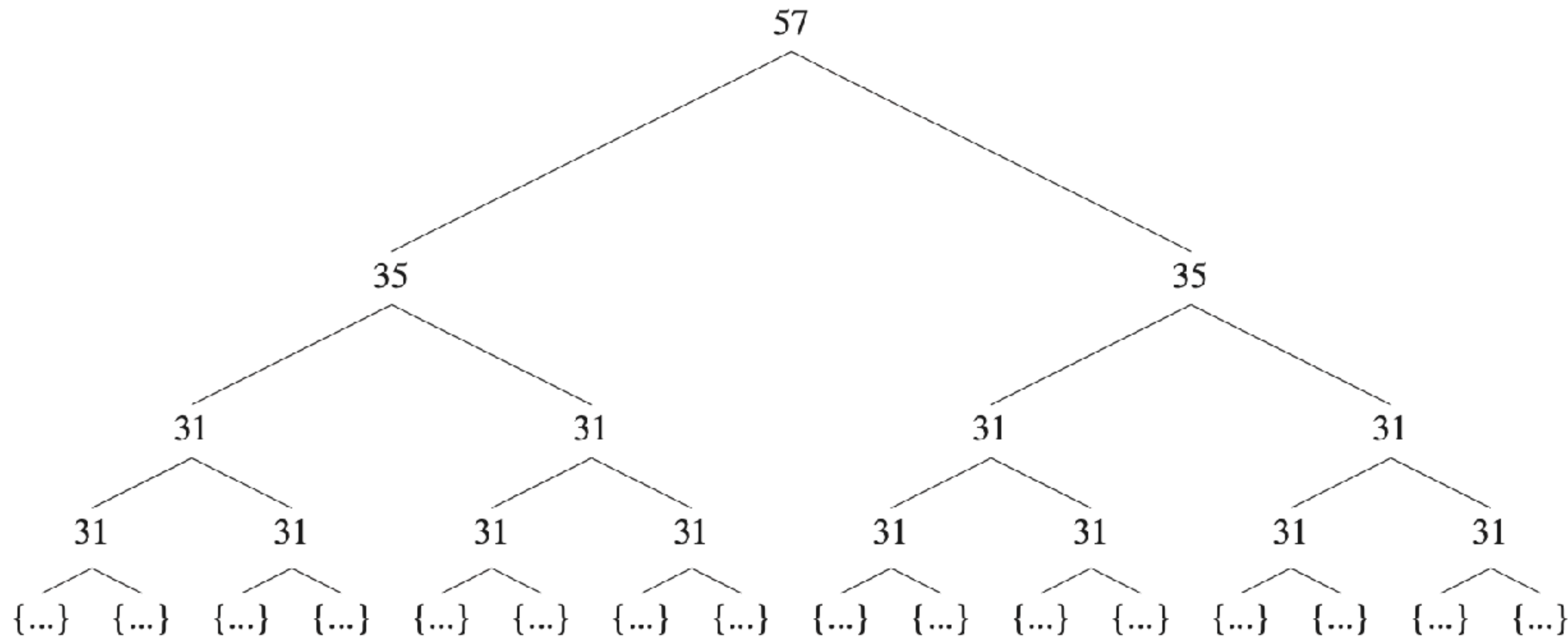
Enumerationsprinzip

120



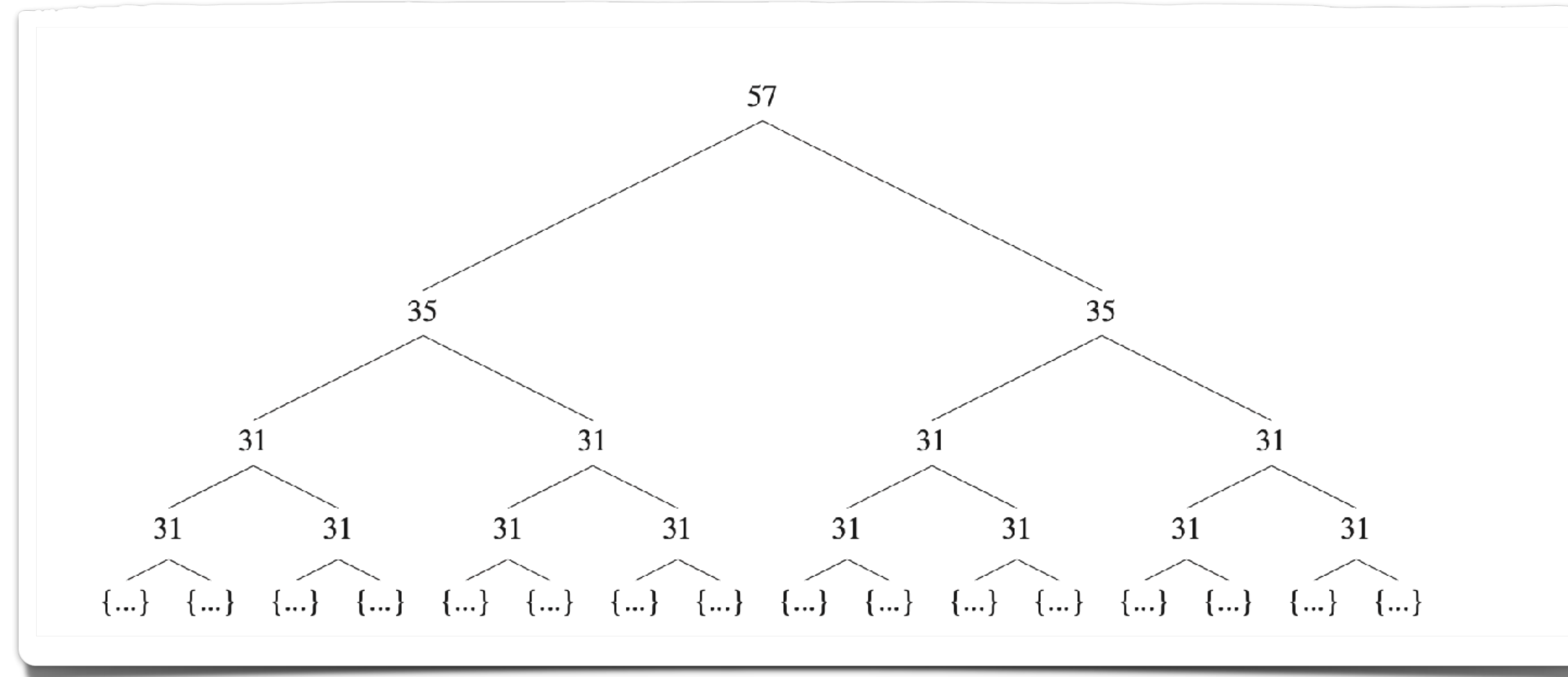
Enumerationsprinzip

120

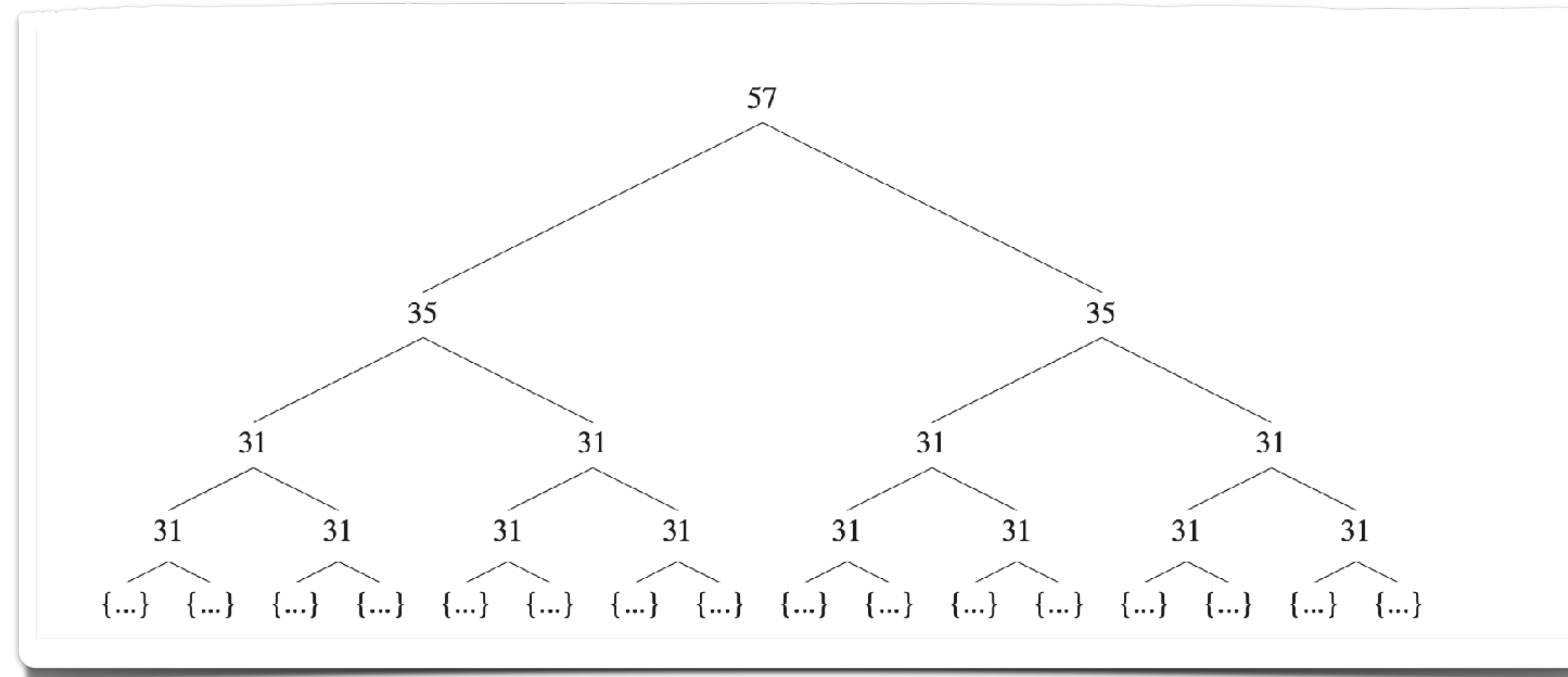


120

Enumerationsprinzip

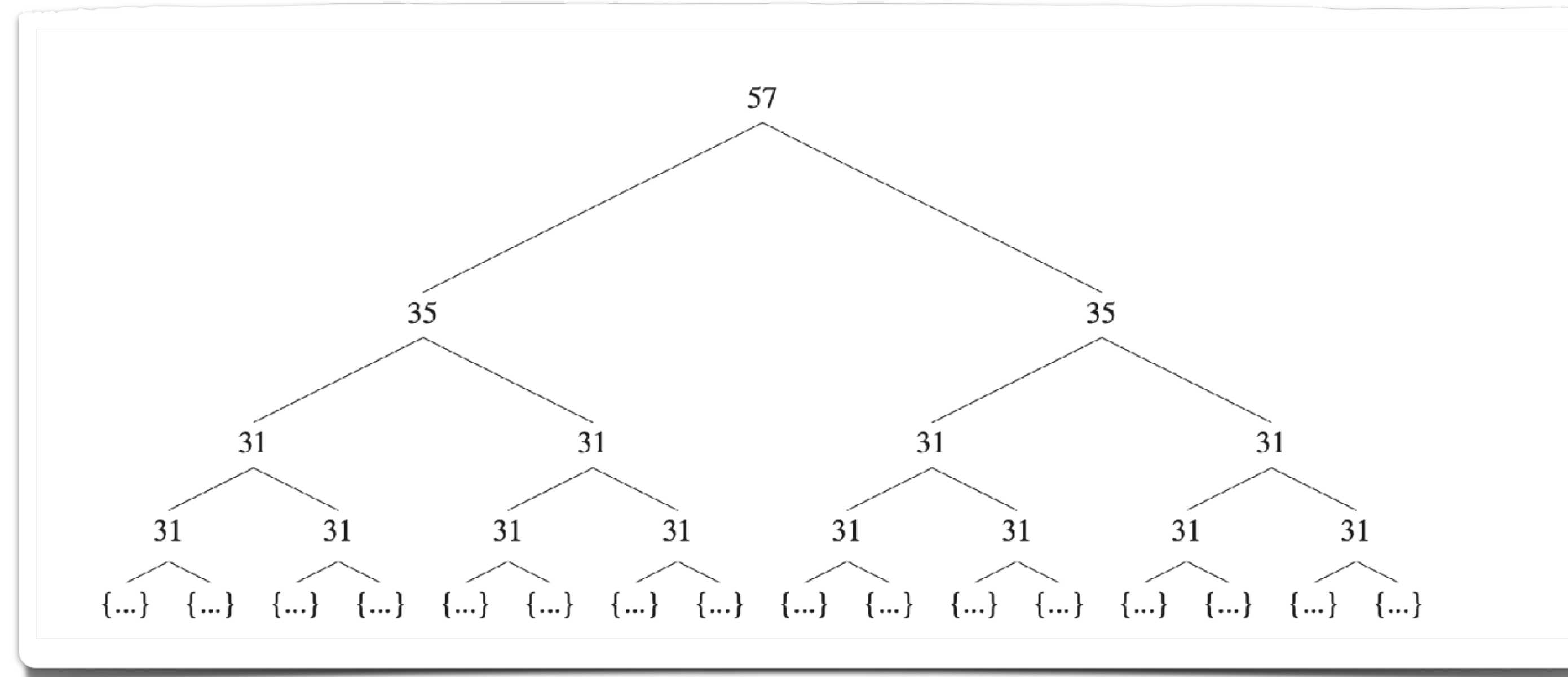


Enumerationsprinzip



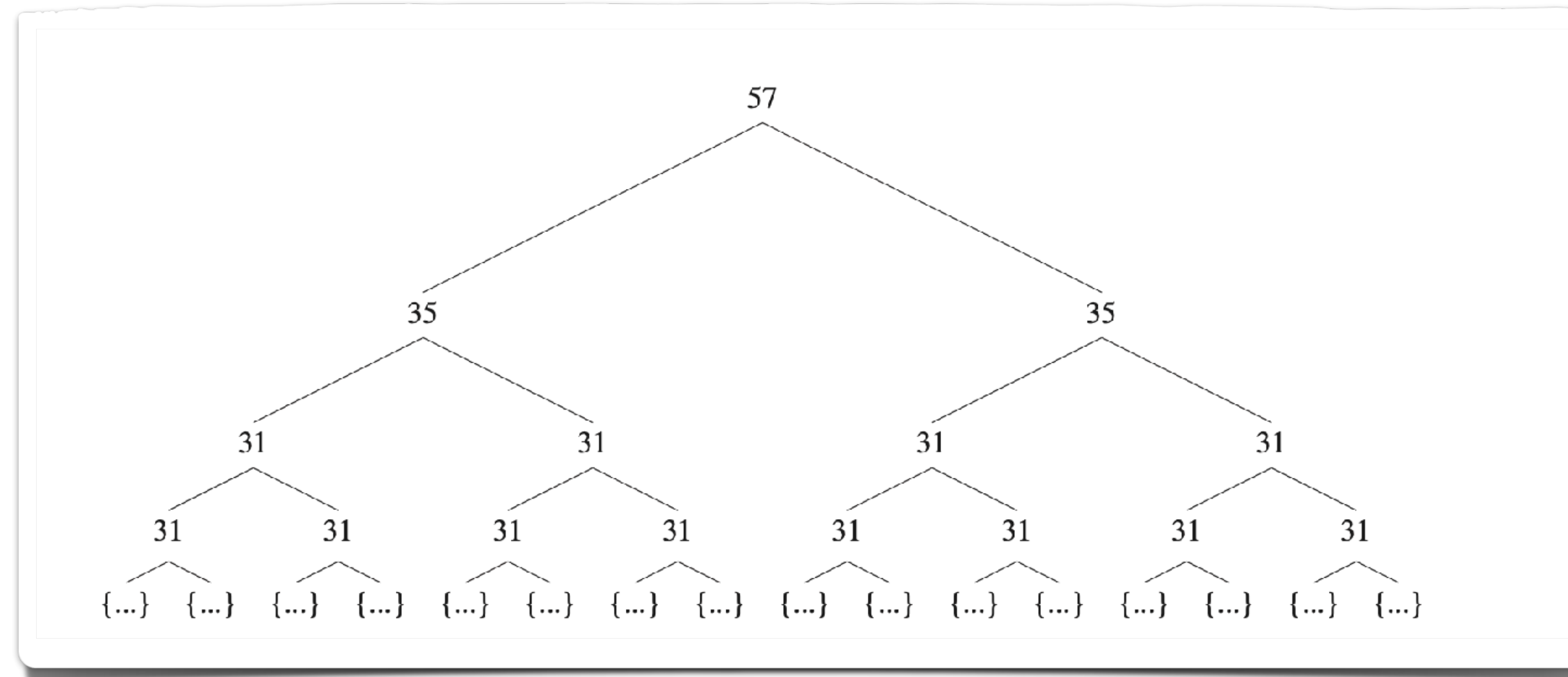
- Exponentiell viele Fälle!

Enumerationsprinzip



- Exponentiell viele Fälle!
- Wie geht das systematisch?

Enumerationsprinzip



- Exponentiell viele Fälle!
- Wie geht das systematisch?
- Wo kann man Arbeit sparen?

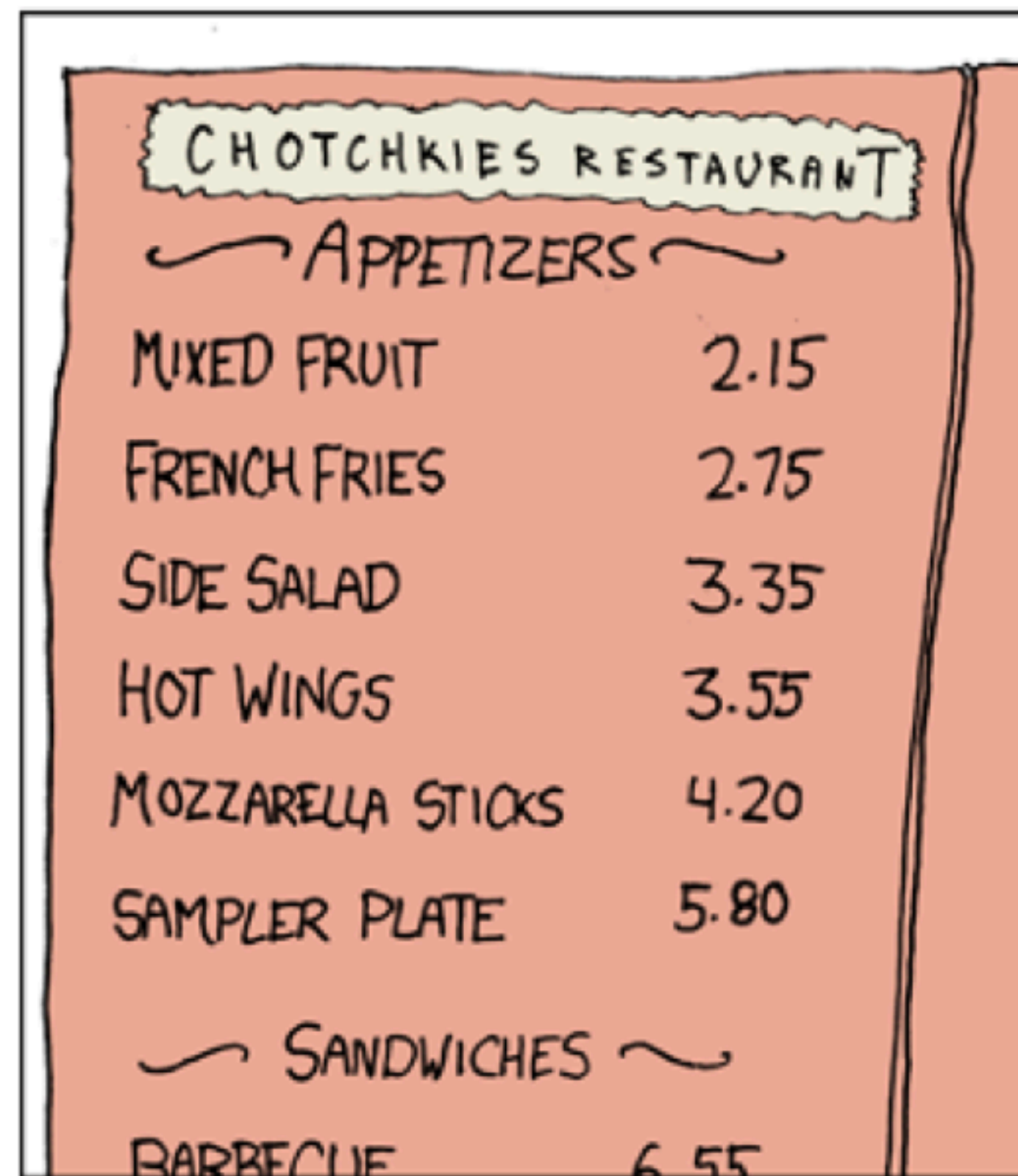
Alltagsanwendung: xkcd #287

Alltagsanwendung: xkcd #287

MY HOBBY:
EMBEDDING NP-COMPLETE PROBLEMS IN RESTAURANT ORDERS

Alltagsanwendung: xkcd #287

MY HOBBY:
EMBEDDING NP-COMPLETE PROBLEMS IN RESTAURANT ORDERS



A hand-drawn menu for 'CHOTCHKIES RESTAURANT'. The menu is divided into two sections: 'APPETIZERS' and 'SANDWICHES'. The appetizers section lists six items with their prices: Mixed Fruit (2.15), French Fries (2.75), Side Salad (3.35), Hot Wings (3.55), Mozzarella Sticks (4.20), and Sampler Plate (5.80). The sandwiches section lists one item: Barbecue (6.55). The menu is drawn in a simple, sketchy style with a light orange background for the menu area.

CHOTCHKIES RESTAURANT	
APPETIZERS	
MIXED FRUIT	2.15
FRENCH FRIES	2.75
SIDE SALAD	3.35
HOT WINGS	3.55
MOZZARELLA STICKS	4.20
SAMPLER PLATE	5.80
SANDWICHES	
BARBECUE	6.55

Alltagsanwendung: xkcd #287

MY HOBBY:
EMBEDDING NP-COMPLETE PROBLEMS IN RESTAURANT ORDERS

CHOTCHKIES RESTAURANT

APPETIZERS

MIXED FRUIT	2.15
FRENCH FRIES	2.75
SIDE SALAD	3.35
HOT WINGS	3.55
MOZZARELLA STICKS	4.20
SAMPLER PLATE	5.80

SANDWICHES

BARBECUE	6.55
----------	------



Vielen Dank!

s.fekete@tu-bs.de