Forelesning 3

Splitt & hersk

Spionproblemet

Spionproblemet

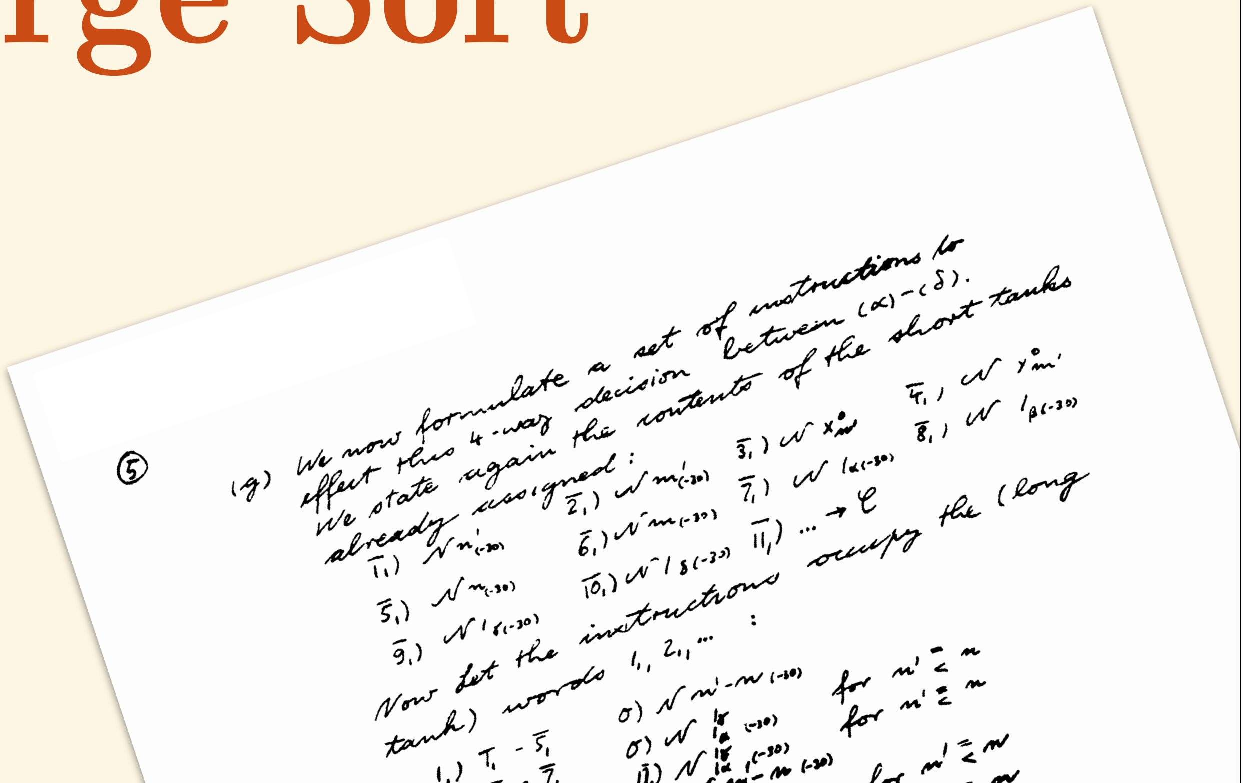
- **En forsamling folk**
- **Noen kjenner andre**
- **Det er ikke nødvendigvis gjensidig**
- **En spion kjenner alle, men ingen kjenner spionen**
- **Har forsamlingen en spion?**

Callback corner

«[Von Neumann's] manuscript, written in ink, is 23 pages long; the first page still shows traces of the penciled phrase "TOP SECRET," which was subsequently erased. (In 1945, work on computers was classified, due to its connections with military problems.)»

Donald Knuth, «Von Neumann's First Computer Program»
http://dl.acm.org/citation.cfm?doid=356580.356581

Merge Sort



$\textsc{Merge}(A, p, q, r)$

MERGE(A, p, q, r)

$n_1 = q - p + 1$

MERGE(A, p, q, r)

```
n_1 = q - p + 1
n_2 = r - q
```

MERGE(A, p, q, r)

$n_1 = q - p + 1$

$n_2 = r - q$

let $L[1\,..\,n_1 + 1]$ and $R[1\,..\,n_2 + 1]$ be new arrays

MERGE(A, p, q, r)

$n_1 = q - p + 1$

$n_2 = r - q$

let $L[1 \,..\, n_1 + 1]$ and $R[1 \,..\, n_2 + 1]$ be new arrays

for $i = 1$ **to** n_1

MERGE(A, p, q, r)

$n_1 = q - p + 1$

$n_2 = r - q$

let $L[1 \,..\, n_1 + 1]$ and $R[1 \,..\, n_2 + 1]$ be new arrays

for $i = 1$ **to** n_1

    $L[i] = A[p + i - 1]$

MERGE(A, p, q, r)

$n_1 = q - p + 1$
$n_2 = r - q$
let $L[1 \,..\, n_1 + 1]$ and $R[1 \,..\, n_2 + 1]$ be new arrays
for $i = 1$ **to** n_1
        $L[i] = A[p + i - 1]$
for $j = 1$ **to** n_2

MERGE(A, p, q, r)

$n_1 = q - p + 1$
$n_2 = r - q$
let $L[1 \,..\, n_1 + 1]$ and $R[1 \,..\, n_2 + 1]$ be new arrays
for $i = 1$ **to** n_1
    $L[i] = A[p + i - 1]$
for $j = 1$ **to** n_2
    $R[j] = A[q + j]$

MERGE(A, p, q, r)

$n_1 = q - p + 1$
$n_2 = r - q$
let $L[1\,..\,n_1 + 1]$ and $R[1\,..\,n_2 + 1]$ be new arrays
for $i = 1$ **to** n_1
    $L[i] = A[p + i - 1]$
for $j = 1$ **to** n_2
    $R[j] = A[q + j]$
$L[n_1 + 1] = \infty$

```
MERGE(A, p, q, r)
n_1 = q - p + 1
n_2 = r - q
let L[1 .. n_1 + 1] and R[1 .. n_2 + 1] be new arrays
for i = 1 to n_1
        L[i] = A[p + i - 1]
for j = 1 to n_2
        R[j] = A[q + j]
L[n_1 + 1] = ∞
R[n_2 + 1] = ∞
```

MERGE(A, p, q, r)

$n_1 = q - p + 1$
$n_2 = r - q$
let $L[1 \,..\, n_1 + 1]$ and $R[1 \,..\, n_2 + 1]$ be new arrays
for $i = 1$ **to** n_1
    $L[i] = A[p + i - 1]$
for $j = 1$ **to** n_2
    $R[j] = A[q + j]$
$L[n_1 + 1] = \infty$
$R[n_2 + 1] = \infty$
$i = 1$

```
MERGE(A, p, q, r)
n1 = q - p + 1
n2 = r - q
let L[1 .. n1 + 1] and R[1 .. n2 + 1] be new arrays
for i = 1 to n1
        L[i] = A[p + i - 1]
for j = 1 to n2
        R[j] = A[q + j]
L[n1 + 1] = ∞
R[n2 + 1] = ∞
i = 1
j = 1
```

MERGE(A, p, q, r)

$n_1 = q - p + 1$
$n_2 = r - q$
let $L[1 \,.\,.\, n_1 + 1]$ and $R[1 \,.\,.\, n_2 + 1]$ be new arrays
for $i = 1$ **to** n_1
        $L[i] = A[p + i - 1]$
for $j = 1$ **to** n_2
        $R[j] = A[q + j]$
$L[n_1 + 1] = \infty$
$R[n_2 + 1] = \infty$
$i = 1$
$j = 1$
for $k = p$ **to** r

```
MERGE(A, p, q, r)
n_1 = q - p + 1
n_2 = r - q
let L[1 .. n_1 + 1] and R[1 .. n_2 + 1] be new arrays
for i = 1 to n_1
        L[i] = A[p + i - 1]
for j = 1 to n_2
        R[j] = A[q + j]
L[n_1 + 1] = ∞
R[n_2 + 1] = ∞
i = 1
j = 1
for k = p to r
        if L[i] ≤ R[j]
```

```
MERGE(A, p, q, r)
n1 = q - p + 1
n2 = r - q
let L[1..n1 + 1] and R[1..n2 + 1] be new arrays
for i = 1 to n1
    L[i] = A[p + i - 1]
for j = 1 to n2
    R[j] = A[q + j]
L[n1 + 1] = ∞
R[n2 + 1] = ∞
i = 1
j = 1
for k = p to r
    if L[i] ≤ R[j]
        A[k] = L[i]
```

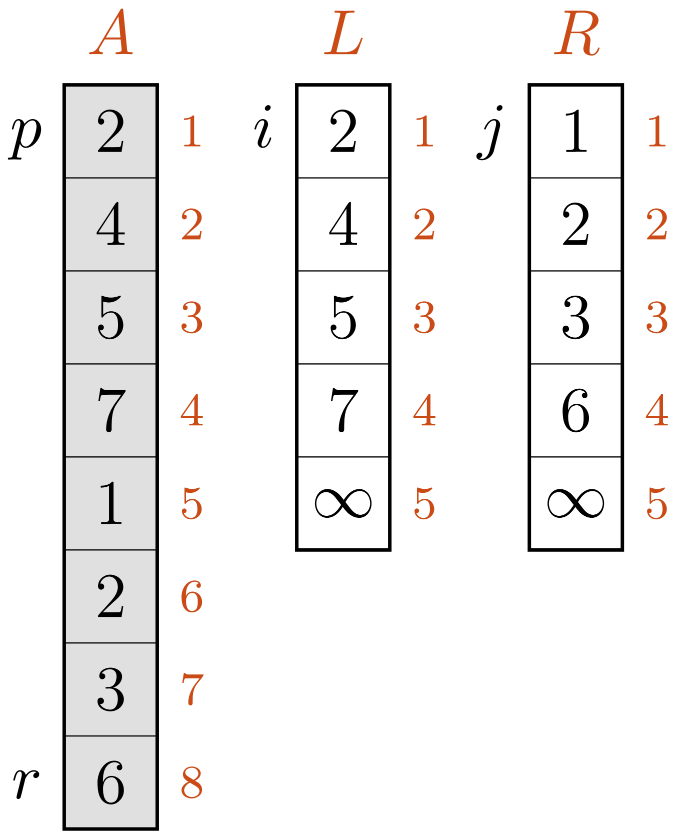
```
MERGE(A, p, q, r)
n1 = q - p + 1
n2 = r - q
let L[1 .. n1 + 1] and R[1 .. n2 + 1] be new arrays
for i = 1 to n1
        L[i] = A[p + i - 1]
for j = 1 to n2
        R[j] = A[q + j]
L[n1 + 1] = ∞
R[n2 + 1] = ∞
i = 1
j = 1
for k = p to r
        if L[i] ≤ R[j]
                A[k] = L[i]
                i = i + 1
```

MERGE(A, p, q, r)

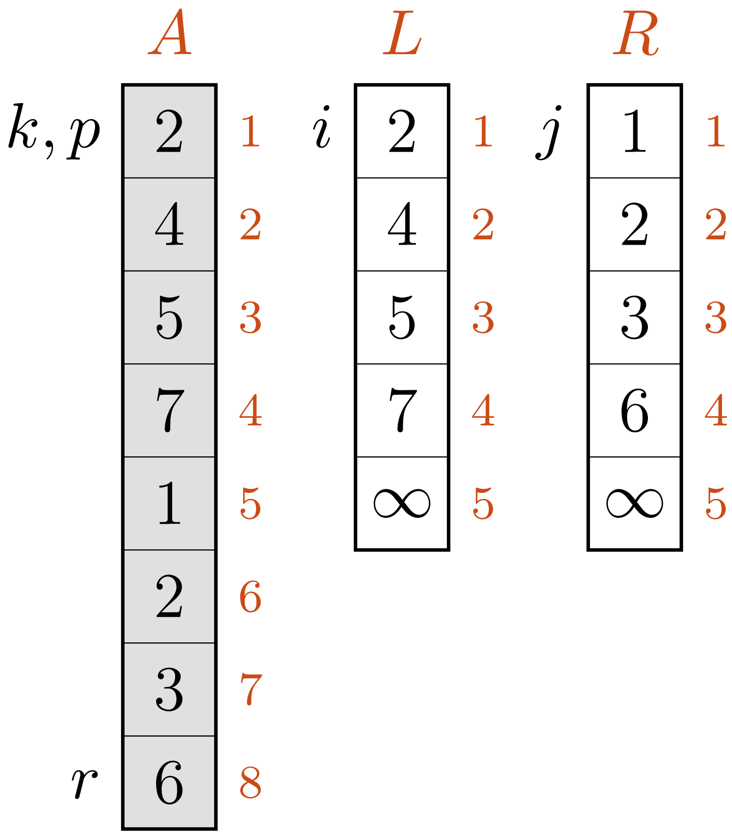
```
n1 = q - p + 1
n2 = r - q
let L[1 .. n1 + 1] and R[1 .. n2 + 1] be new arrays
for i = 1 to n1
        L[i] = A[p + i - 1]
for j = 1 to n2
        R[j] = A[q + j]
L[n1 + 1] = ∞
R[n2 + 1] = ∞
i = 1
j = 1
for k = p to r
        if L[i] ≤ R[j]
                A[k] = L[i]
                i = i + 1
        else A[k] = R[j]
```

```
MERGE(A, p, q, r)
n1 = q − p + 1
n2 = r − q
let L[1 .. n1 + 1] and R[1 .. n2 + 1] be new arrays
for i = 1 to n1
        L[i] = A[p + i − 1]
for j = 1 to n2
        R[j] = A[q + j]
L[n1 + 1] = ∞
R[n2 + 1] = ∞
i = 1
j = 1
for k = p to r
        if L[i] ≤ R[j]
                A[k] = L[i]
                i = i + 1
        else A[k] = R[j]
                j = j + 1
```

```
MERGE(A, p, q, r)
  copy into L and R
  for k = p to r
      if L[i] ≤ R[j]
          A[k] = L[i]
          i = i + 1
      else A[k] = R[j]
          j = j + 1
```

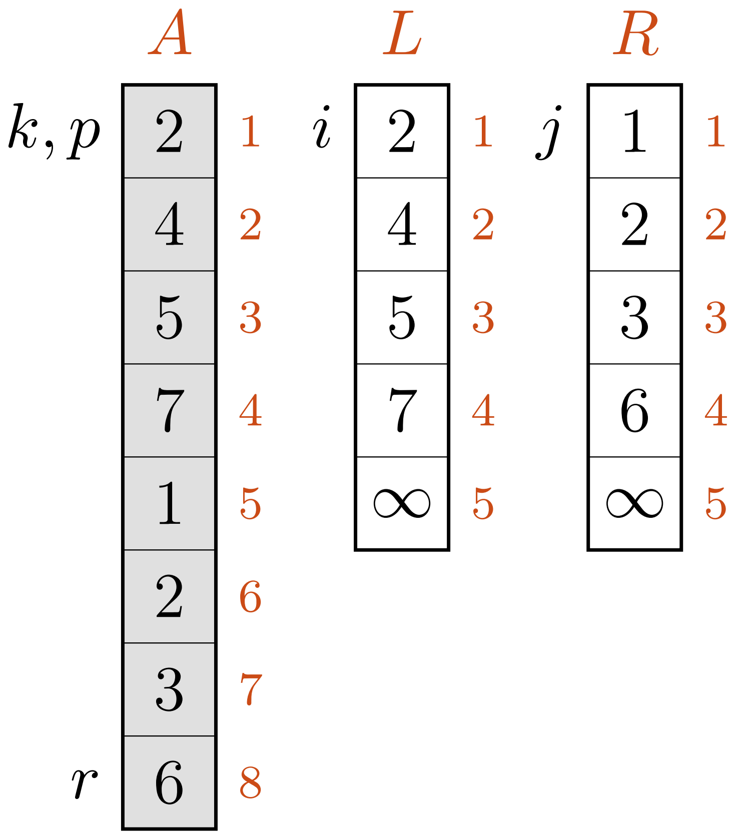


```
MERGE(A, p, q, r)
  copy into L and R
  for k = p to r
      if L[i] ≤ R[j]
          A[k] = L[i]
          i = i + 1
      else A[k] = R[j]
          j = j + 1
```



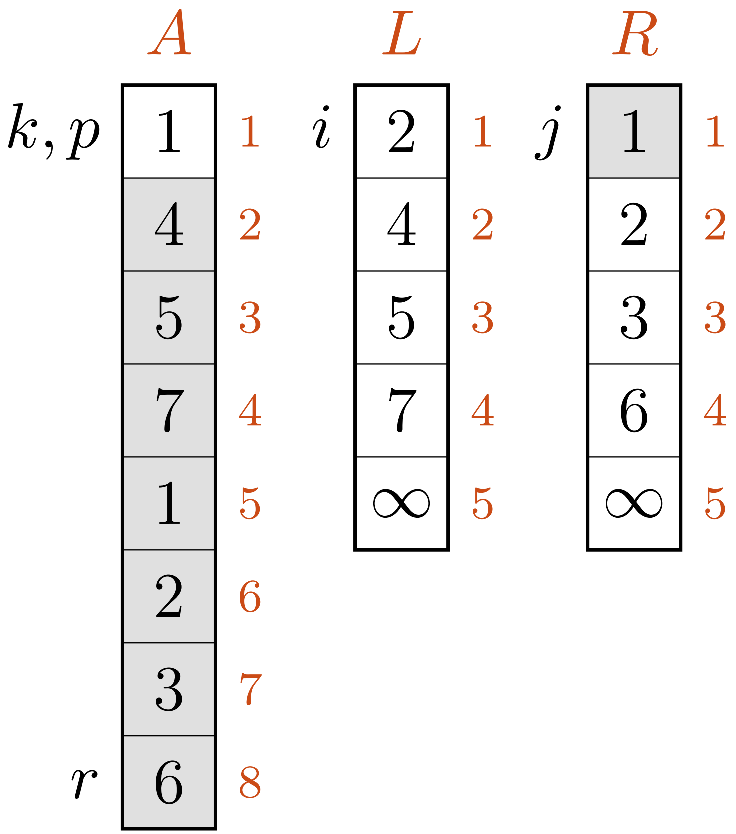
MERGE(A, p, q, r)

```
copy into L and R
for k = p to r
    if L[i] ≤ R[j]
        A[k] = L[i]
        i = i + 1
    else A[k] = R[j]
        j = j + 1
```

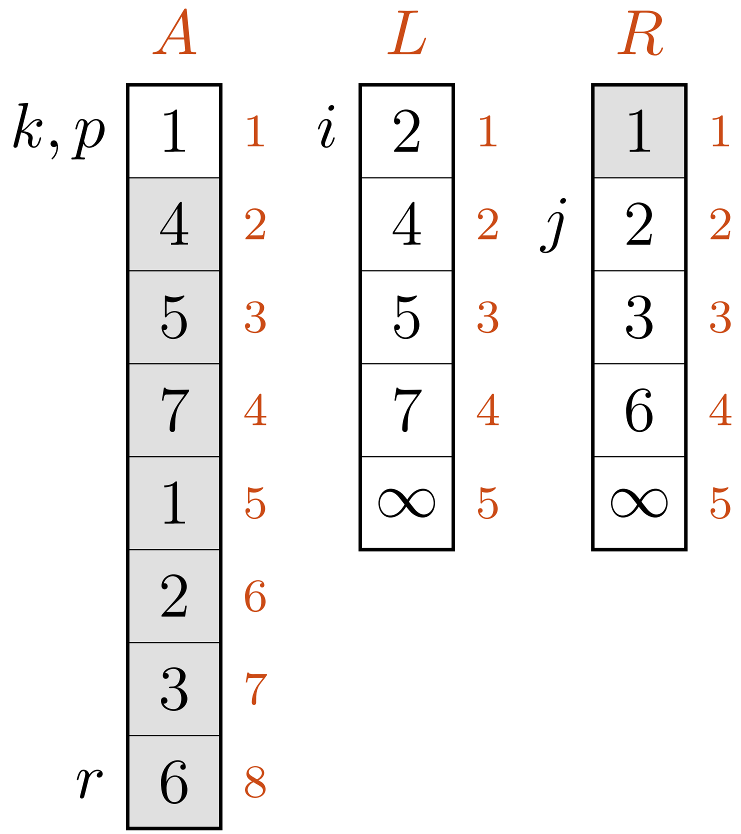


MERGE(A, p, q, r)

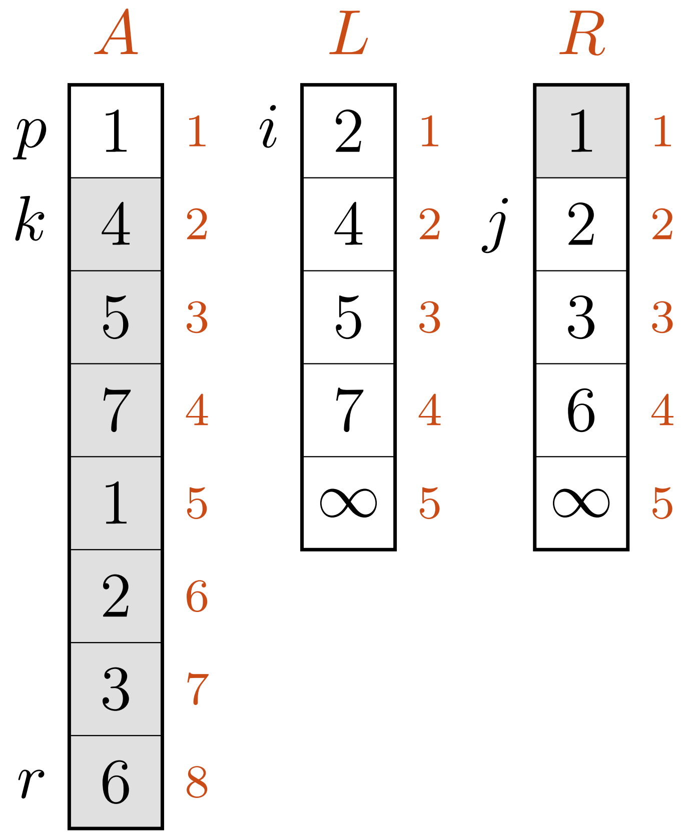
copy into L and R
for $k = p$ **to** r
    if $L[i] \leq R[j]$
        $A[k] = L[i]$
        $i = i + 1$
    else $A[k] = R[j]$
        $j = j + 1$



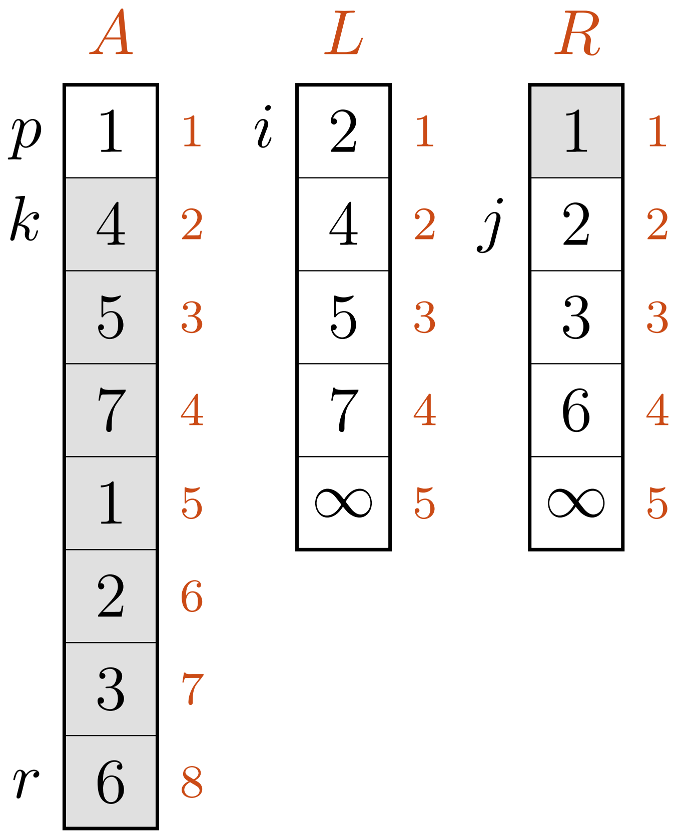
```
MERGE(A, p, q, r)
    copy into L and R
    for k = p to r
        if L[i] ≤ R[j]
            A[k] = L[i]
            i = i + 1
        else A[k] = R[j]
            j = j + 1
```



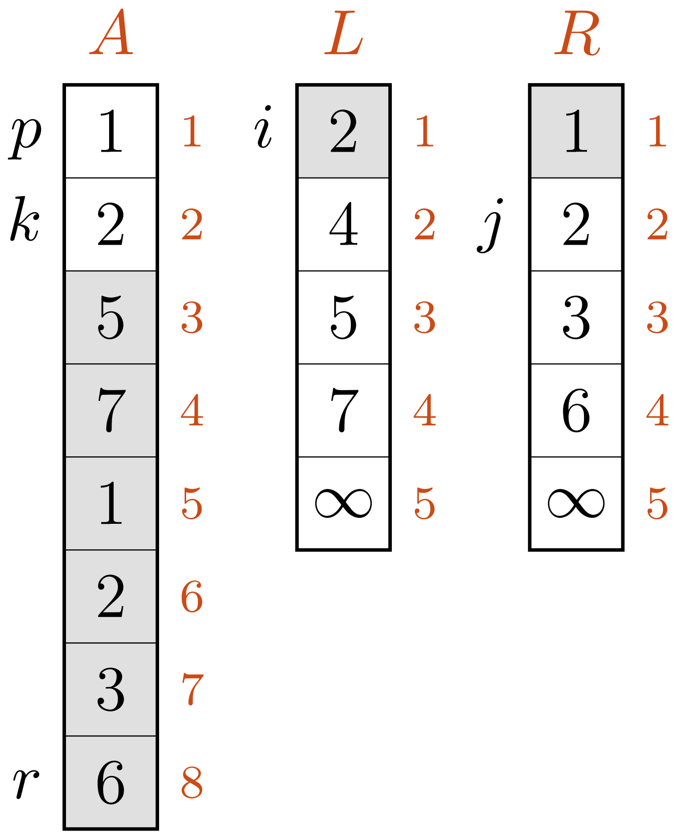
```
MERGE(A, p, q, r)
  copy into L and R
  for k = p to r
      if L[i] ≤ R[j]
          A[k] = L[i]
          i = i + 1
      else A[k] = R[j]
          j = j + 1
```



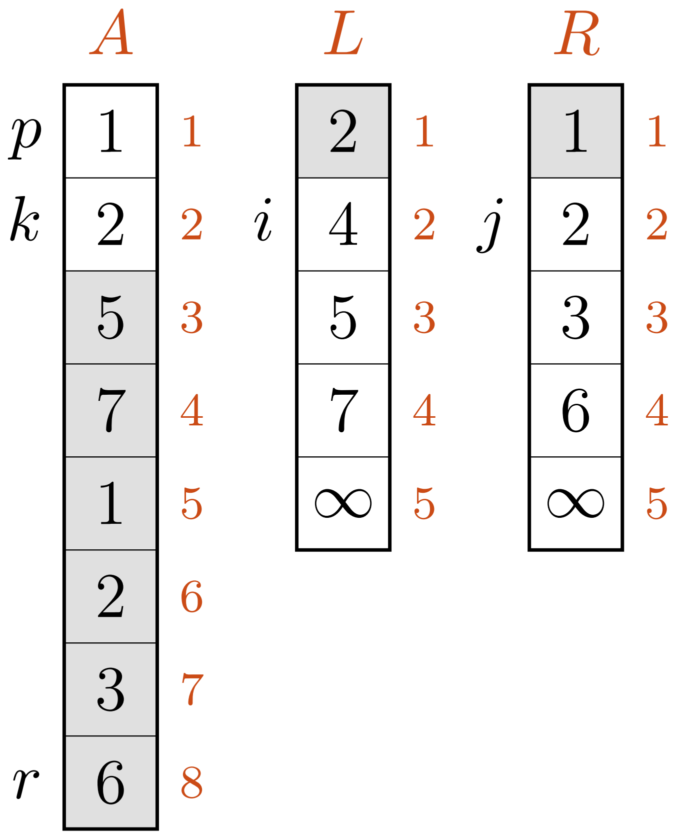
```
MERGE(A, p, q, r)
    copy into L and R
    for k = p to r
        if L[i] ≤ R[j]
            A[k] = L[i]
            i = i + 1
        else A[k] = R[j]
            j = j + 1
```



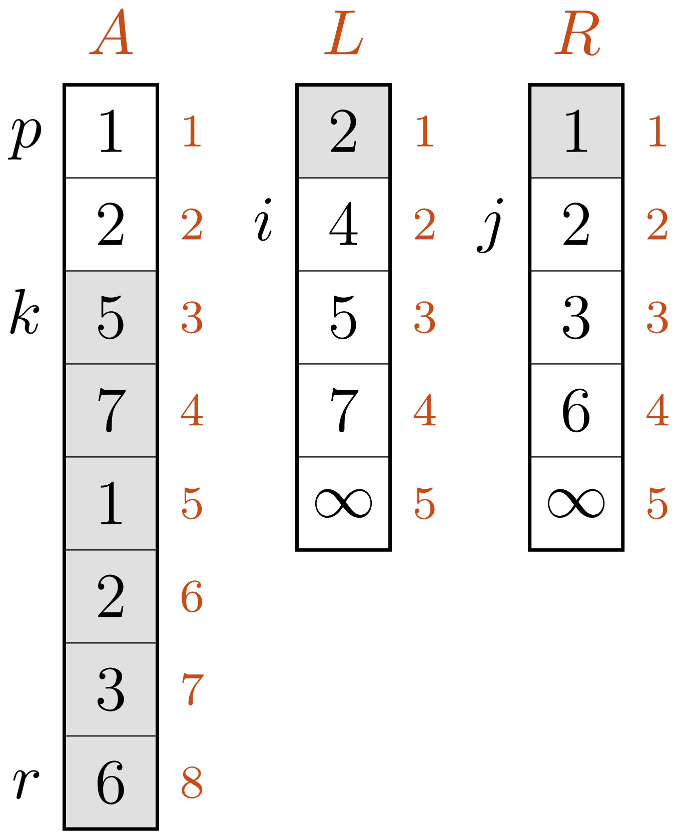
```
Merge(A, p, q, r)
    copy into L and R
    for k = p to r
        if L[i] ≤ R[j]
            A[k] = L[i]
            i = i + 1
        else A[k] = R[j]
            j = j + 1
```



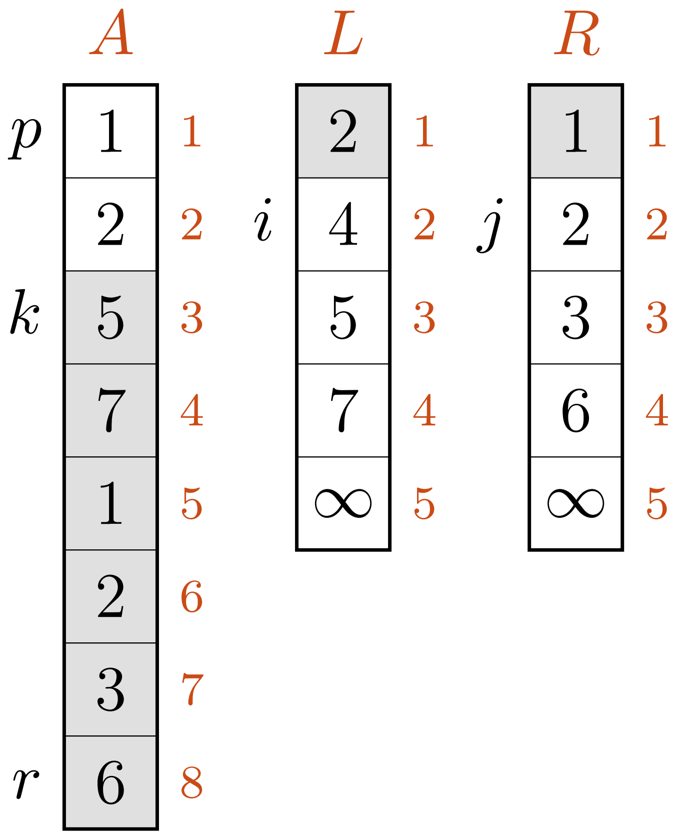
```
MERGE(A, p, q, r)
  copy into L and R
  for k = p to r
      if L[i] ≤ R[j]
          A[k] = L[i]
          i = i + 1
      else A[k] = R[j]
          j = j + 1
```



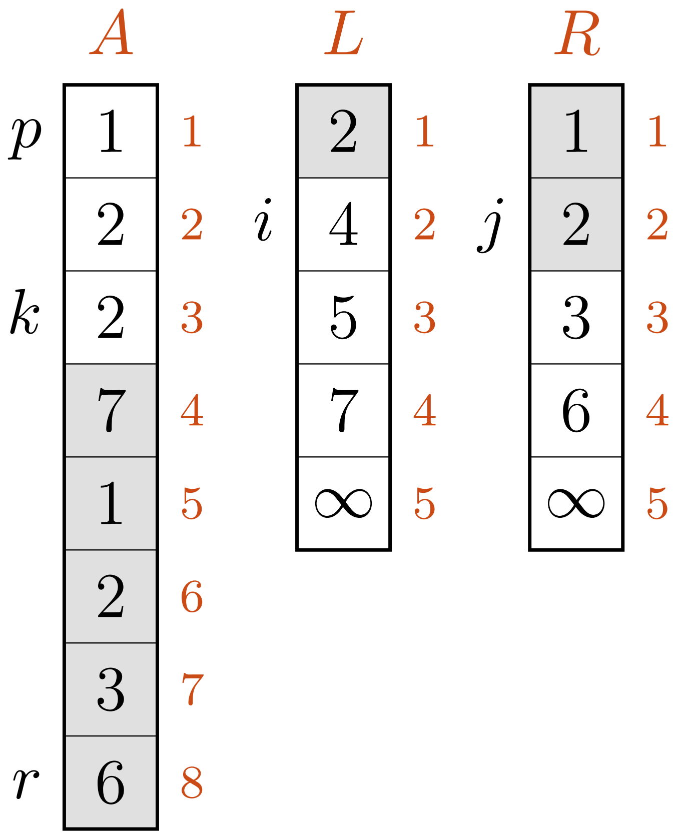
```
MERGE(A, p, q, r)
    copy into L and R
    for k = p to r
        if L[i] ≤ R[j]
            A[k] = L[i]
            i = i + 1
        else A[k] = R[j]
            j = j + 1
```



```
MERGE(A, p, q, r)
  copy into L and R
  for k = p to r
      if L[i] ≤ R[j]
          A[k] = L[i]
          i = i + 1
      else A[k] = R[j]
          j = j + 1
```

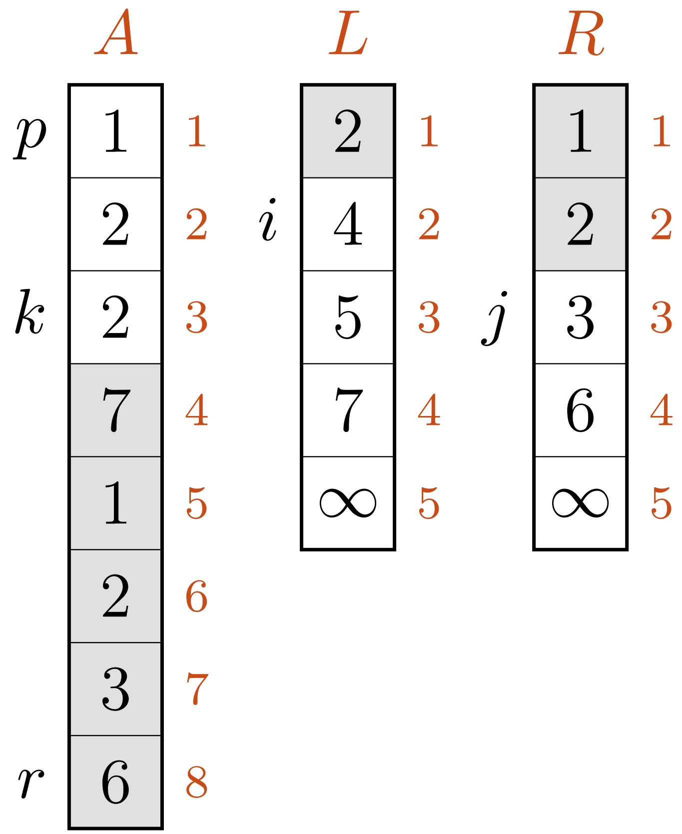


```
MERGE(A, p, q, r)
copy into L and R
for k = p to r
    if L[i] ≤ R[j]
        A[k] = L[i]
        i = i + 1
    else A[k] = R[j]
        j = j + 1
```

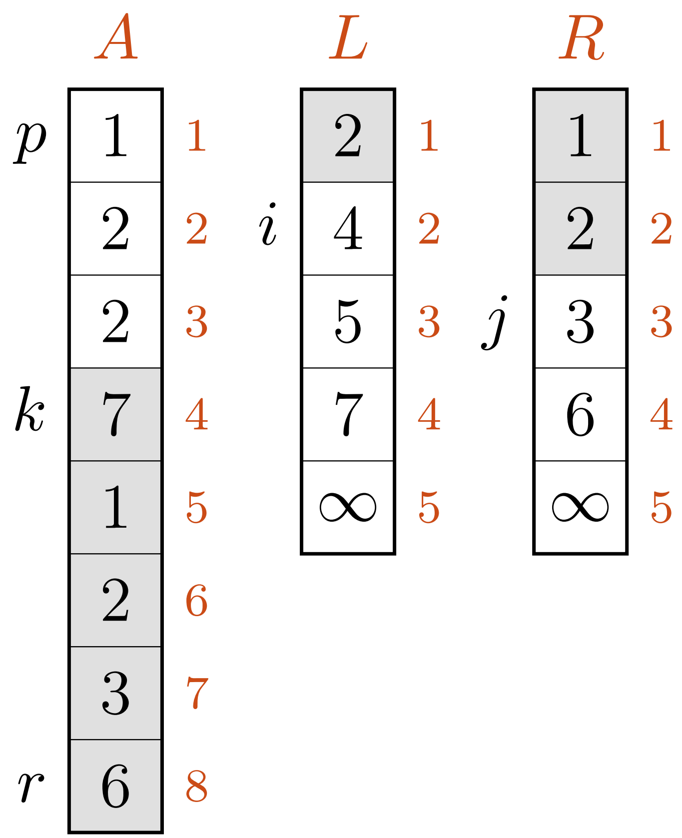


MERGE(A, p, q, r)

```
copy into L and R
for k = p to r
        if L[i] ≤ R[j]
                A[k] = L[i]
                i = i + 1
        else A[k] = R[j]
                j = j + 1
```

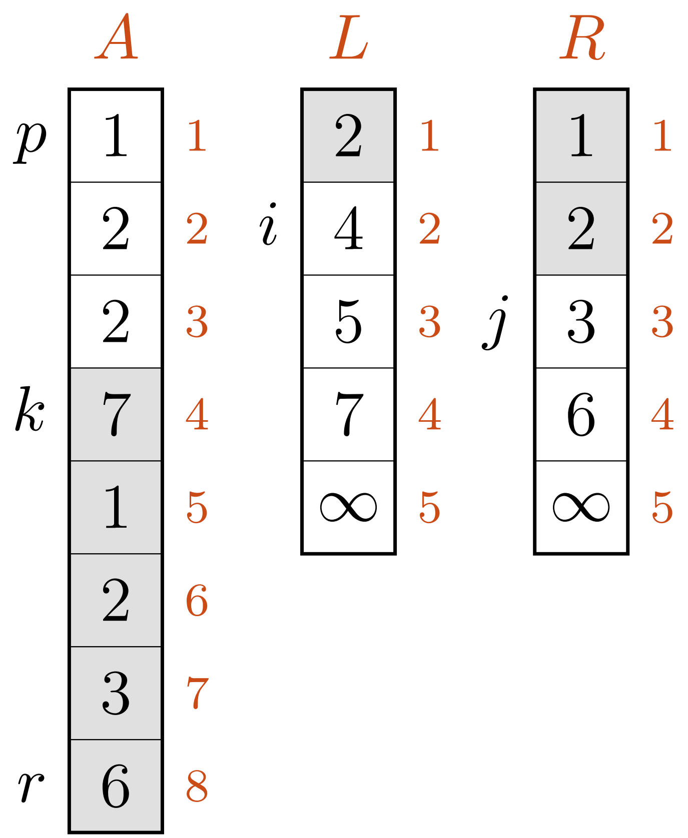


```
MERGE(A, p, q, r)
copy into L and R
for k = p to r
    if L[i] ≤ R[j]
        A[k] = L[i]
        i = i + 1
    else A[k] = R[j]
        j = j + 1
```

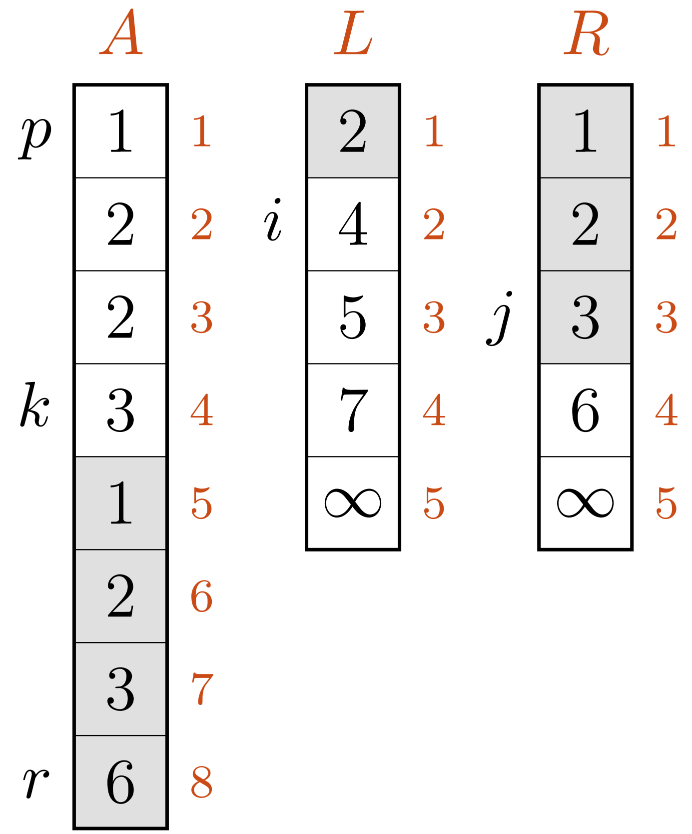


MERGE(A, p, q, r)

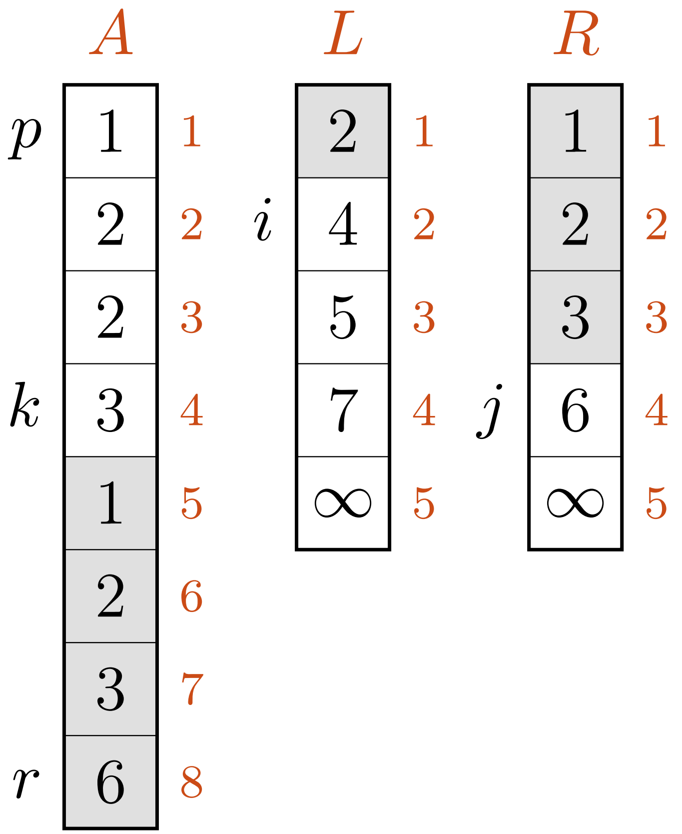
```
copy into L and R
for k = p to r
    if L[i] ≤ R[j]
        A[k] = L[i]
        i = i + 1
    else A[k] = R[j]
        j = j + 1
```



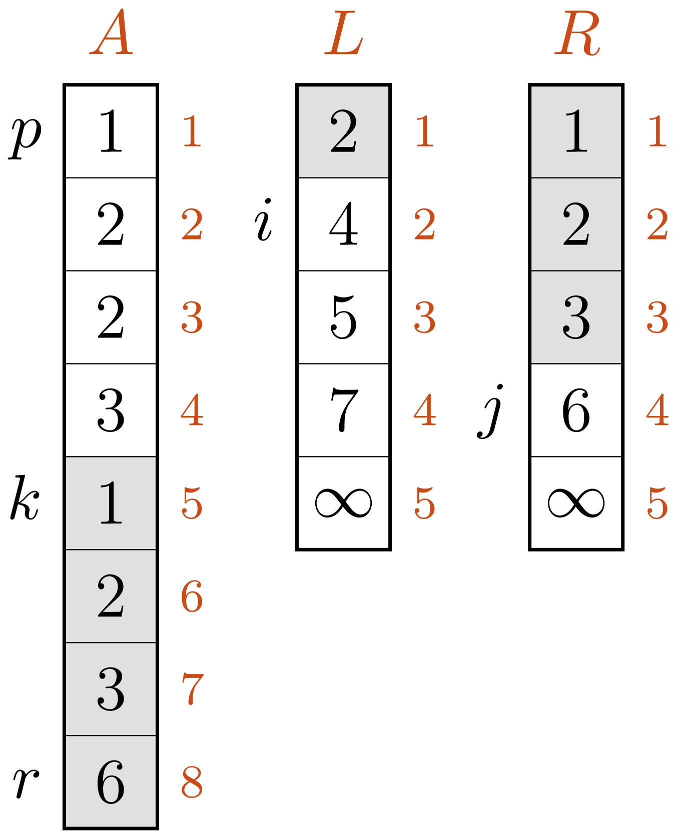
```
MERGE(A, p, q, r)
    copy into L and R
    for k = p to r
        if L[i] ≤ R[j]
            A[k] = L[i]
            i = i + 1
        else A[k] = R[j]
            j = j + 1
```



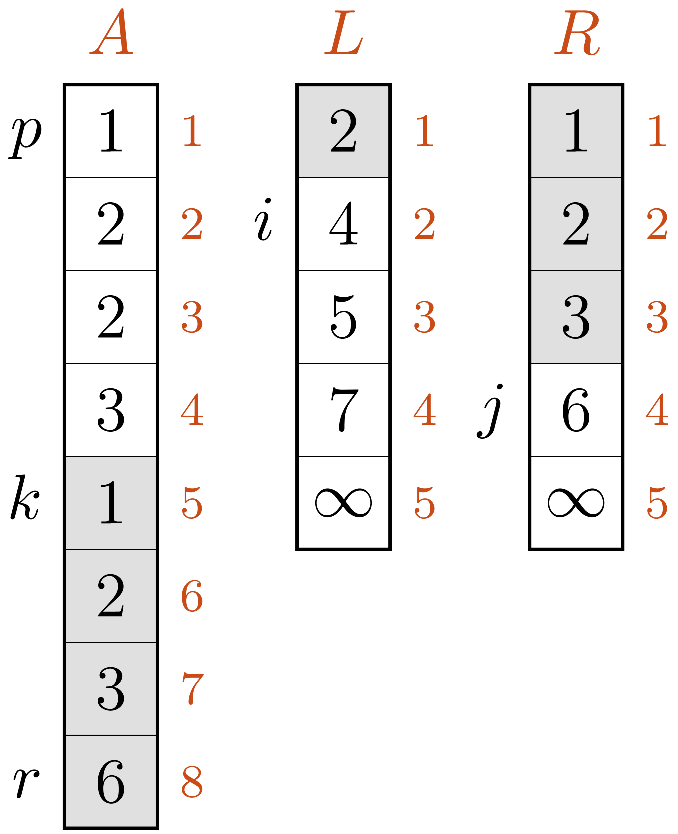
```
MERGE(A, p, q, r)
copy into L and R
for k = p to r
        if L[i] ≤ R[j]
              A[k] = L[i]
              i = i + 1
        else A[k] = R[j]
              j = j + 1
```



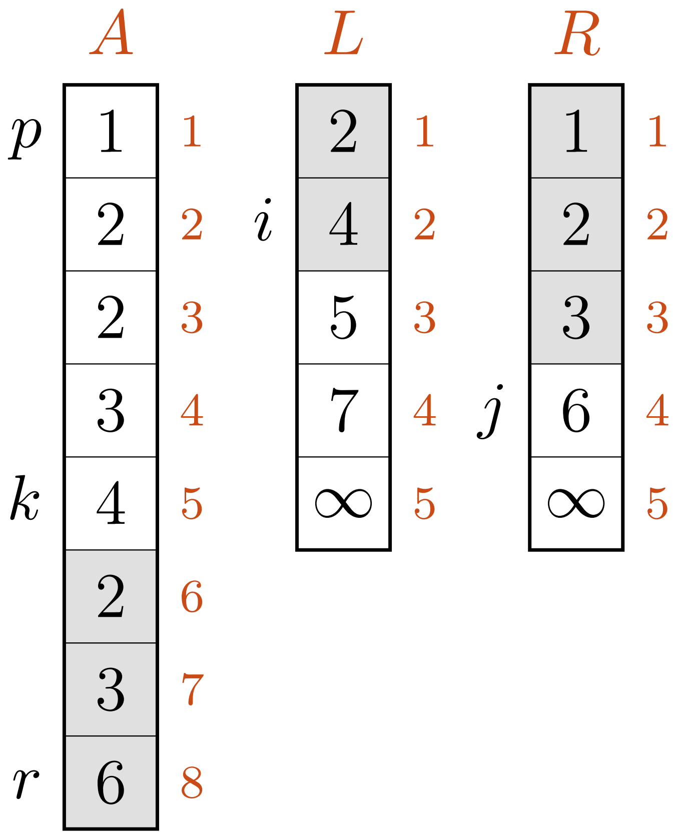
```
MERGE(A, p, q, r)
    copy into L and R
    for k = p to r
        if L[i] ≤ R[j]
            A[k] = L[i]
            i = i + 1
        else A[k] = R[j]
            j = j + 1
```



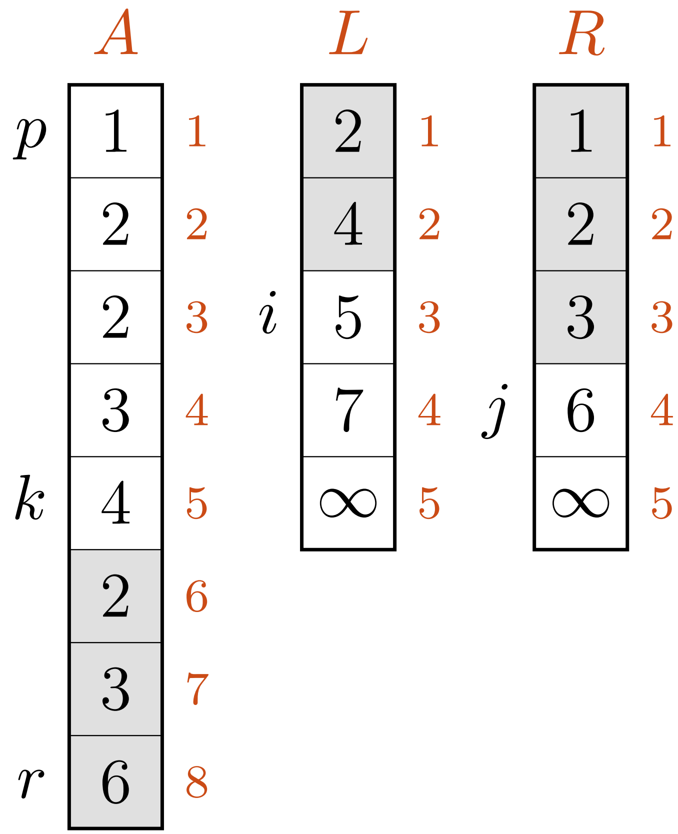
```
MERGE(A, p, q, r)
copy into L and R
for k = p to r
    if L[i] ≤ R[j]
        A[k] = L[i]
        i = i + 1
    else A[k] = R[j]
        j = j + 1
```



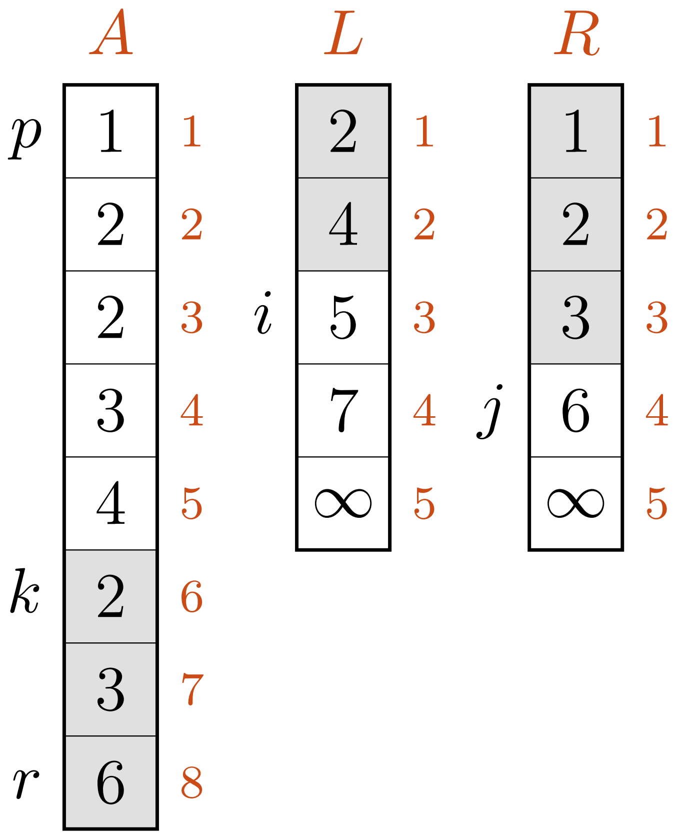
```
MERGE(A, p, q, r)
    copy into L and R
    for k = p to r
        if L[i] ≤ R[j]
            A[k] = L[i]
            i = i + 1
        else A[k] = R[j]
            j = j + 1
```



```
MERGE(A, p, q, r)
copy into L and R
for k = p to r
        if L[i] ≤ R[j]
                A[k] = L[i]
                i = i + 1
        else A[k] = R[j]
                j = j + 1
```

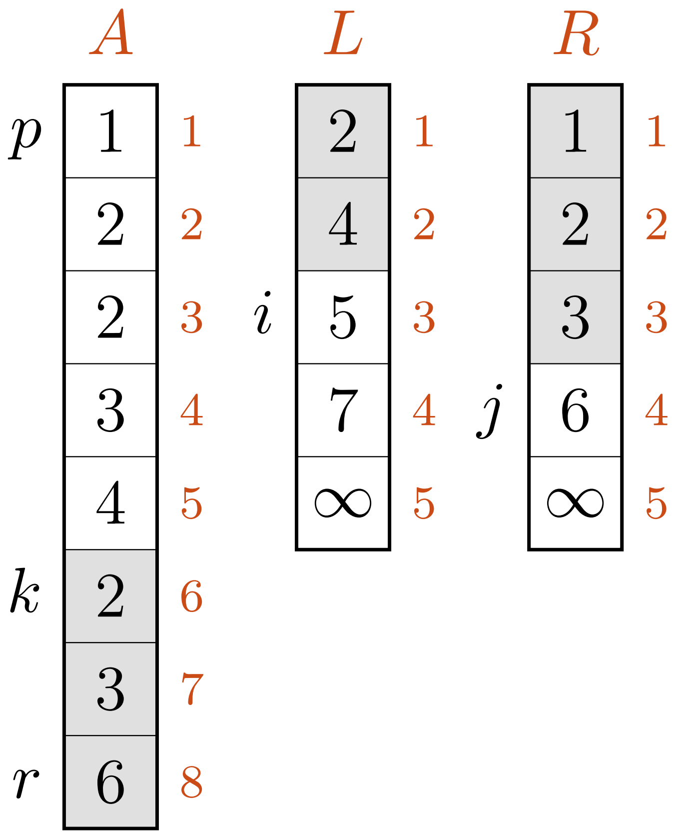


```
Merge(A, p, q, r)
copy into L and R
for k = p to r
        if L[i] ≤ R[j]
                A[k] = L[i]
                i = i + 1
        else A[k] = R[j]
                j = j + 1
```

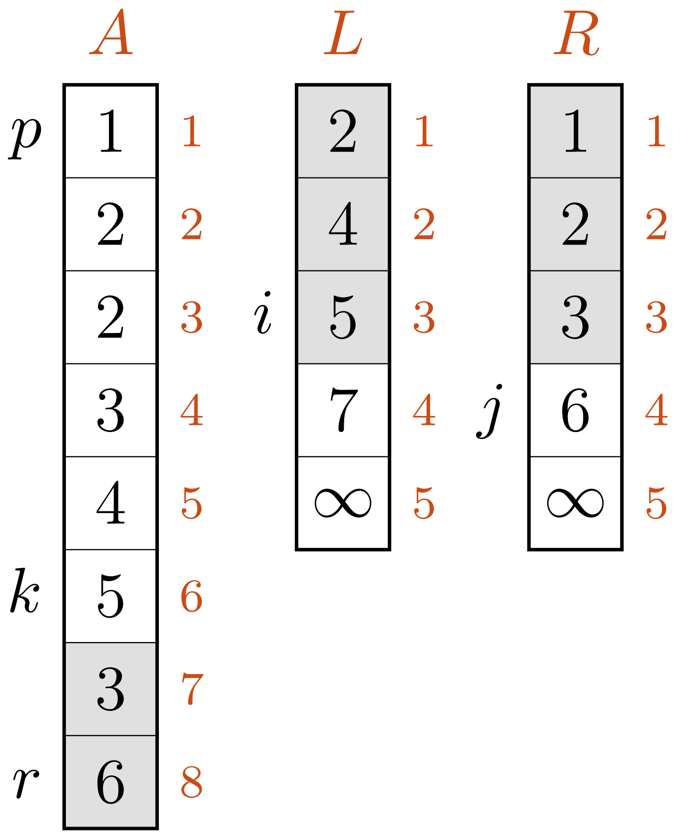


MERGE(A, p, q, r)

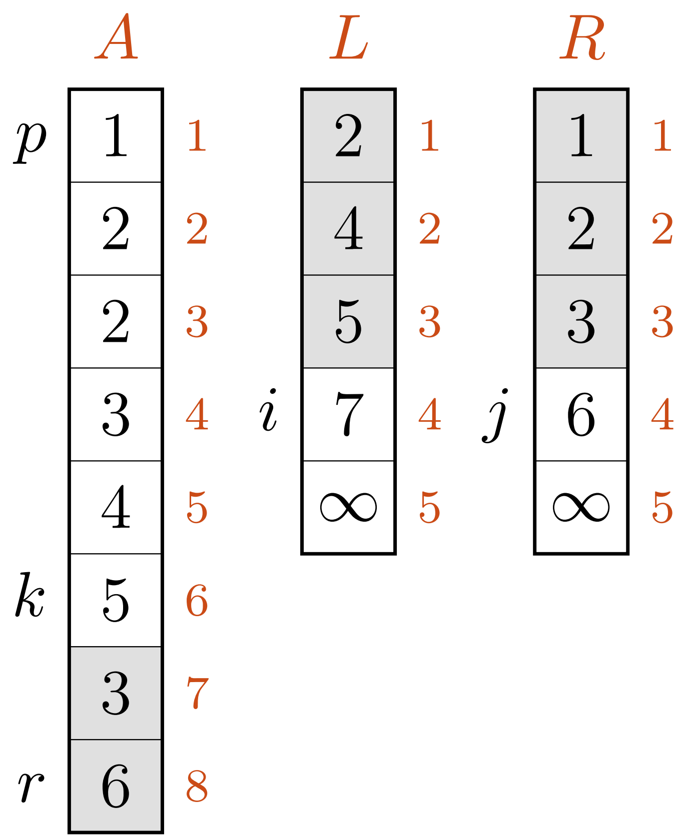
```
copy into L and R
for k = p to r
        if L[i] ≤ R[j]
                A[k] = L[i]
                i = i + 1
        else A[k] = R[j]
                j = j + 1
```



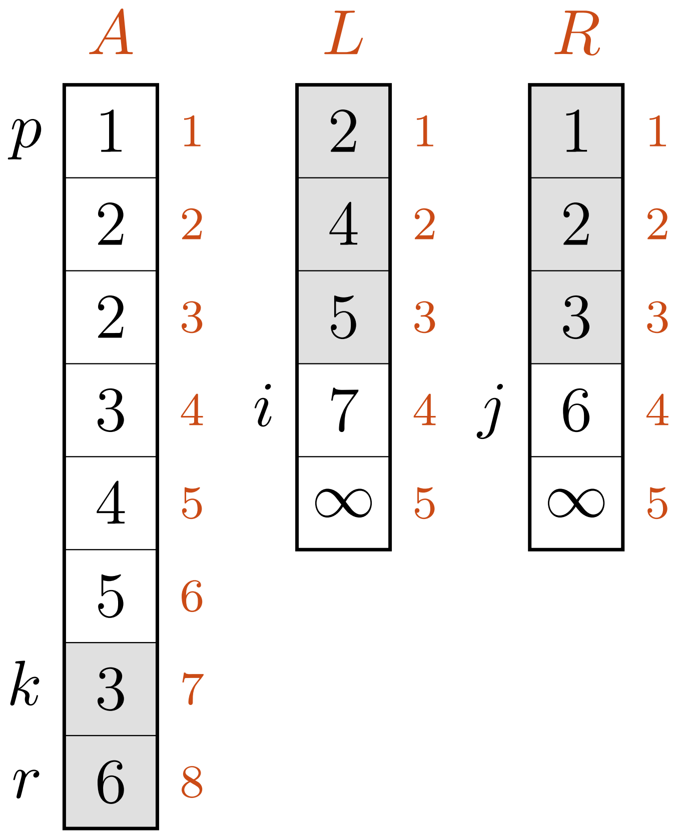
```
MERGE(A, p, q, r)
    copy into L and R
    for k = p to r
        if L[i] ≤ R[j]
            A[k] = L[i]
            i = i + 1
        else A[k] = R[j]
            j = j + 1
```



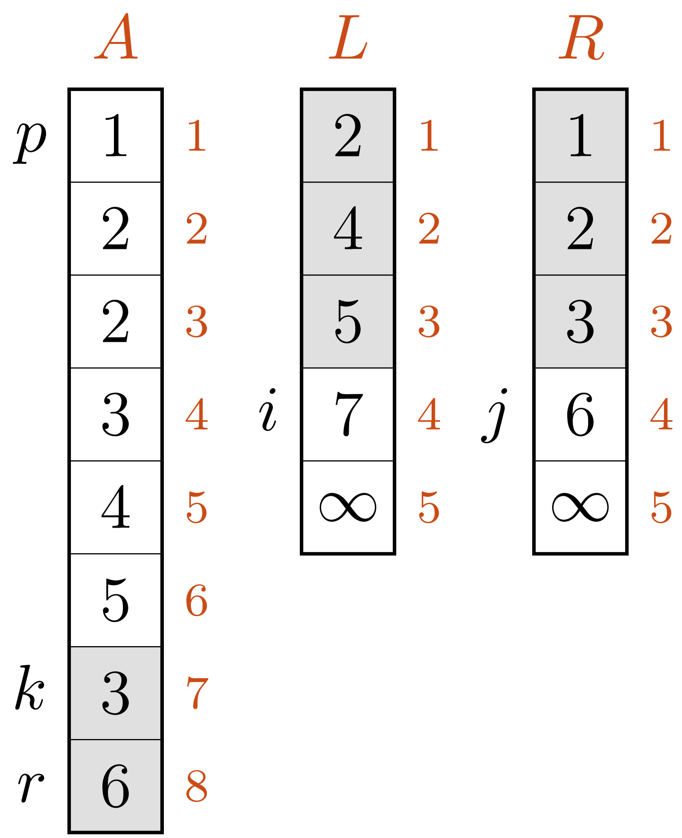
```
MERGE(A, p, q, r)
copy into L and R
for k = p to r
      if L[i] ≤ R[j]
            A[k] = L[i]
            i = i + 1
      else A[k] = R[j]
            j = j + 1
```



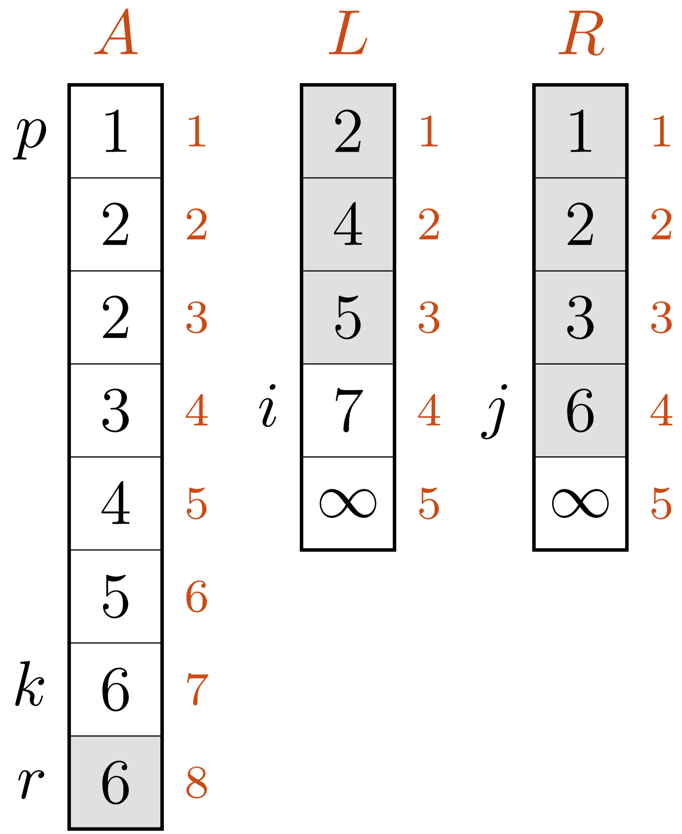
```
MERGE(A, p, q, r)
    copy into L and R
    for k = p to r
        if L[i] ≤ R[j]
            A[k] = L[i]
            i = i + 1
        else A[k] = R[j]
            j = j + 1
```



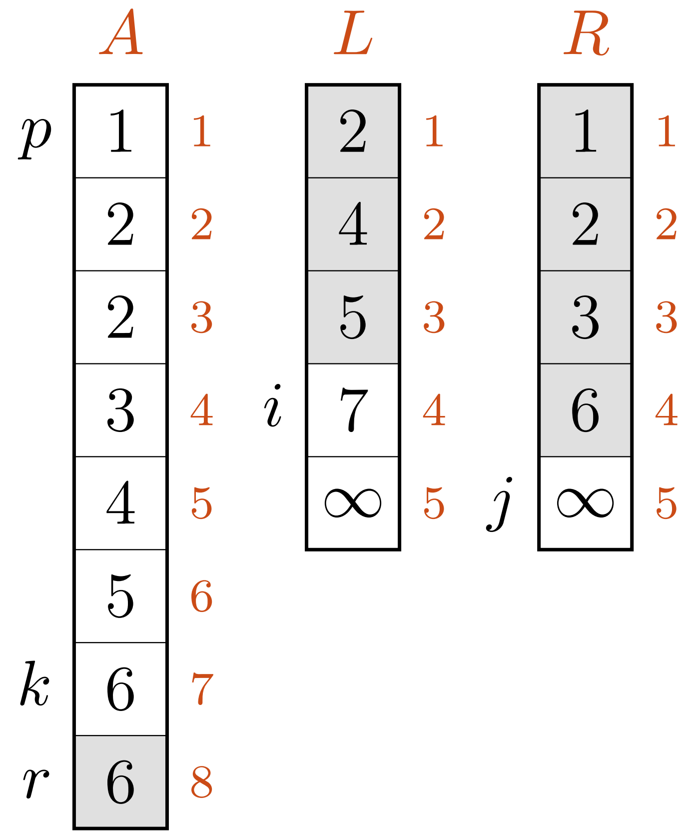
```
MERGE(A, p, q, r)
  copy into L and R
  for k = p to r
      if L[i] ≤ R[j]
          A[k] = L[i]
          i = i + 1
      else A[k] = R[j]
          j = j + 1
```



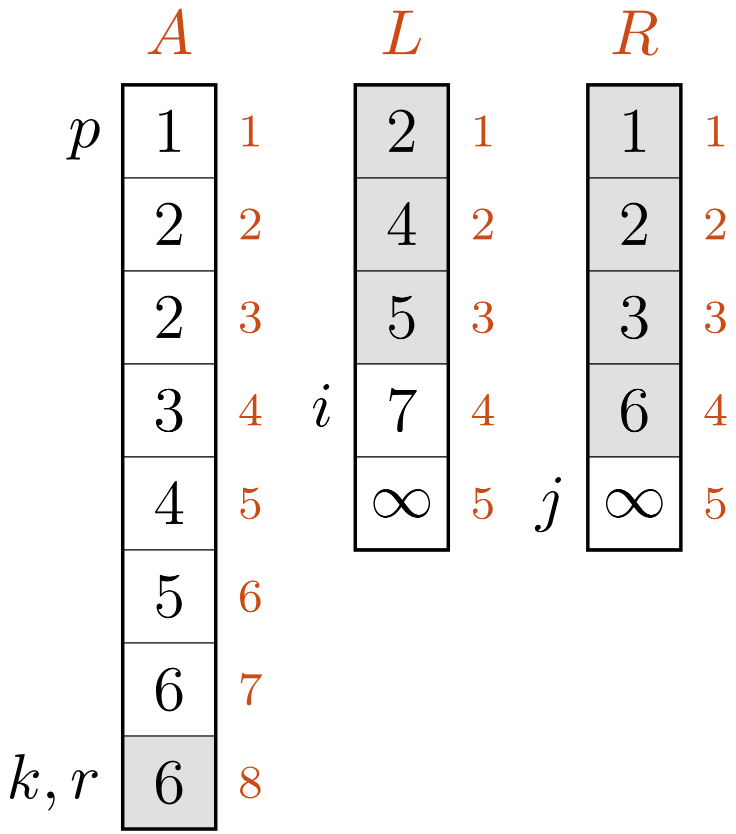
```
MERGE(A, p, q, r)
    copy into L and R
    for k = p to r
        if L[i] ≤ R[j]
            A[k] = L[i]
            i = i + 1
        else A[k] = R[j]
            j = j + 1
```



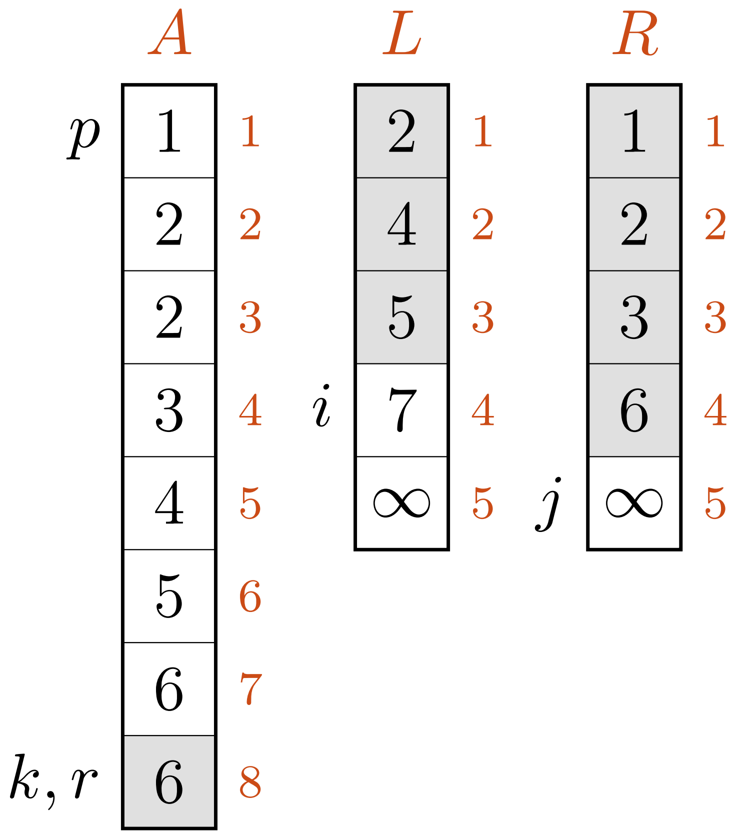
```
Merge(A, p, q, r)
    copy into L and R
    for k = p to r
            if L[i] ≤ R[j]
                    A[k] = L[i]
                    i = i + 1
            else A[k] = R[j]
                    j = j + 1
```



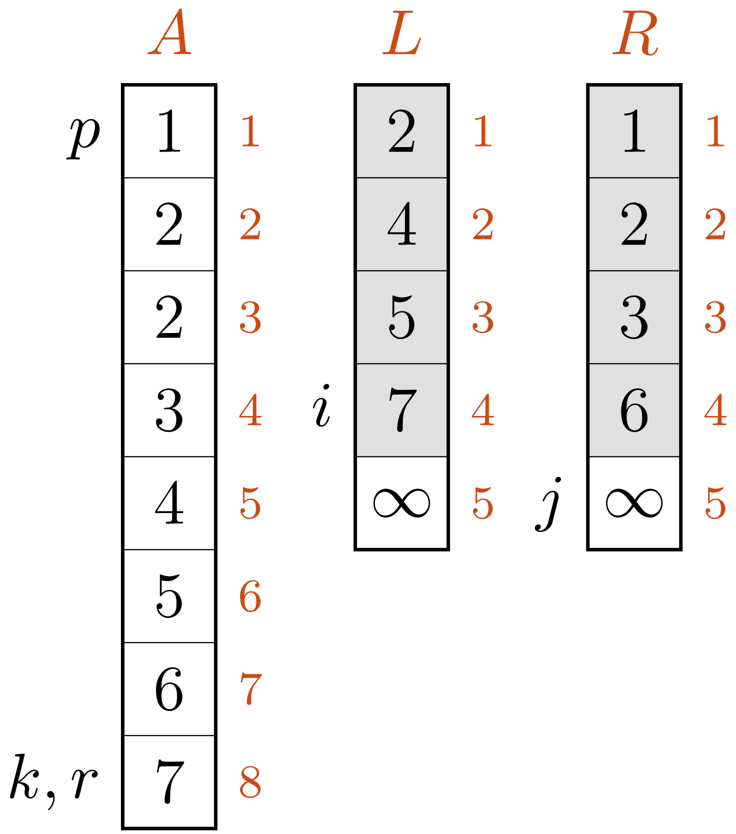
```
MERGE(A, p, q, r)
  copy into L and R
  for k = p to r
      if L[i] ≤ R[j]
          A[k] = L[i]
          i = i + 1
      else A[k] = R[j]
          j = j + 1
```



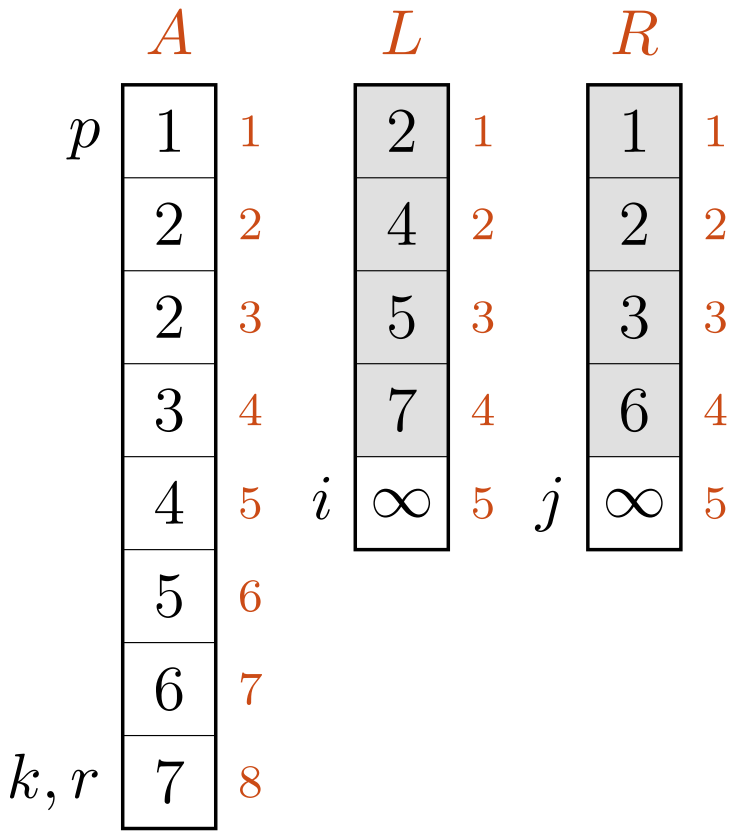
```
MERGE(A, p, q, r)
    copy into L and R
    for k = p to r
        if L[i] ≤ R[j]
            A[k] = L[i]
            i = i + 1
        else A[k] = R[j]
            j = j + 1
```



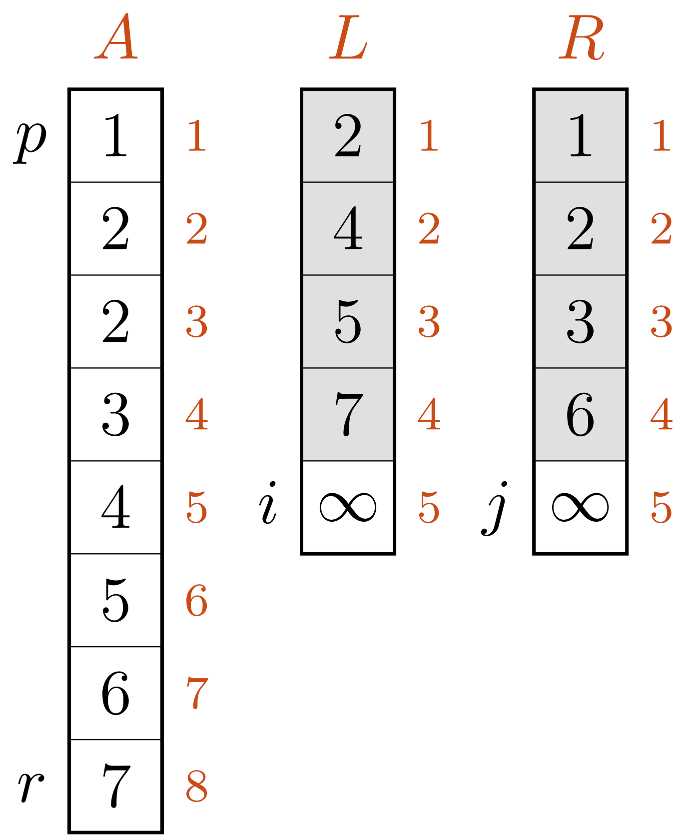
```
MERGE(A, p, q, r)
  copy into L and R
  for k = p to r
      if L[i] ≤ R[j]
          A[k] = L[i]
          i = i + 1
      else A[k] = R[j]
          j = j + 1
```



```
Merge(A, p, q, r)
  copy into L and R
  for k = p to r
      if L[i] ≤ R[j]
          A[k] = L[i]
          i = i + 1
      else A[k] = R[j]
          j = j + 1
```



```
MERGE(A, p, q, r)
copy into L and R
for k = p to r
        if L[i] ≤ R[j]
                A[k] = L[i]
                i = i + 1
        else A[k] = R[j]
                j = j + 1
```



Max-subaray:
Neste callback corner

Pensum

- Kap. 4. Divide-and-conquer: Innledning, 4.1 og 4.3–4.5
- Kap. 7. Quicksort
- Oppgaver 2.3-5 og 4.5-3 med løsning (binærsøk)

Potenser og logaritmer

$$2 \times 2 \times \cdots \times 2$$

$$\underbrace{2 \times 2 \times \cdots \times 2}_{m}$$

$$\mathrm{length}(\underbrace{2 \times 2 \times \cdots \times 2}) = m$$

$$\mathrm{length}\underbrace{(2 \times 2 \times \cdots \times 2)} = m$$

$$\lg\underbrace{(2 \times 2 \times \cdots \times 2)} = m$$

$$\log_2\underbrace{(2 \times 2 \times \cdots \times 2)} = m$$

$$\log_3(3 \times 3 \times \cdots \times 3) = m$$

$$\log_k\underbrace{(k \times k \times \cdots \times k)} = m$$

$$\frac{n}{\underbrace{2 \times 2 \times \cdots \times 2}_{i}} = \frac{n}{2^i} = 1 \iff i = \lg n$$

$$\left(n = 2^{\lg n}\right)$$

$$
\begin{array}{rr}
 & 1 \\
+ & 2 \\
+ & 2 \times 2 \\
+ & 2 \times 2 \times 2 \\
+ & 2 \times 2 \times 2 \times 2 \\
\hline
= & 2 \times 2 \times 2 \times 2 \times 2 - 1 \\
\hline\hline
\end{array}
$$

$$
\begin{array}{rr}
 & 2 \\
+ & 2 \times 2 \\
+ & 2 \times 2 \times 2 \\
+ & 2 \times 2 \times 2 \times 2 \\
\hline
= & 2 \times 2 \times 2 \times 2 \times 2 - 2 \\
\hline\hline
\end{array}
$$

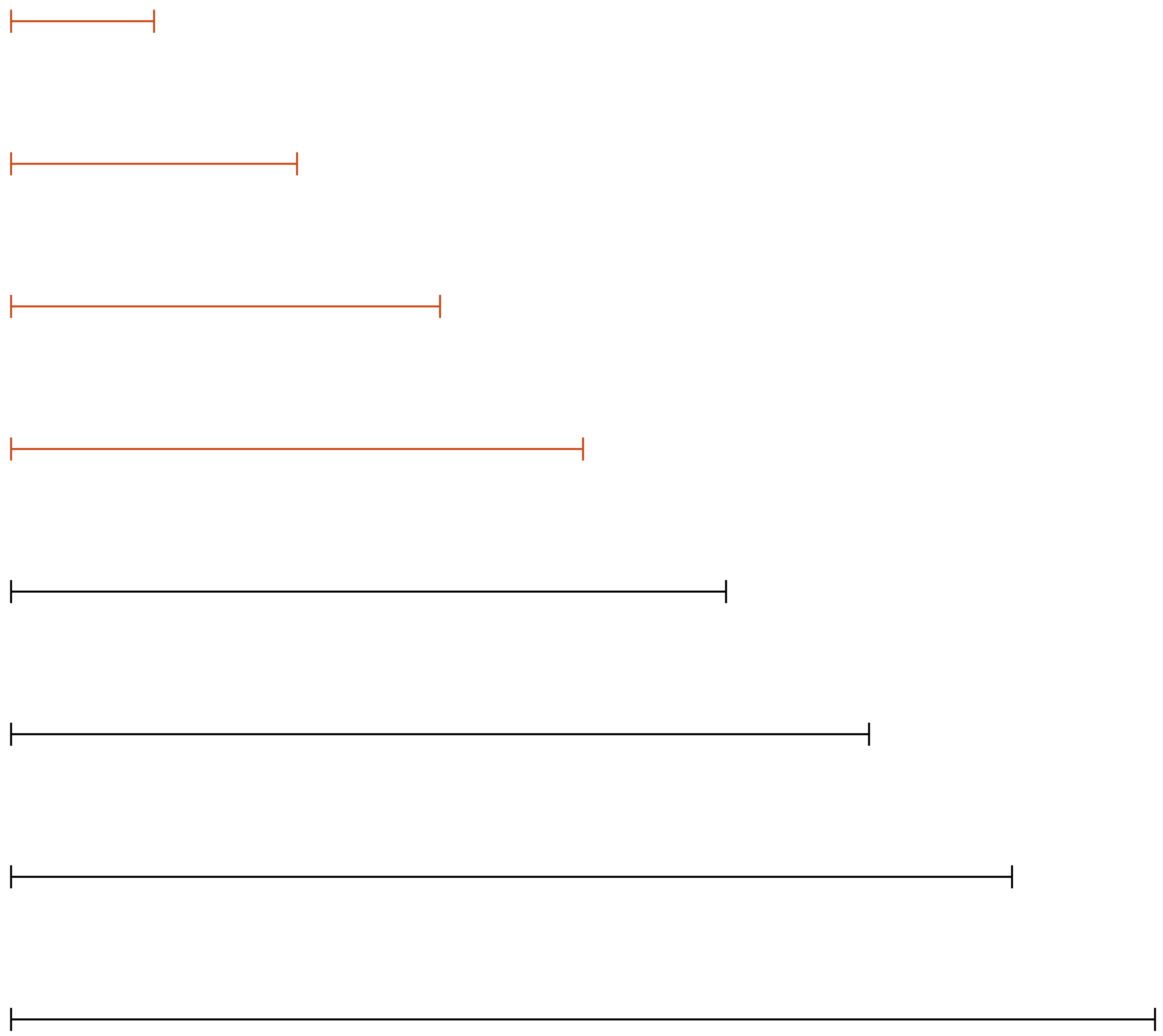
$$
\begin{array}{lr}
 & 2 \\
+ & 2 \times 2 \\
 & \vdots \\
+ & 2 \times \cdots \times 2 \\
+ & 2 \times 2 \times \cdots \times 2 \\
\hline
= & \\
\hline\hline
\end{array}
$$

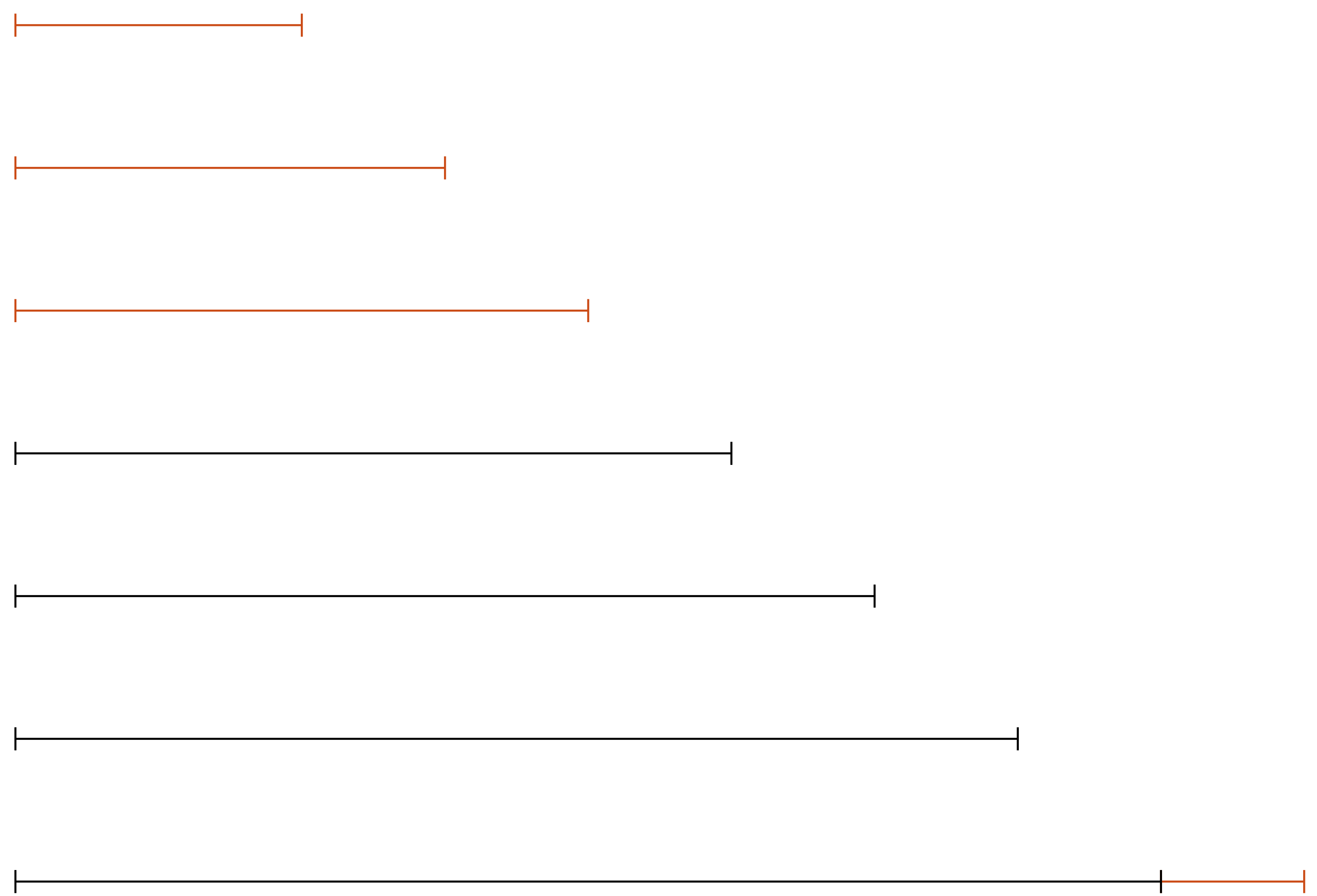
$$
\begin{array}{lr}
 & 2 \\
+ & 2 \times 2 \\
 & \vdots \\
+ & 2 \times \cdots \times 2 \\
= & 2 \times 2 \times \cdots \times 2 - 2 \\
+ & 2 \times 2 \times \cdots \times 2 \\
\hline
= & \\
\hline\hline
\end{array}
$$

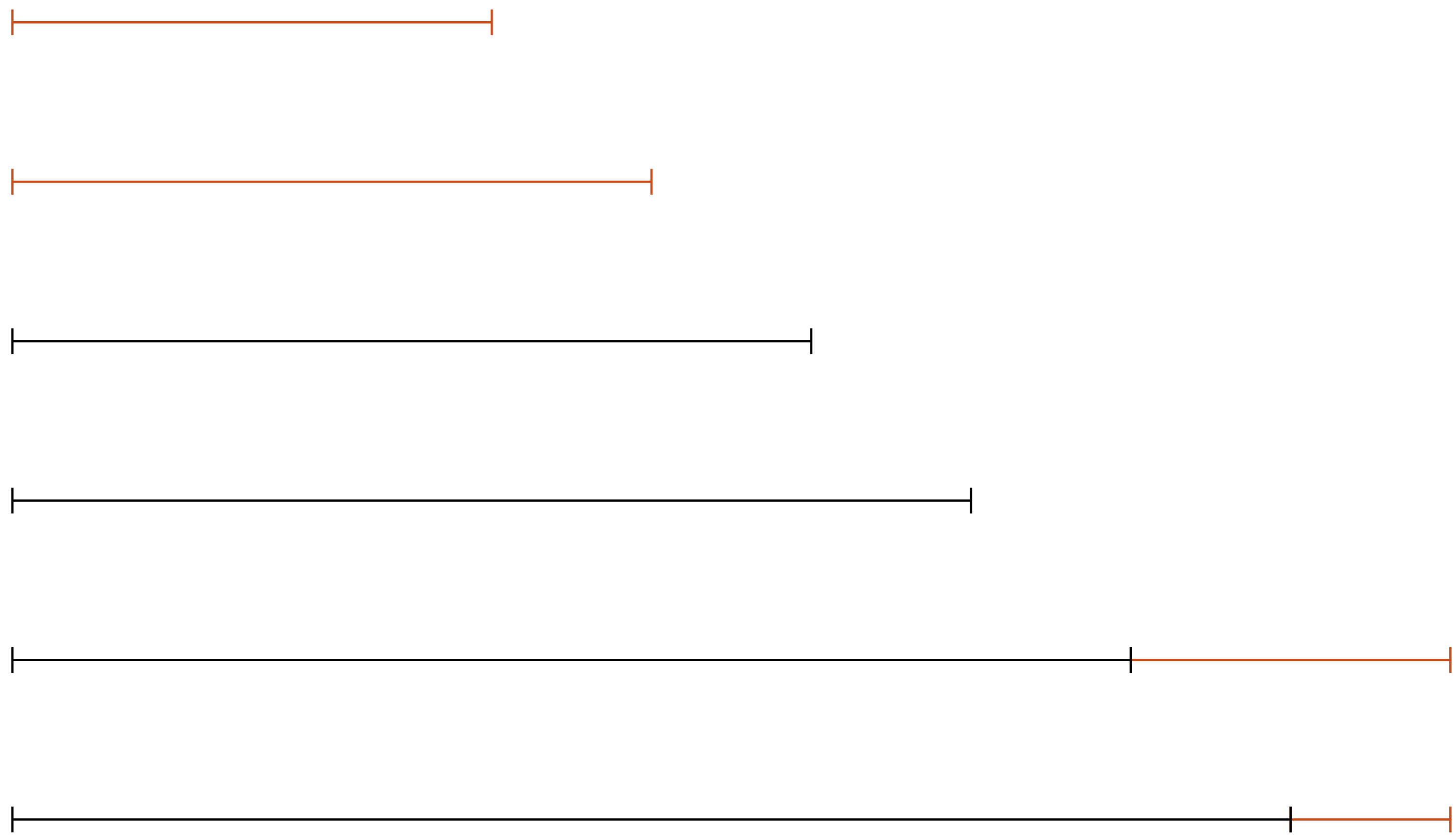
$$
\begin{array}{cr}
 & 2 \\
+ & 2 \times 2 \\
 & \vdots \\
+ & 2 \times \cdots \times 2 \\
= & 2 \times 2 \times \cdots \times 2 - 2 \\
+ & 2 \times 2 \times \cdots \times 2 \\
\hline
= & 2 \times 2 \times 2 \times \cdots \times 2 - 2 \\
\hline\hline
\end{array}
$$

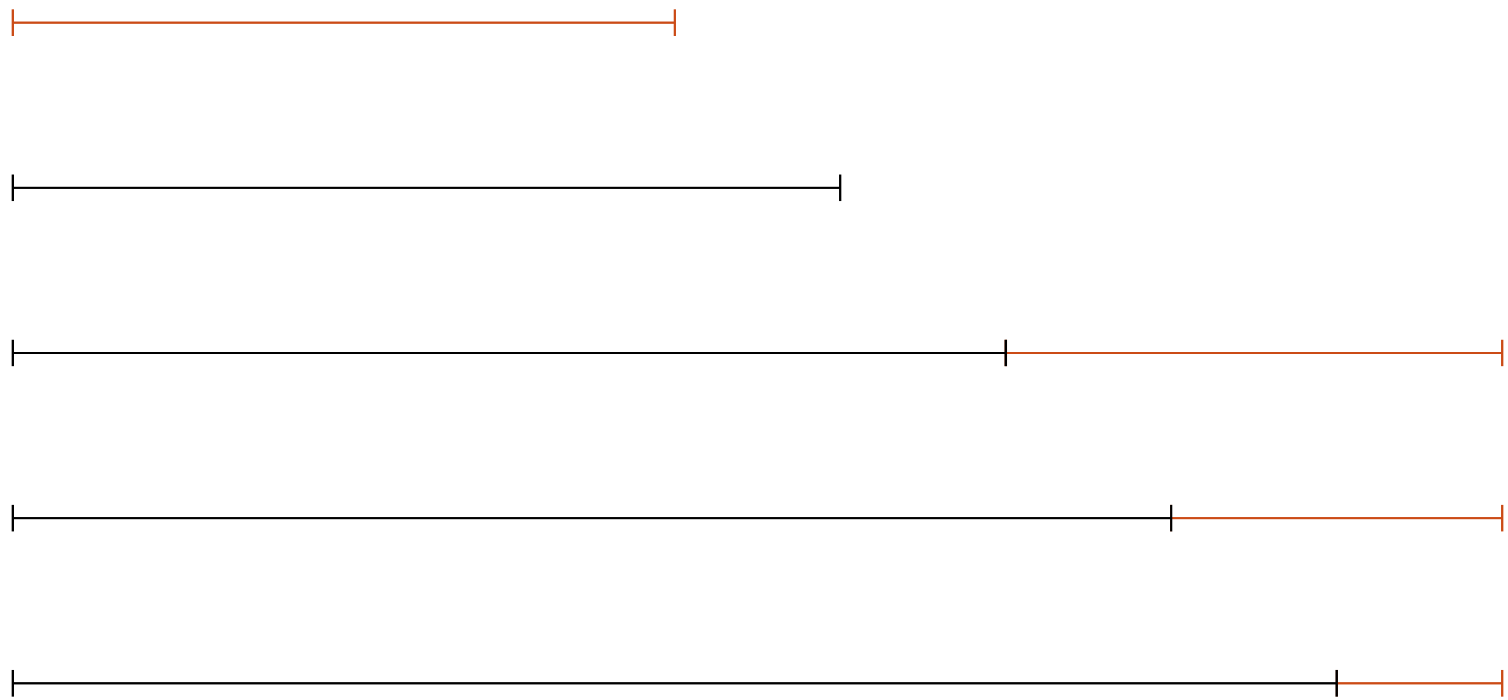
$$\begin{array}{lr}
 & 2 \\
+ & 2\times 2 \\
 & \vdots \\
+ & 2\times\cdots\times 2 \\
+ & 2\times 2\times\cdots\times 2 \\
\hline
= & 2\times 2\times 2\times\cdots\times 2 - 2
\end{array}$$

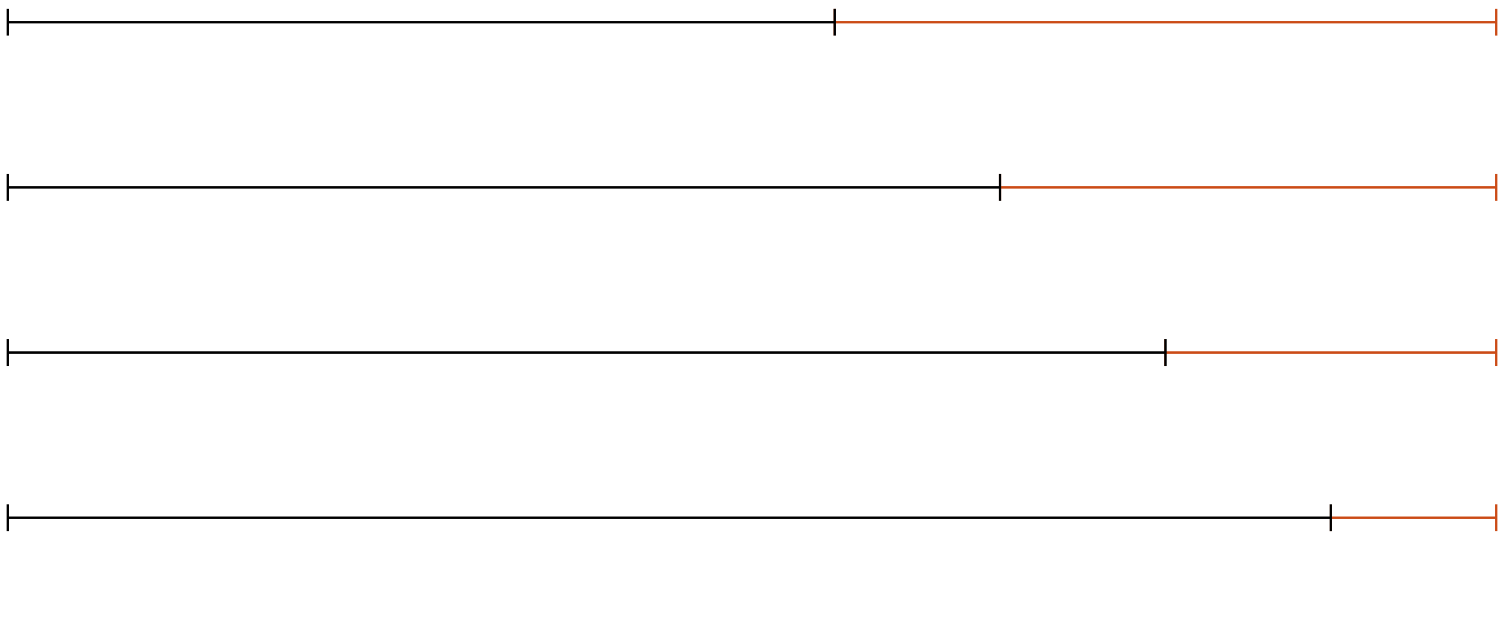
Litt grunnlag

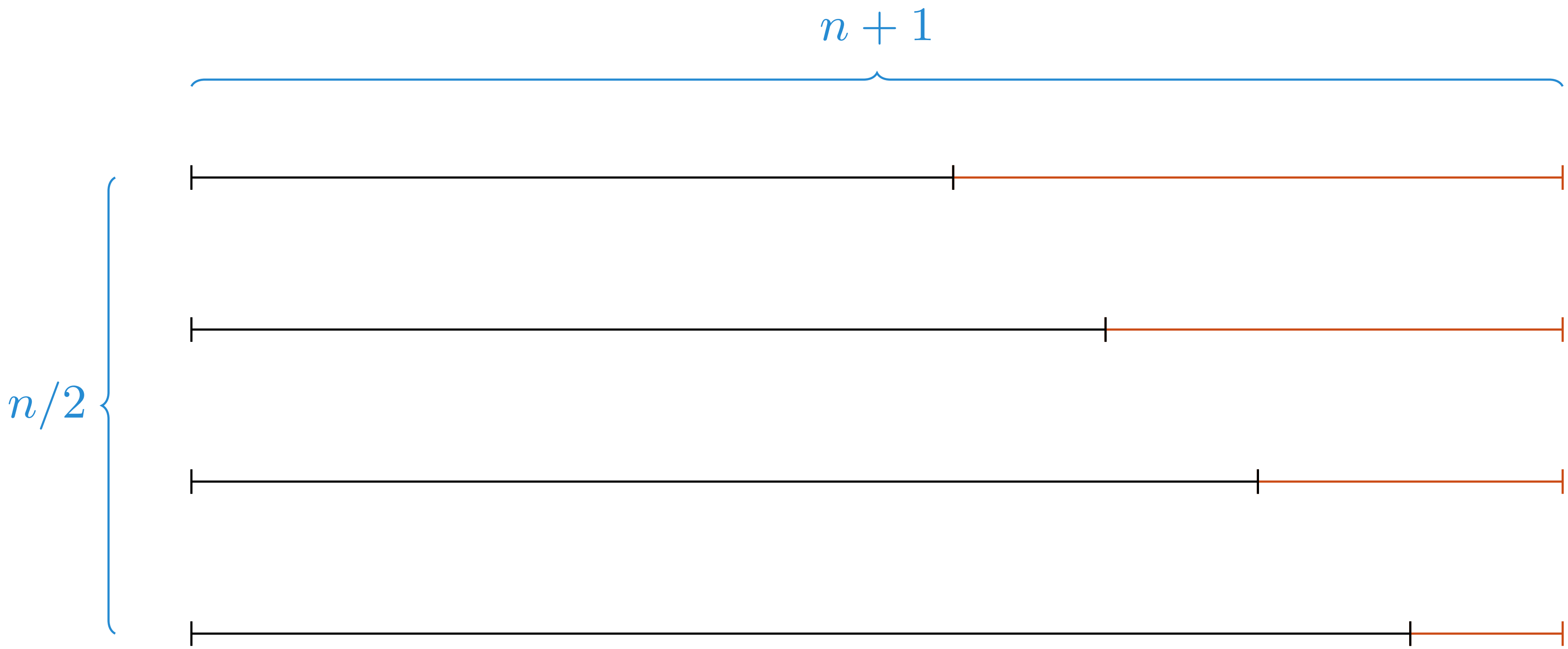




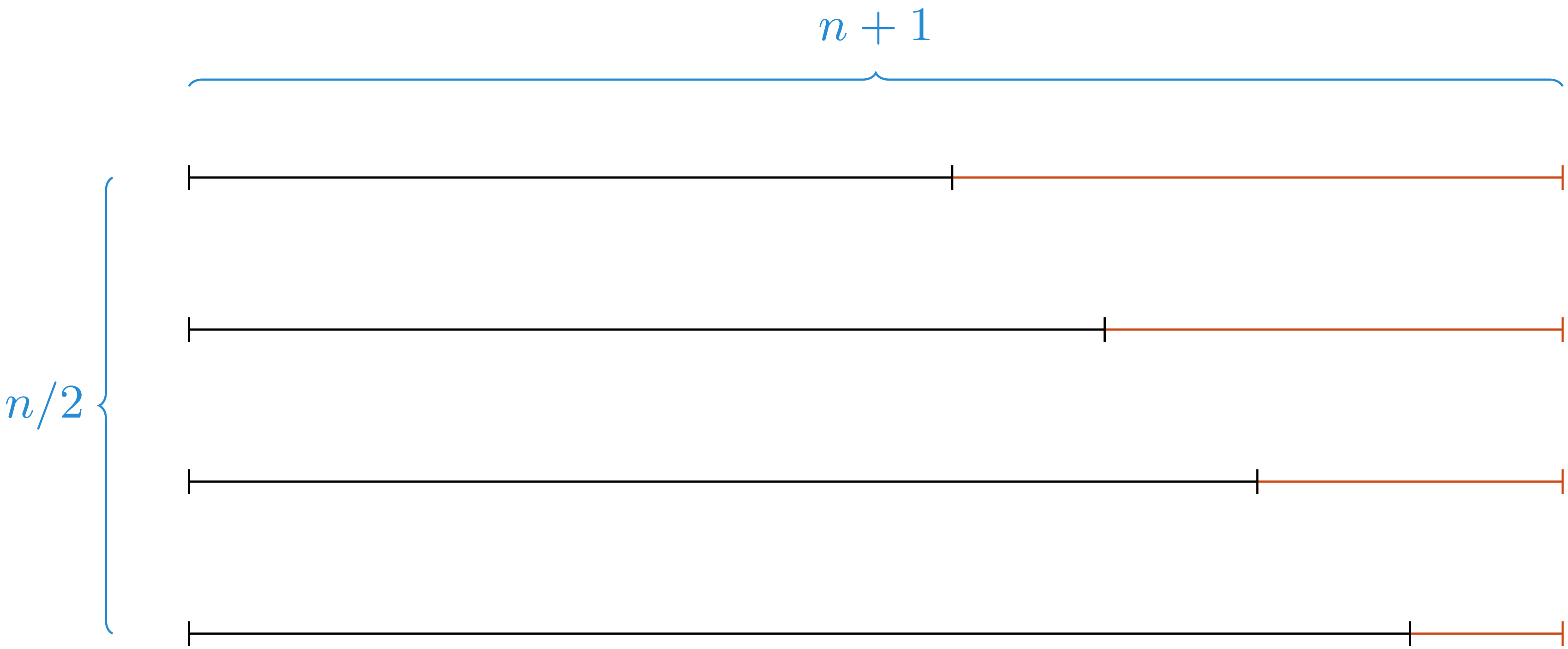


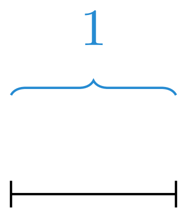


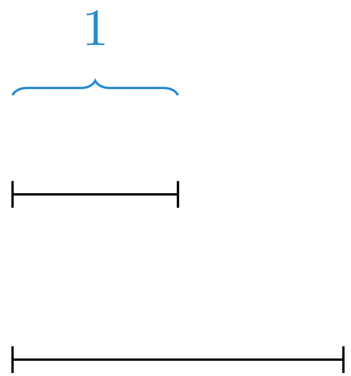


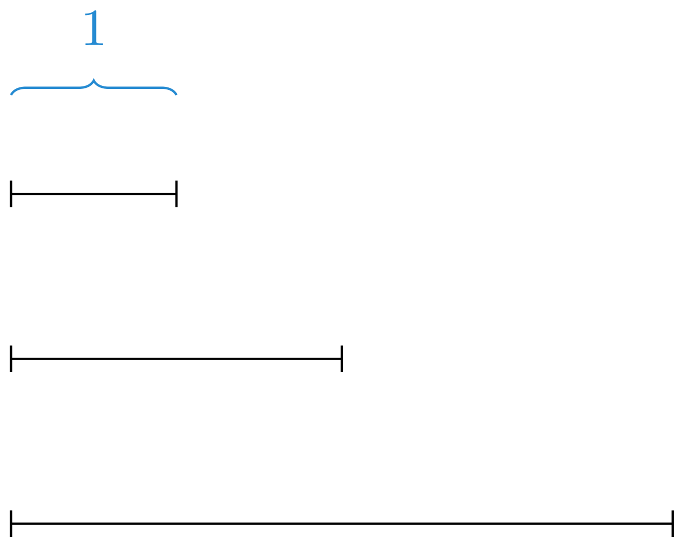

$n + 1$
$n/2$

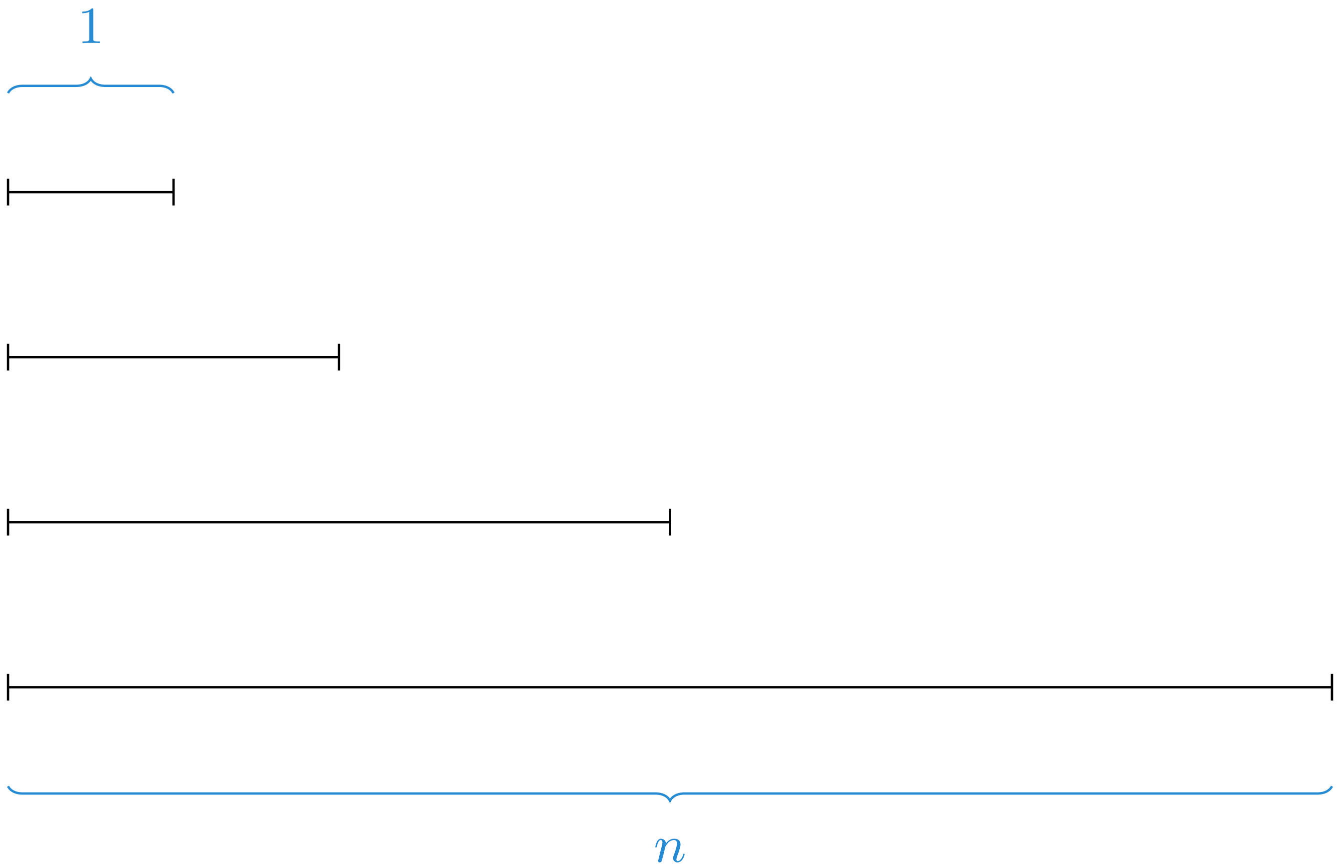
$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$$

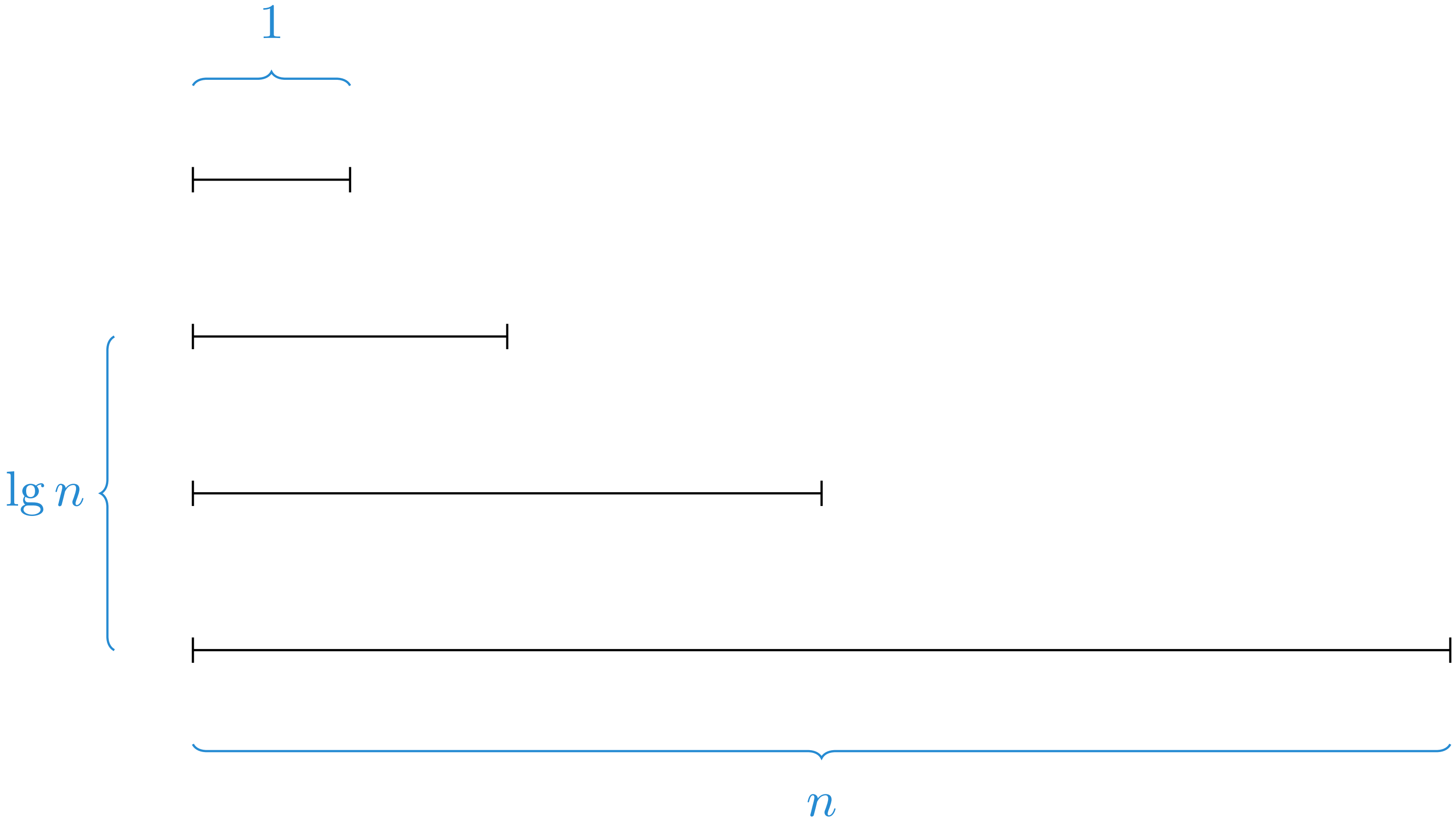


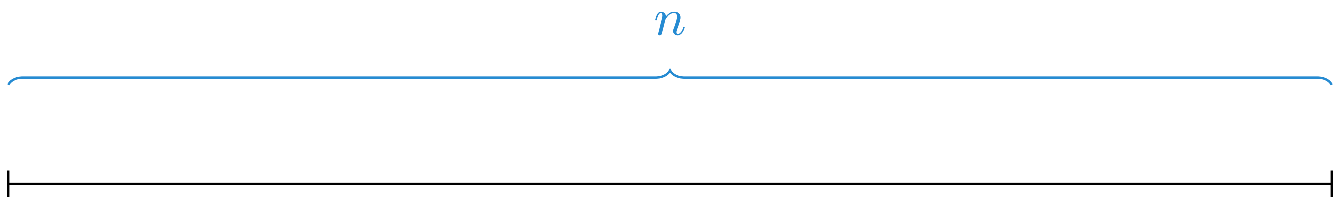

1

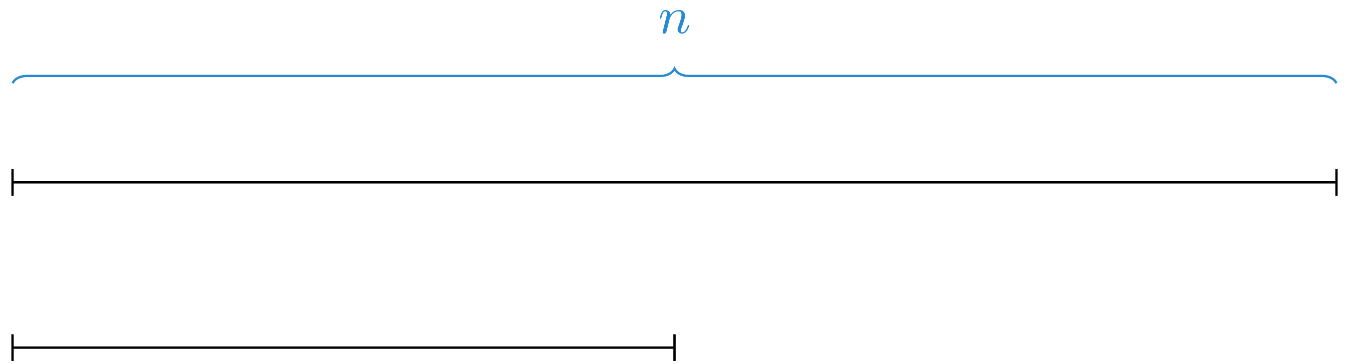

1

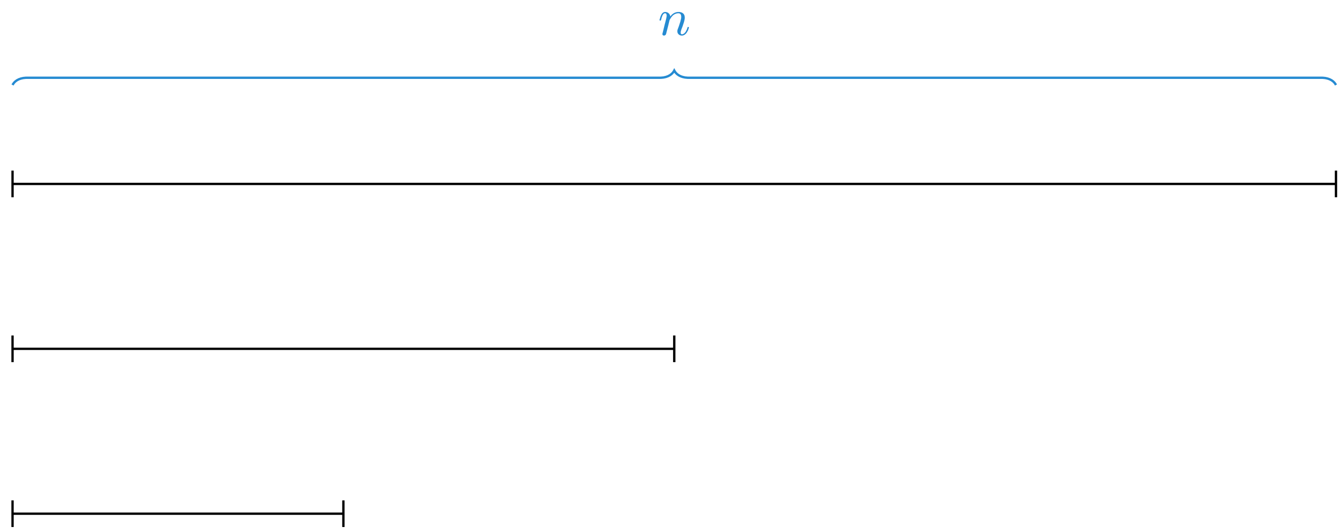

1

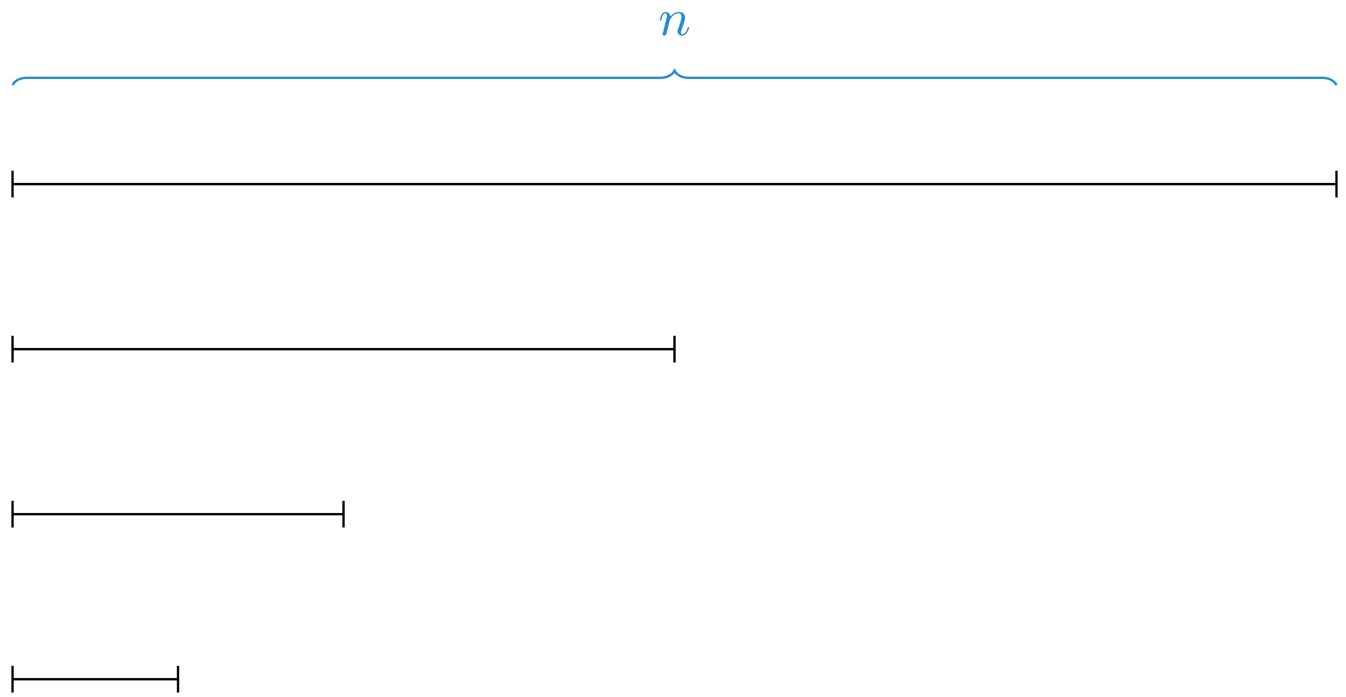

1
n

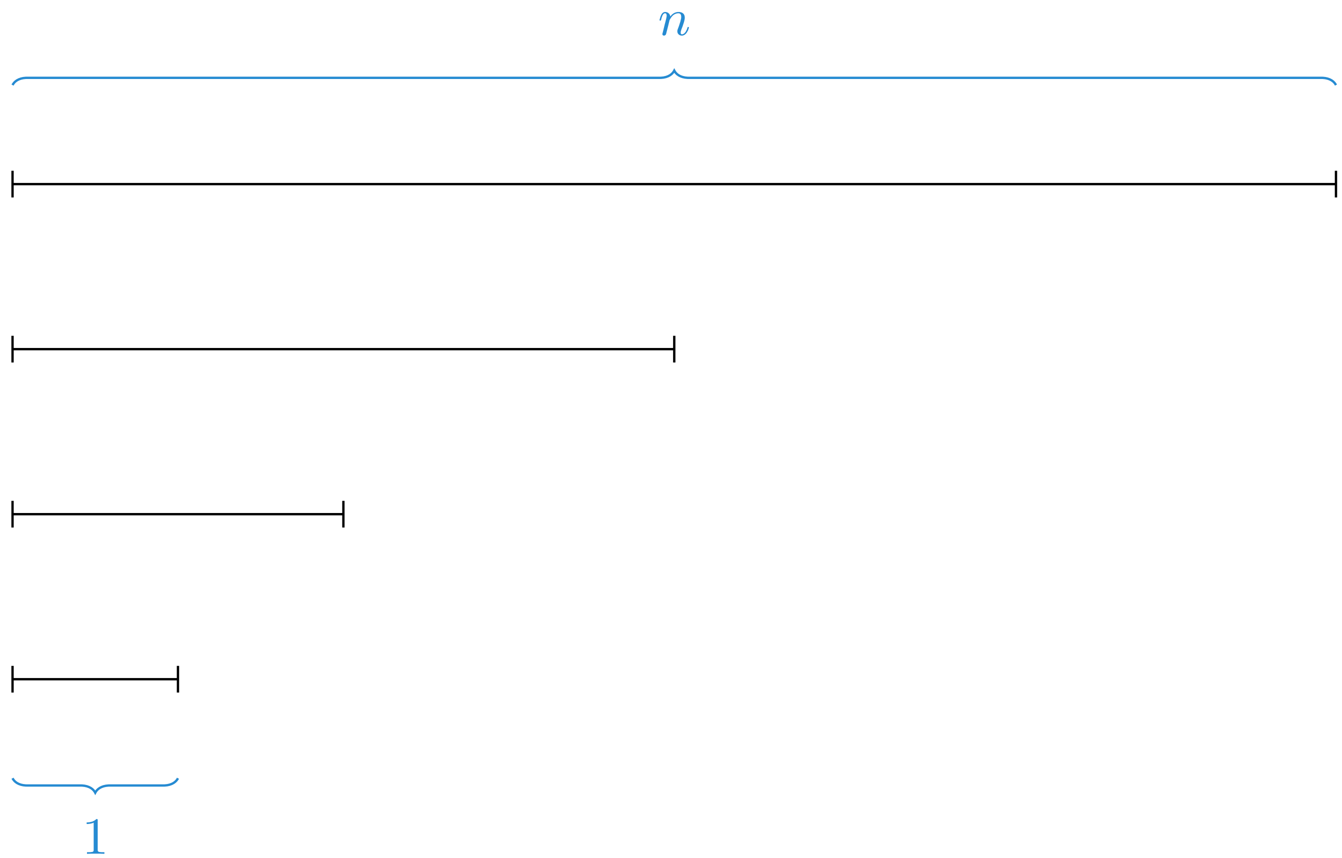

1
lg n
n

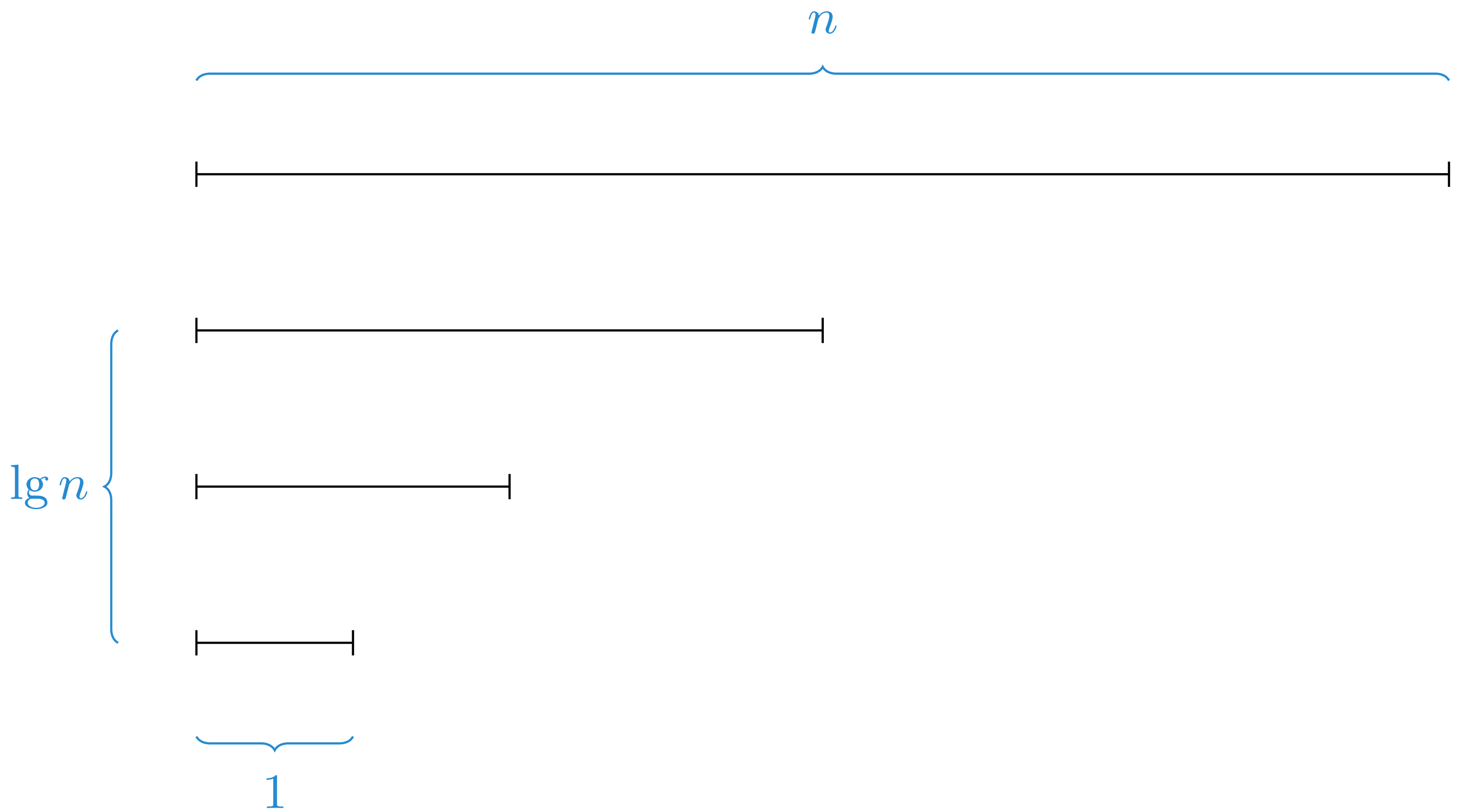

n

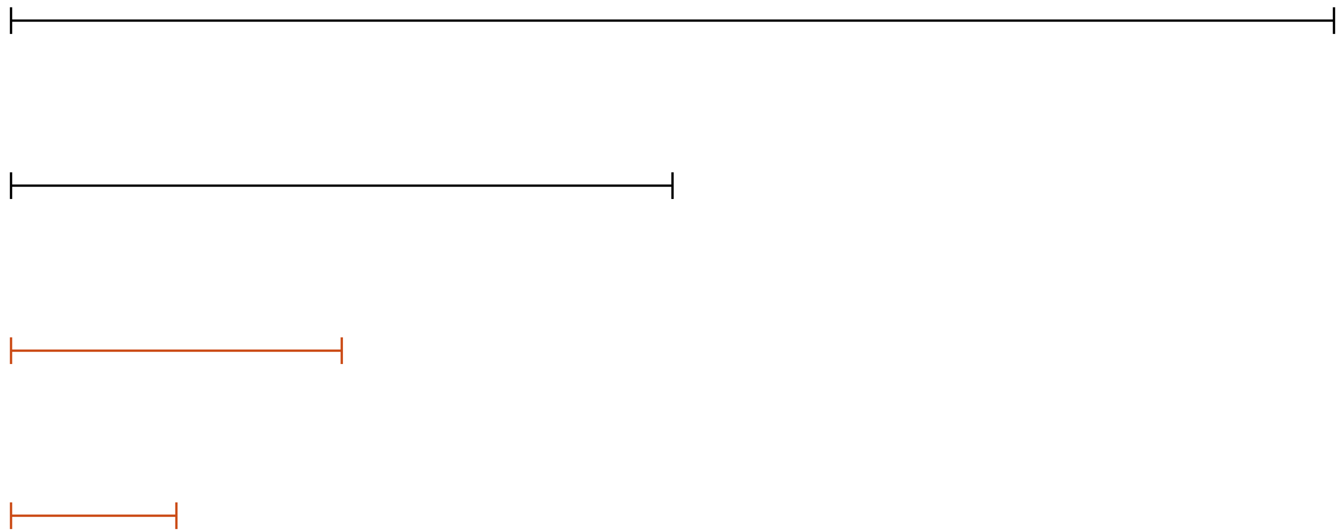

n

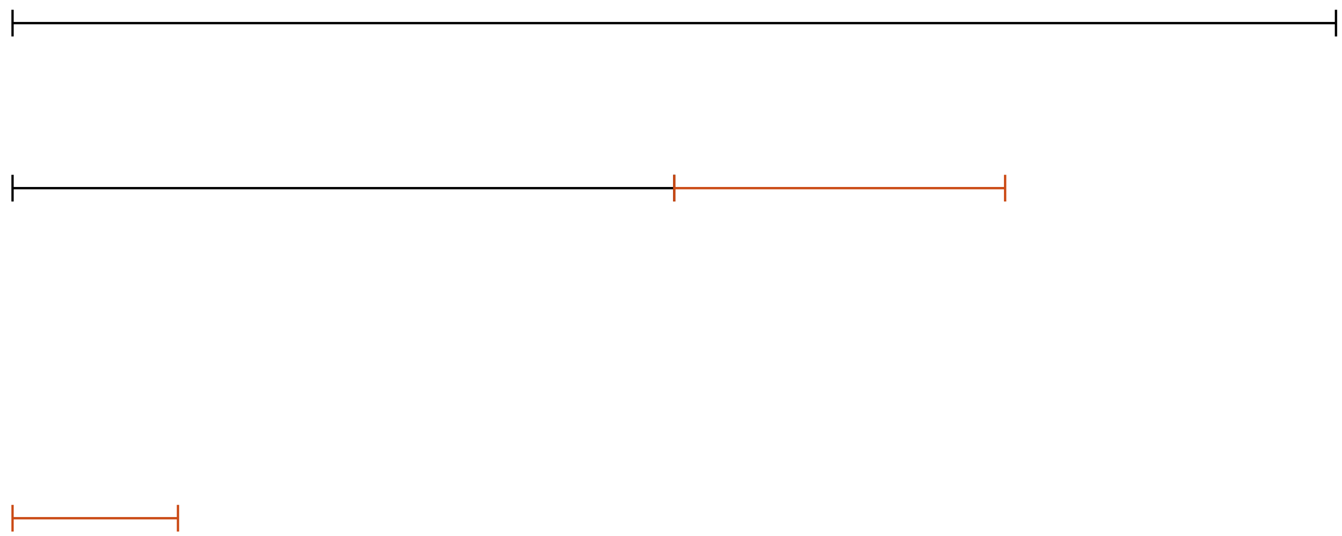

n

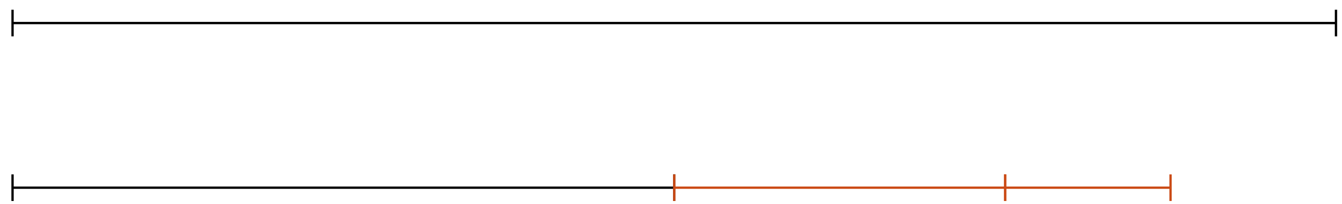

n

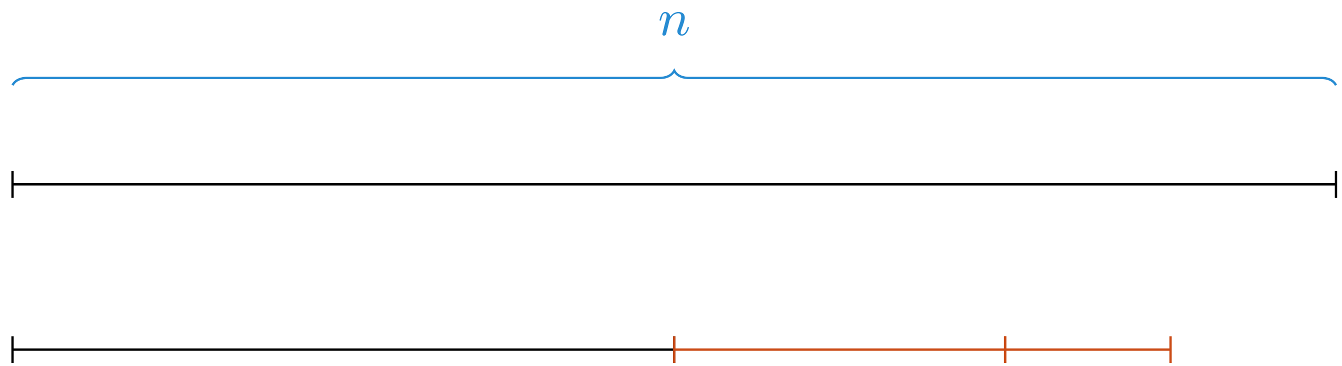

n
1


n
$\lg n$
1

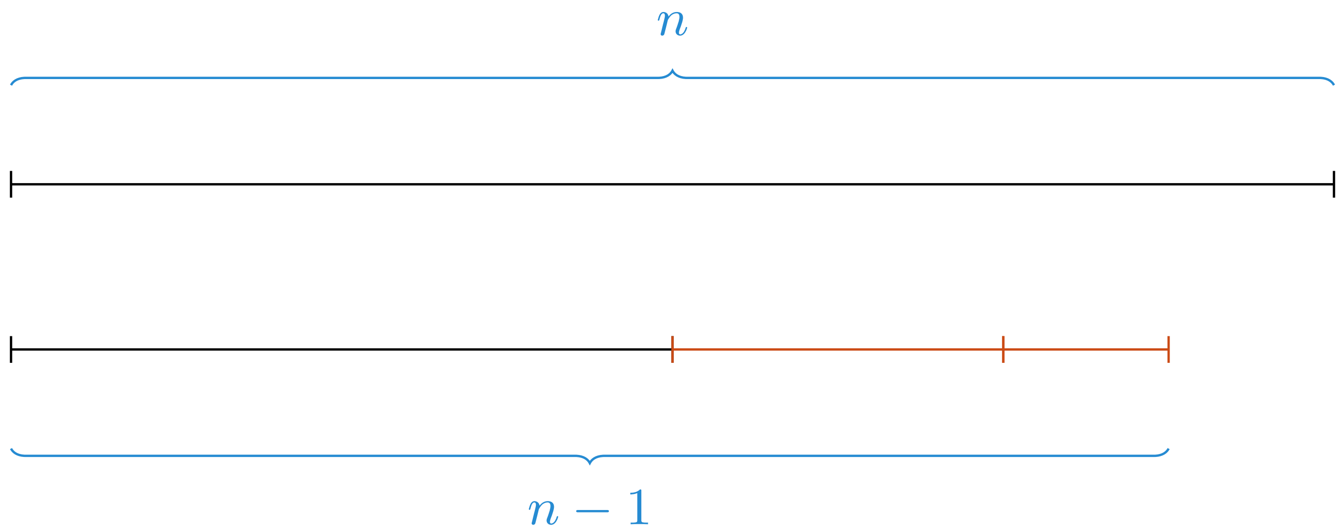


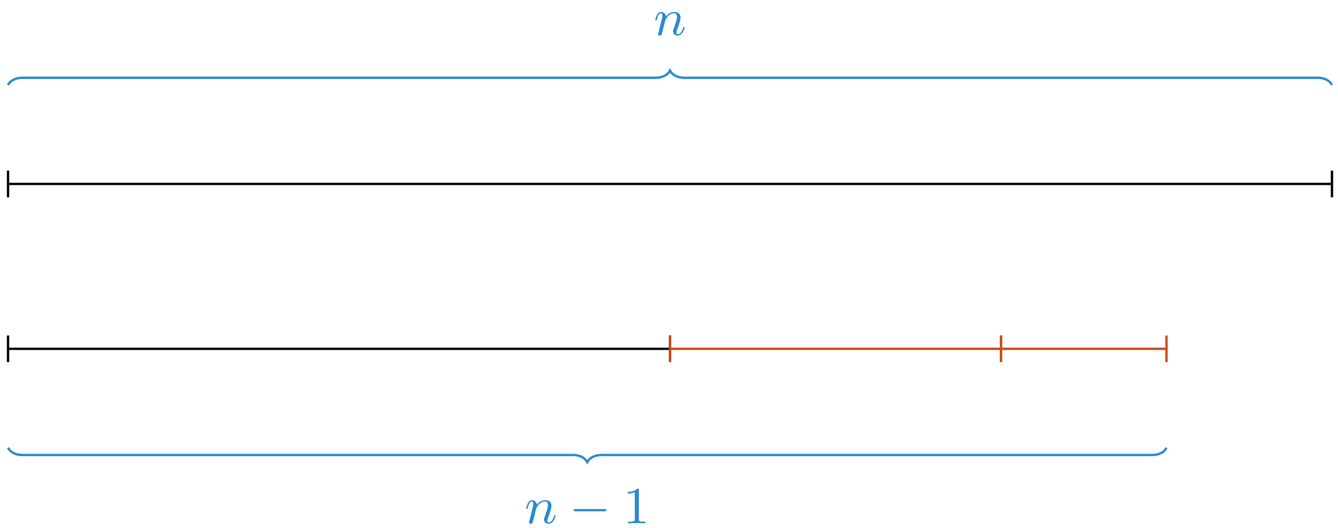




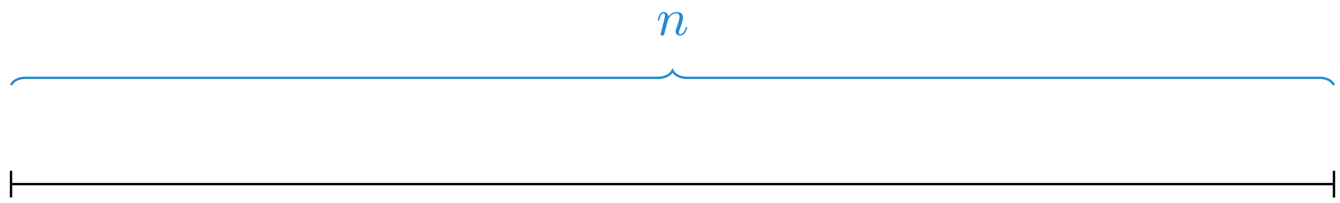


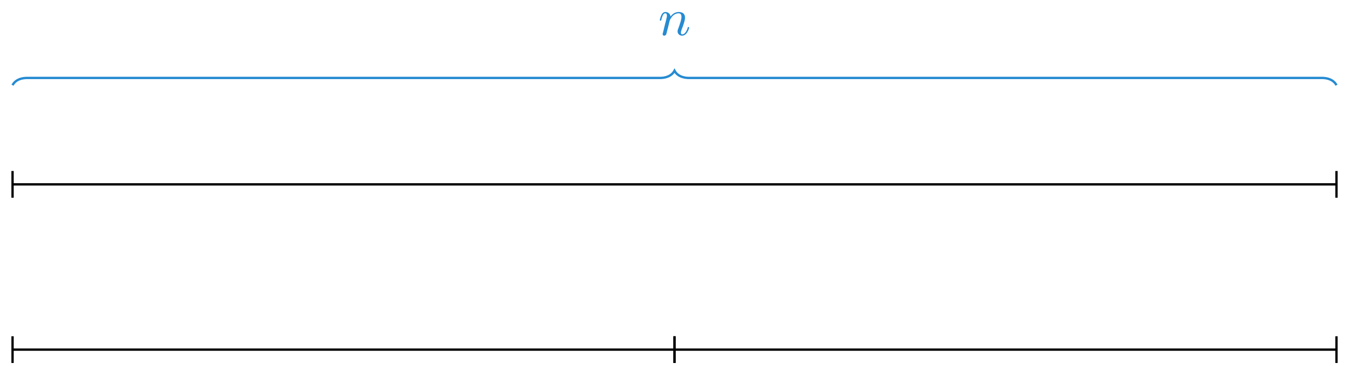
n

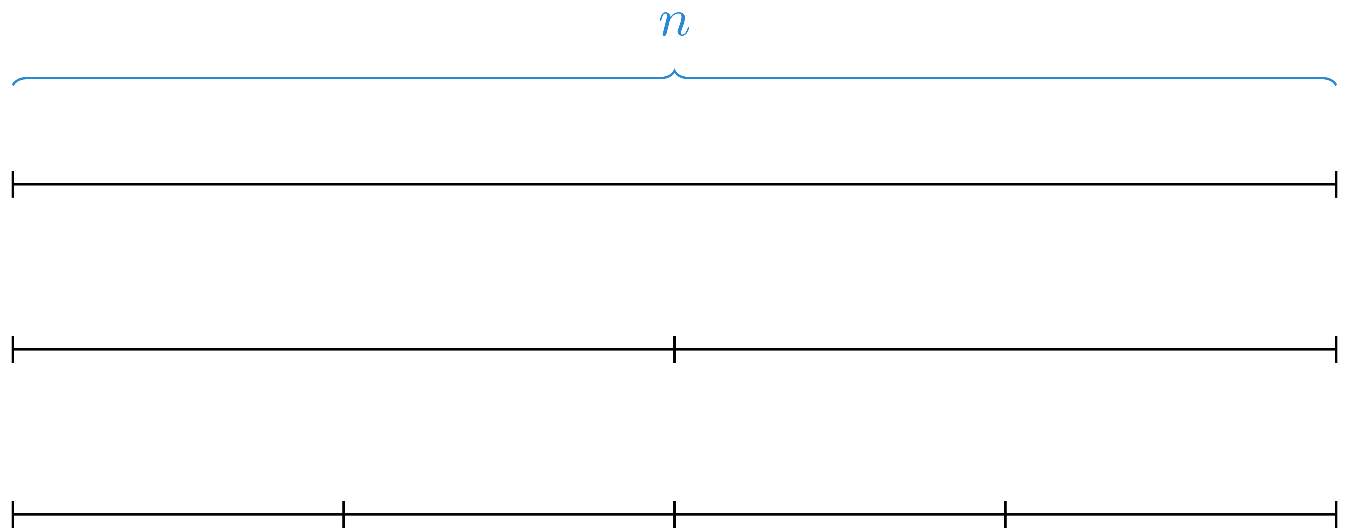

n
$n-1$

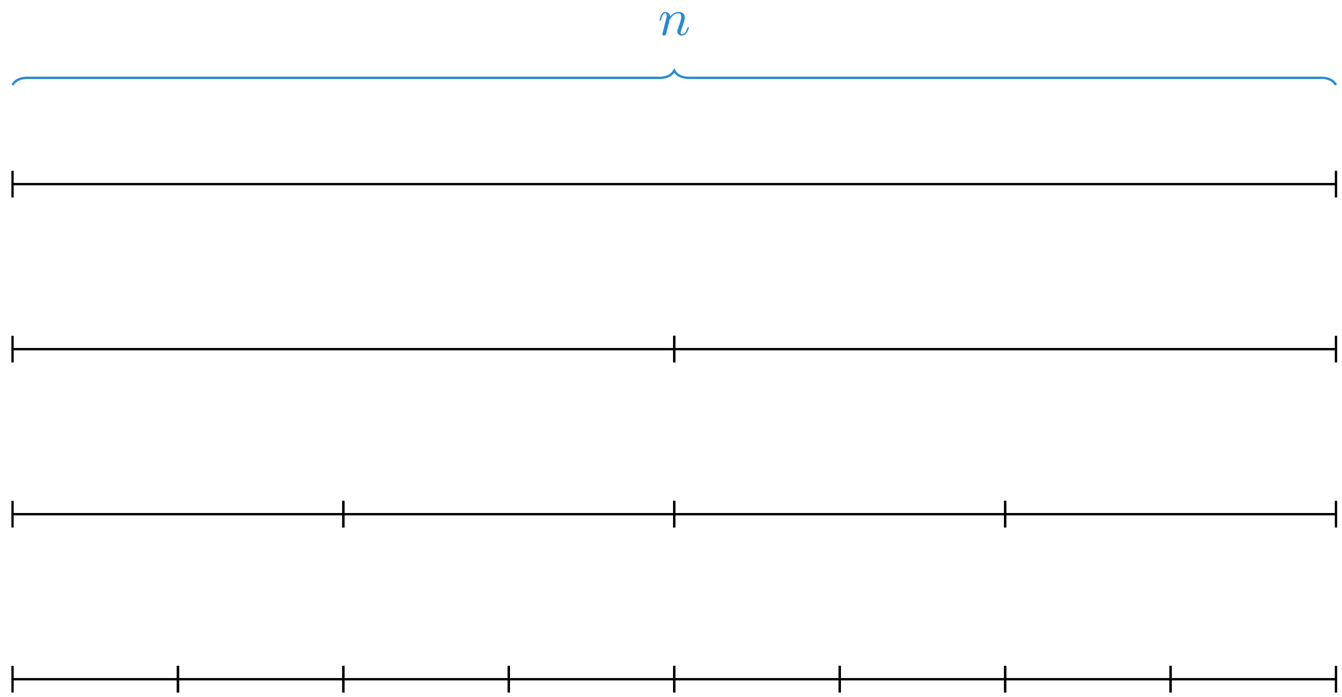


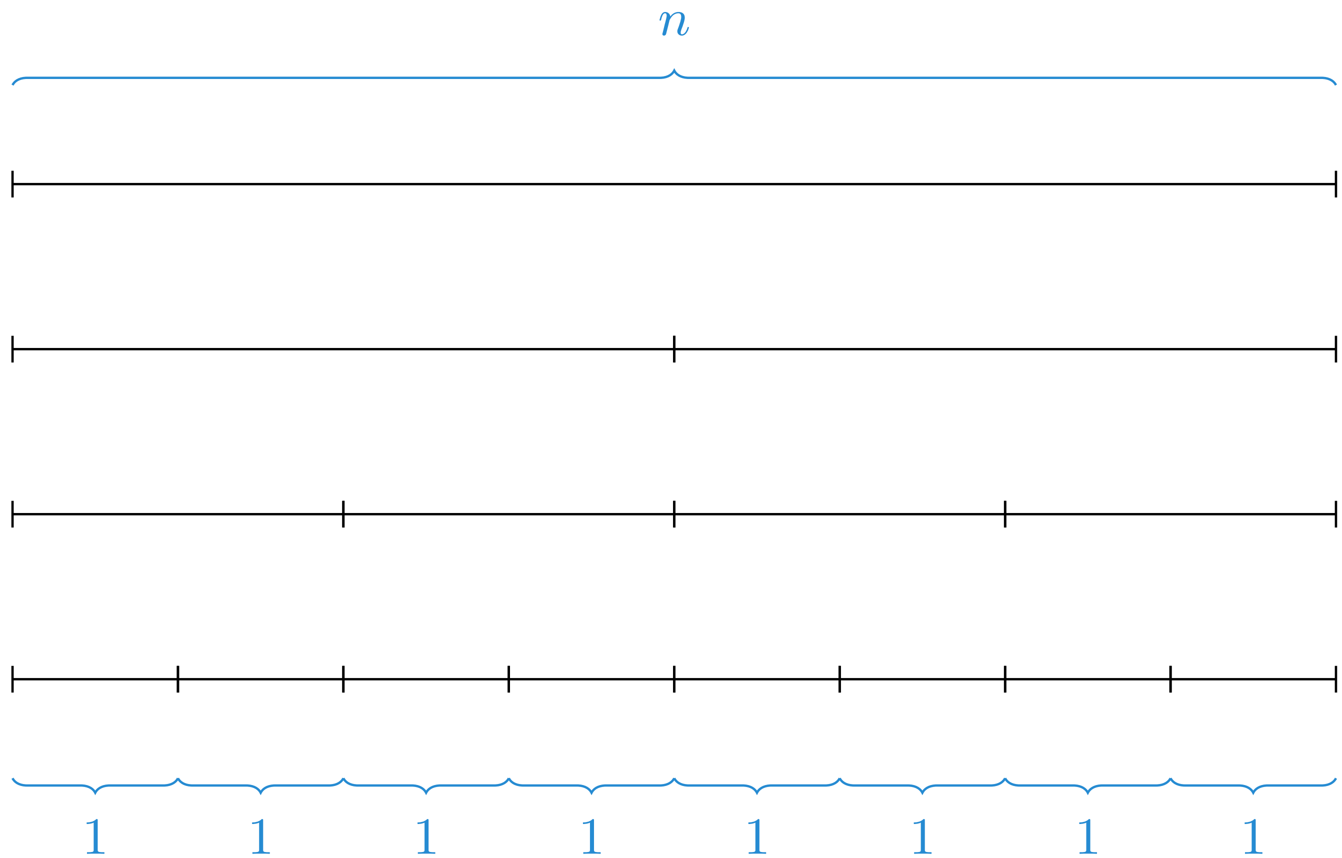
$$\sum_{i=1}^{\lg n} 2^i = 2n - 1$$


n

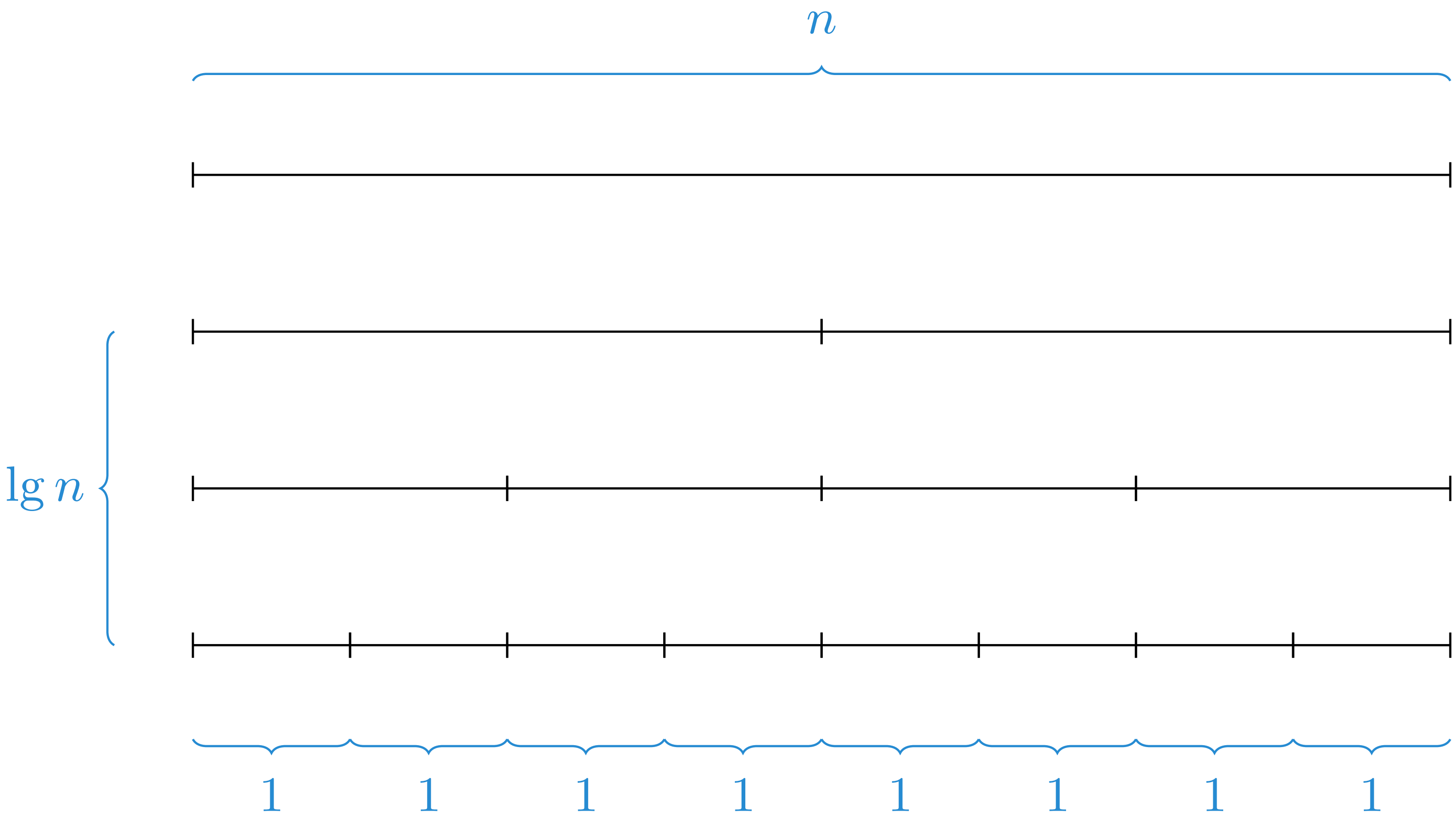

n


n

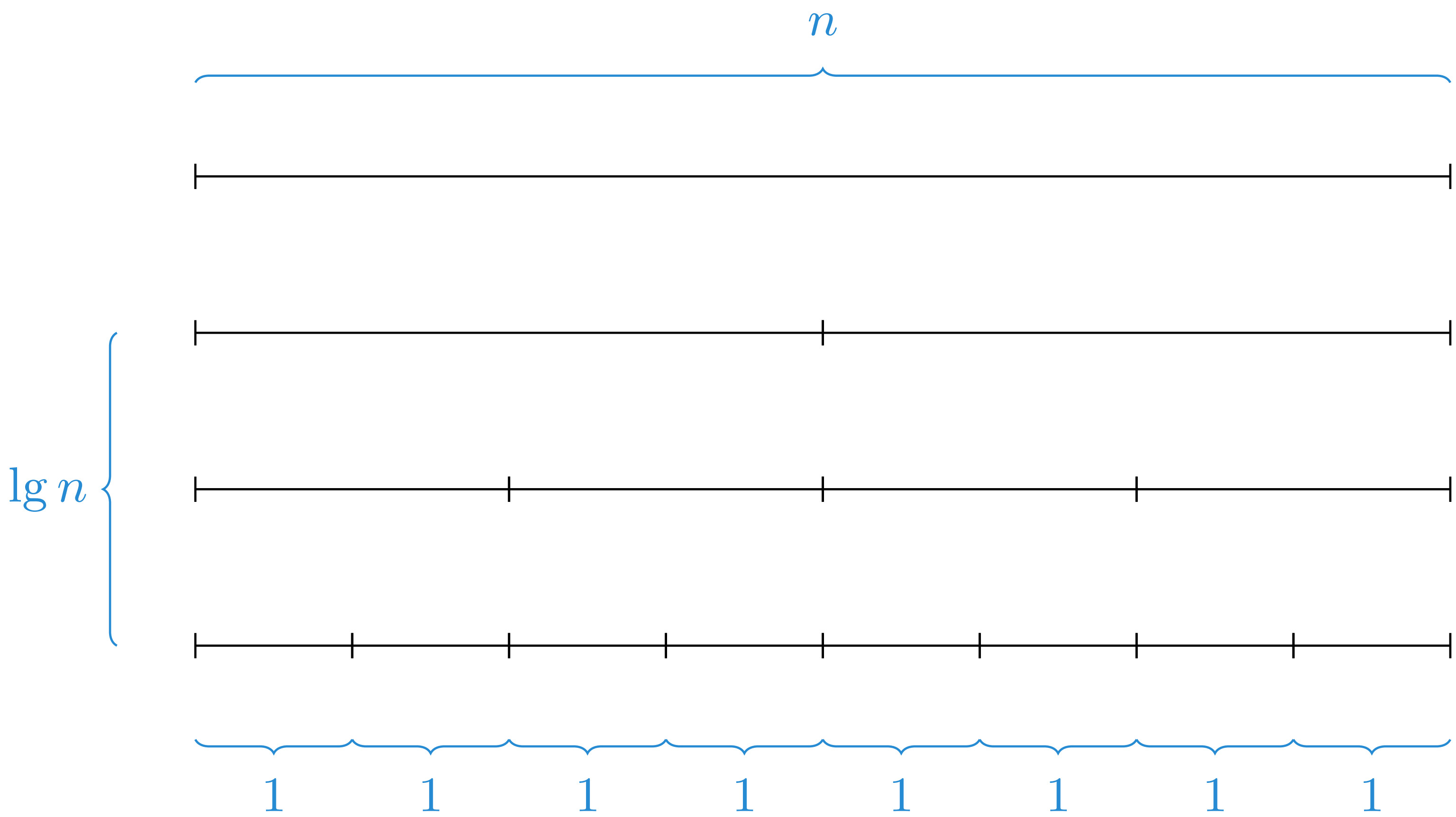

n



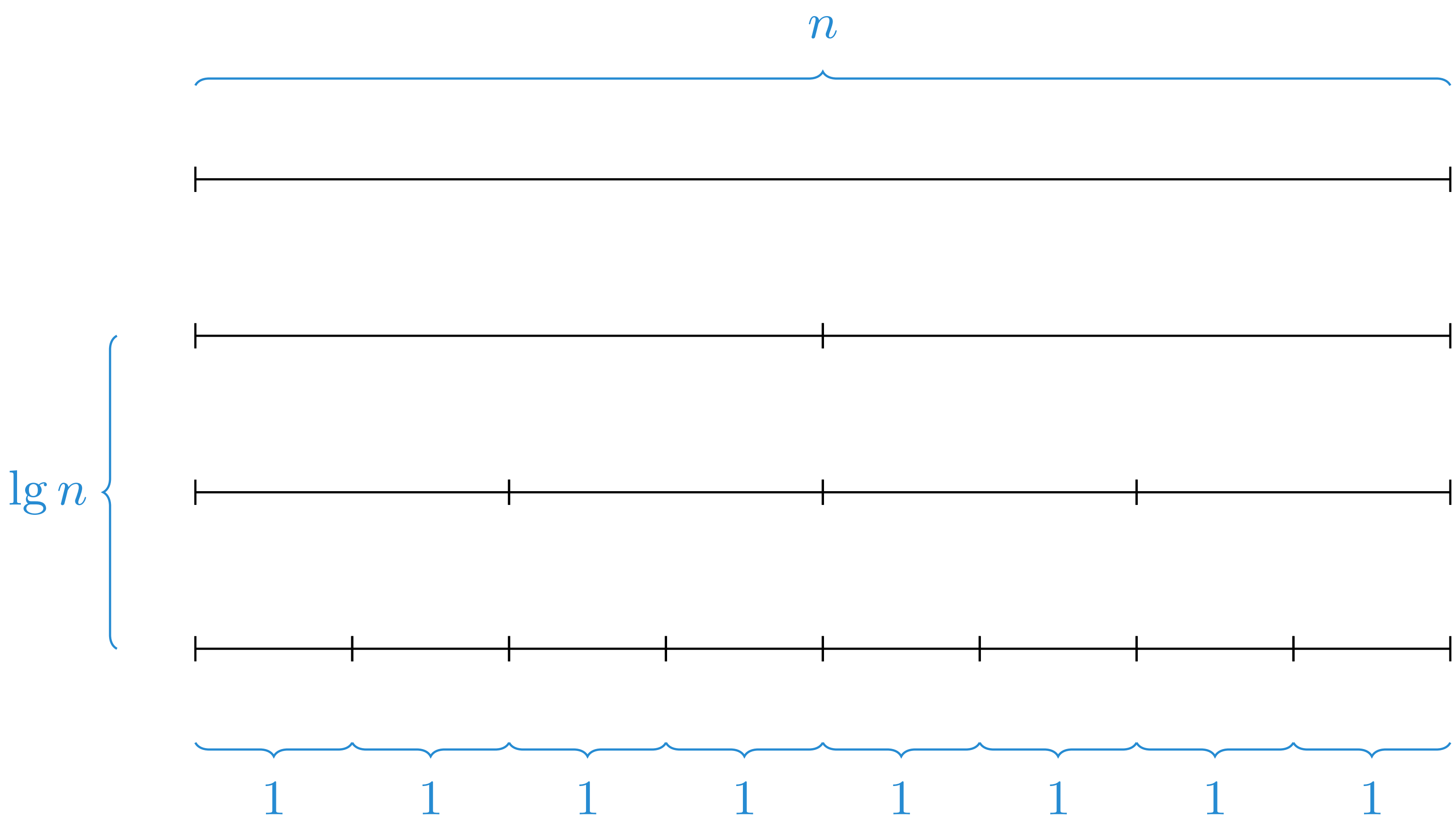
n
1
1
1
1
1
1
1
1



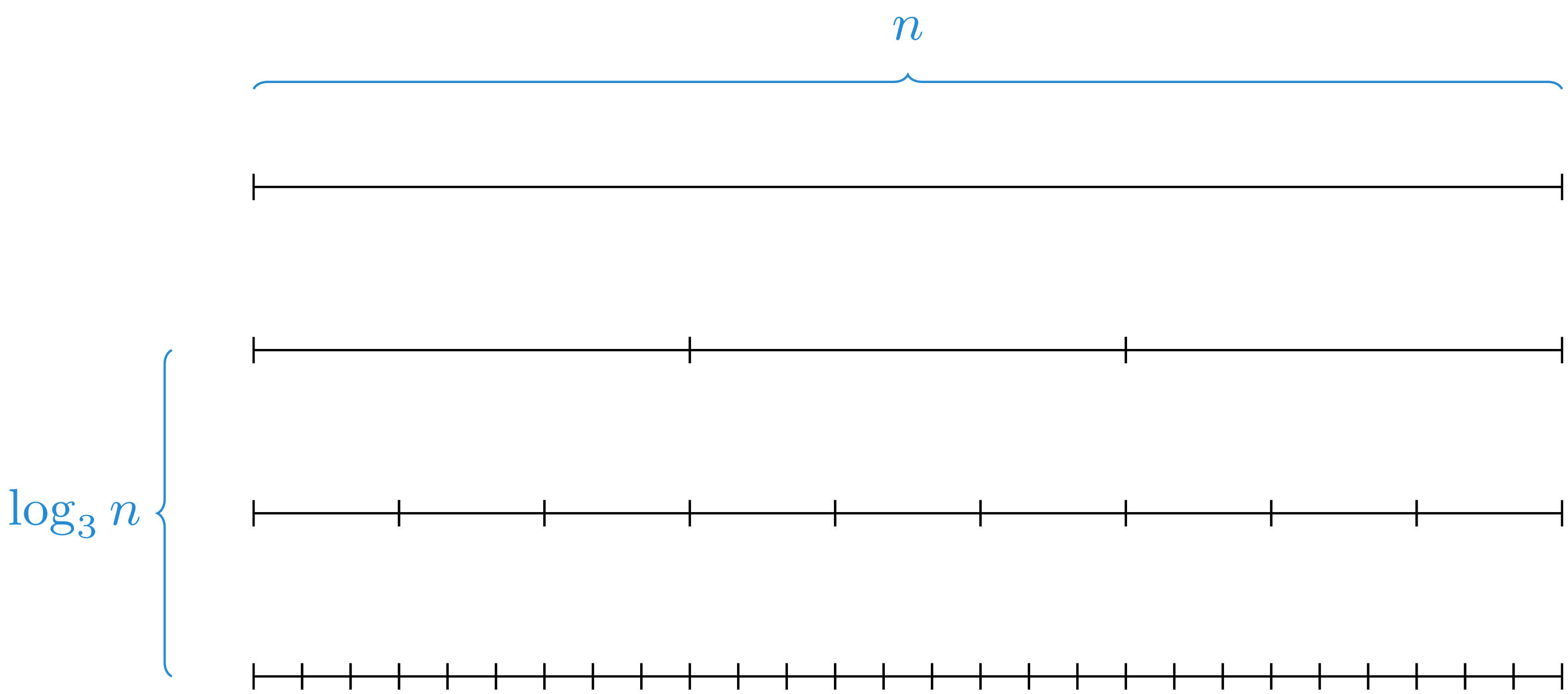
n
$\lg n$
1
1
1
1
1
1
1
1



$$\sum_{i=0}^{\lg n} 2^i \cdot \frac{n}{2^i} = n \cdot (\lg n + 1)$$

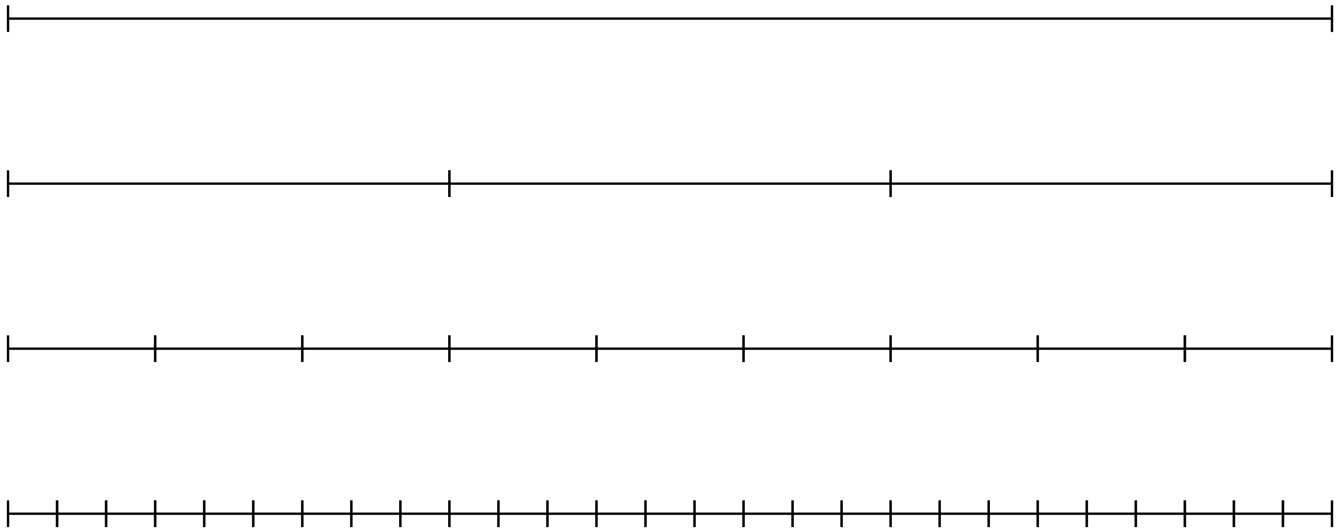


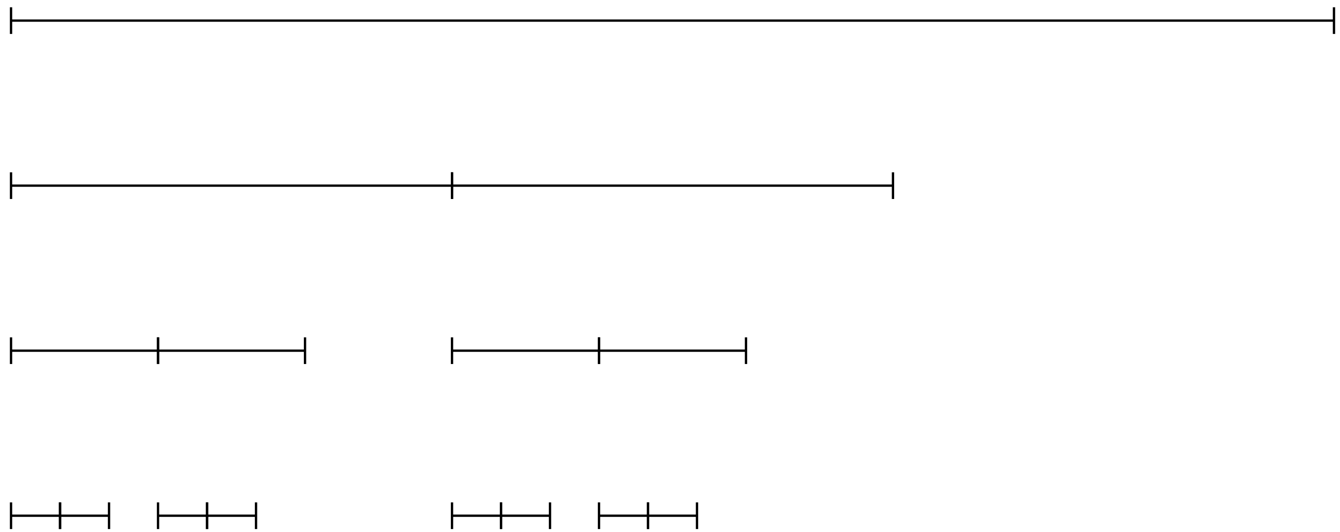
$$\sum_{i=0}^{\lg n} 2^i \cdot \frac{n}{2^i} = n \cdot (\lg n + 1) = n \lg n + n$$

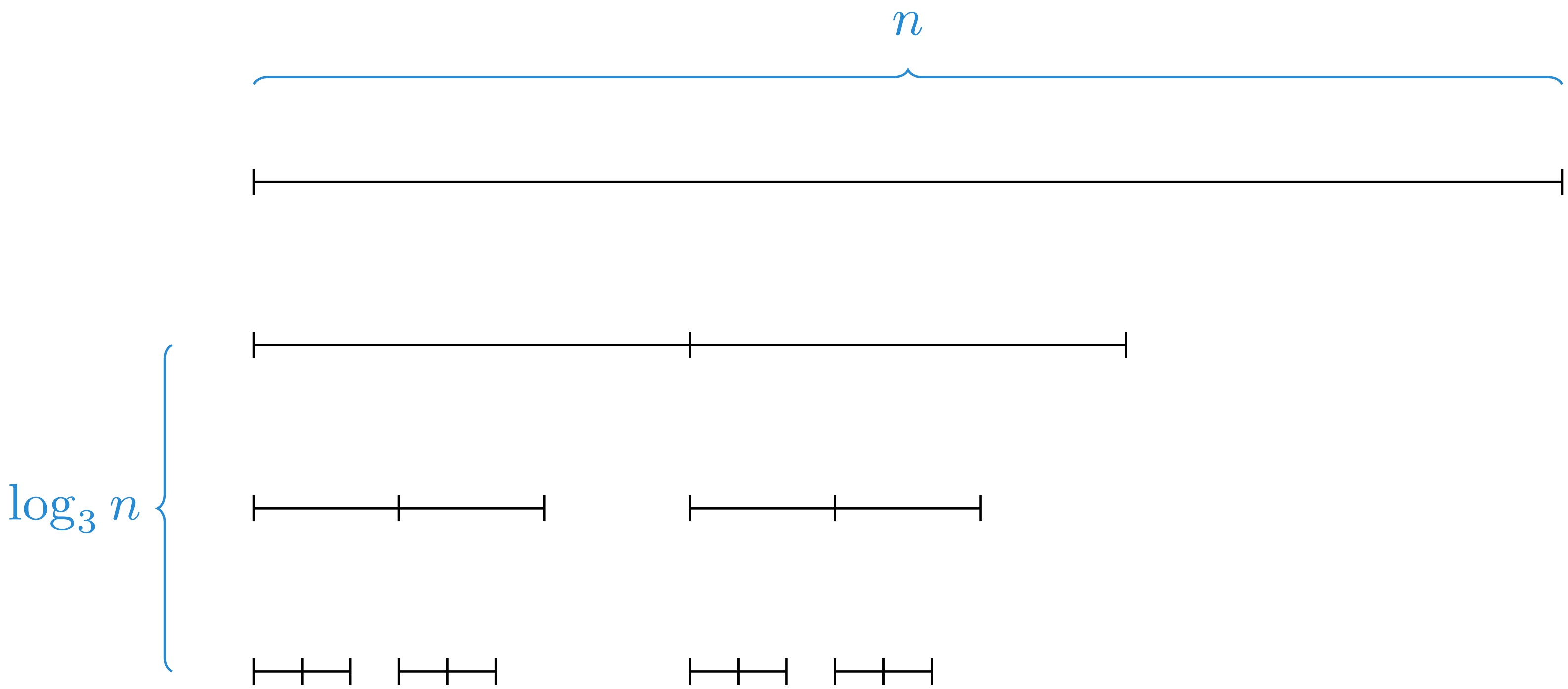


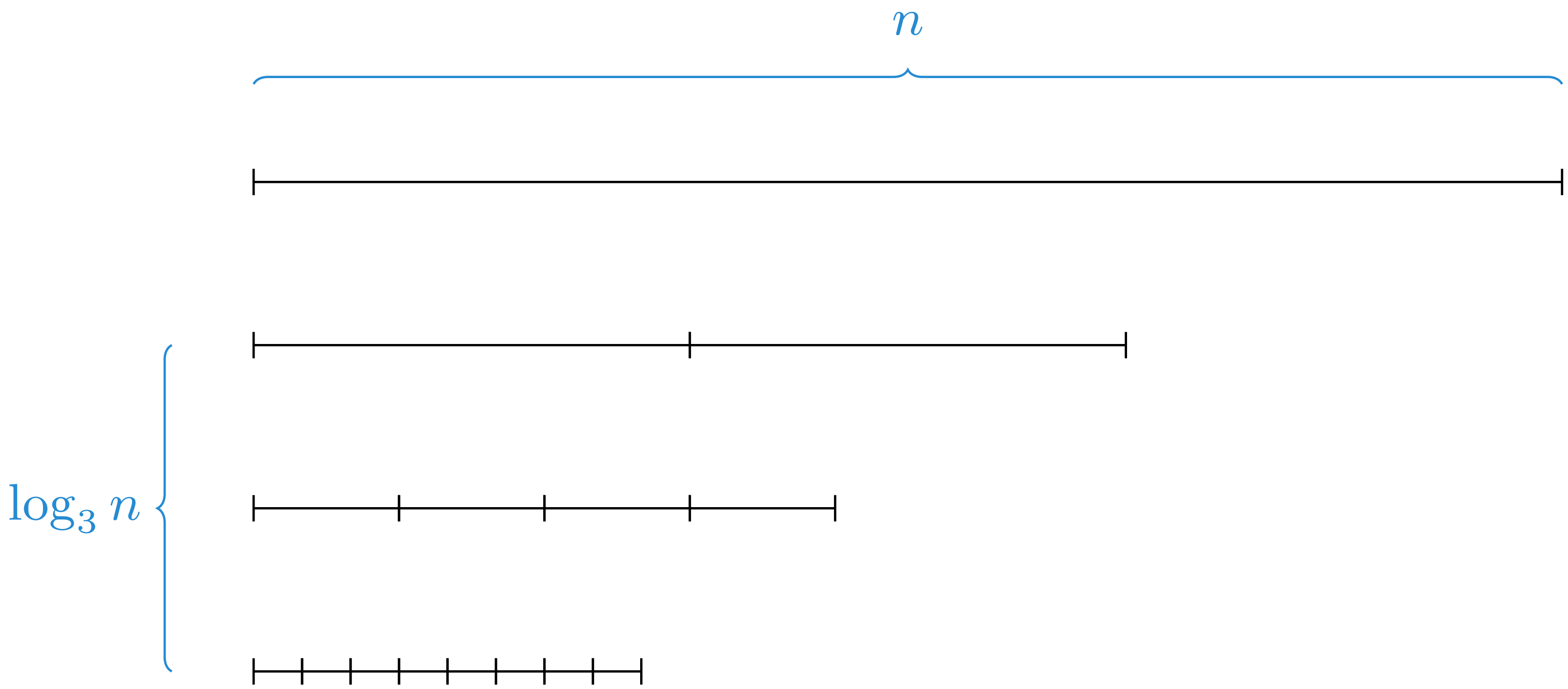
$1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1$

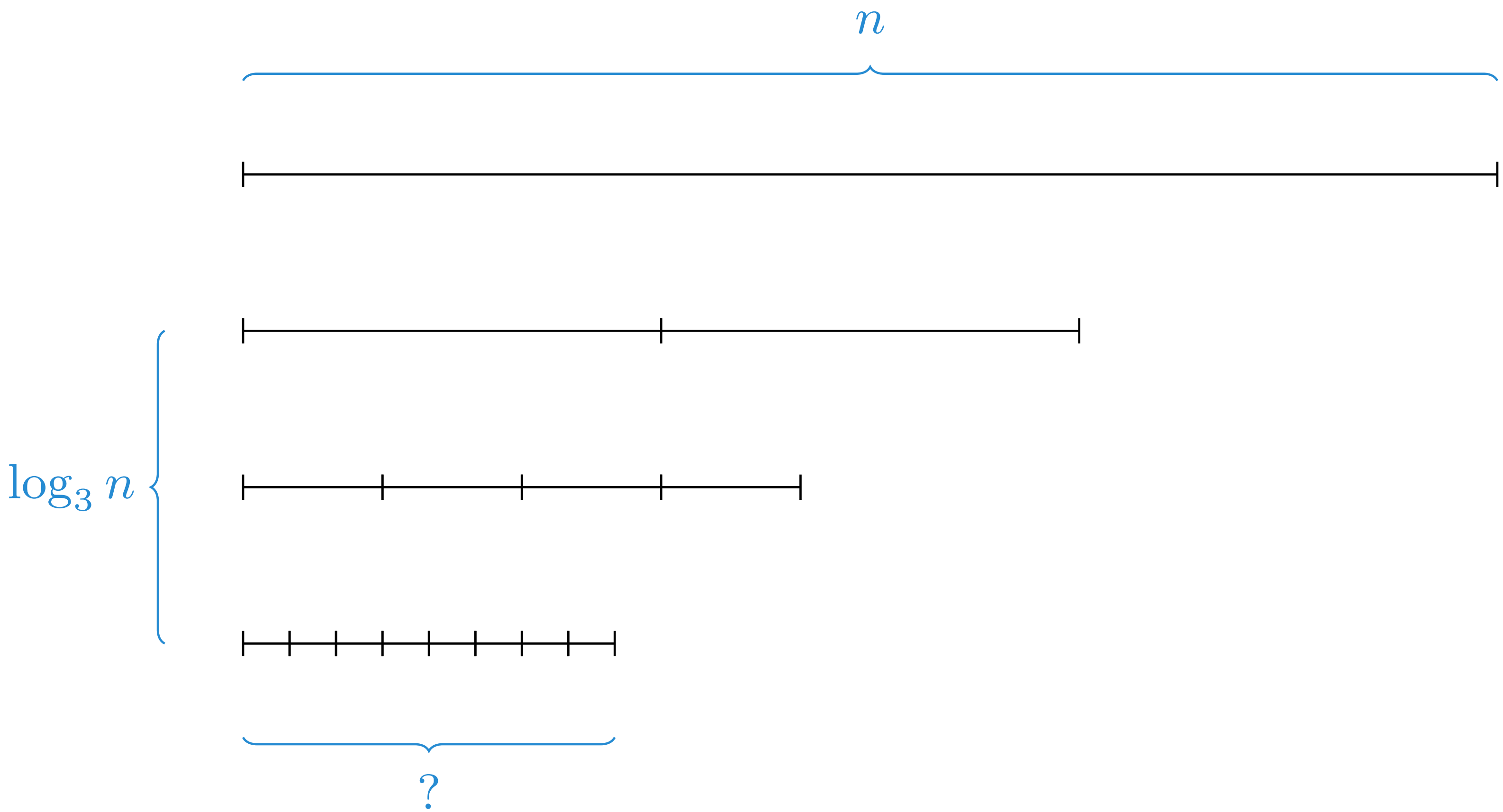
$$\sum_{i=0}^{\log_3 n} 3^i \cdot \frac{n}{3^i} = n \cdot (\log_3 n + 1) = n \log_3 + n$$

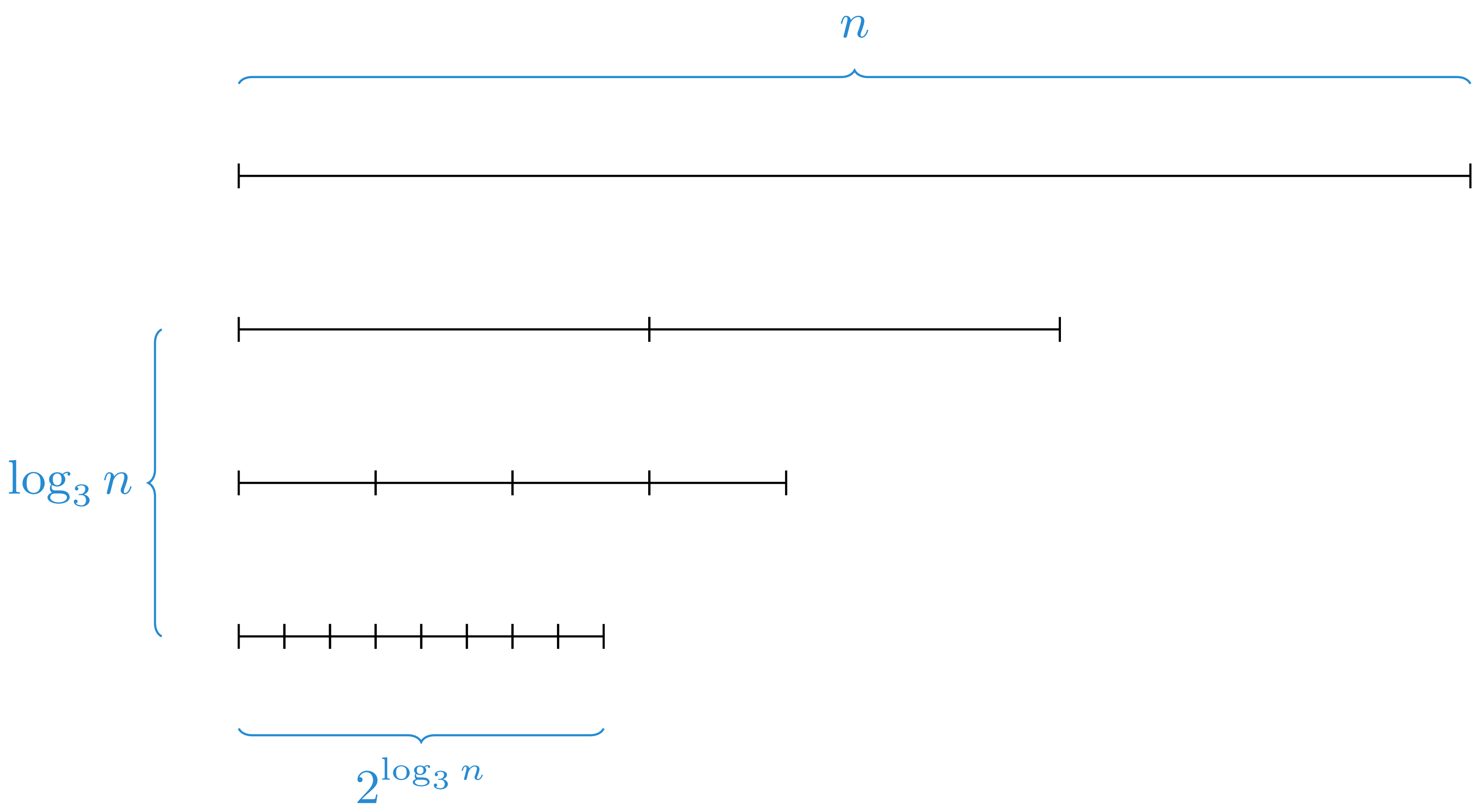


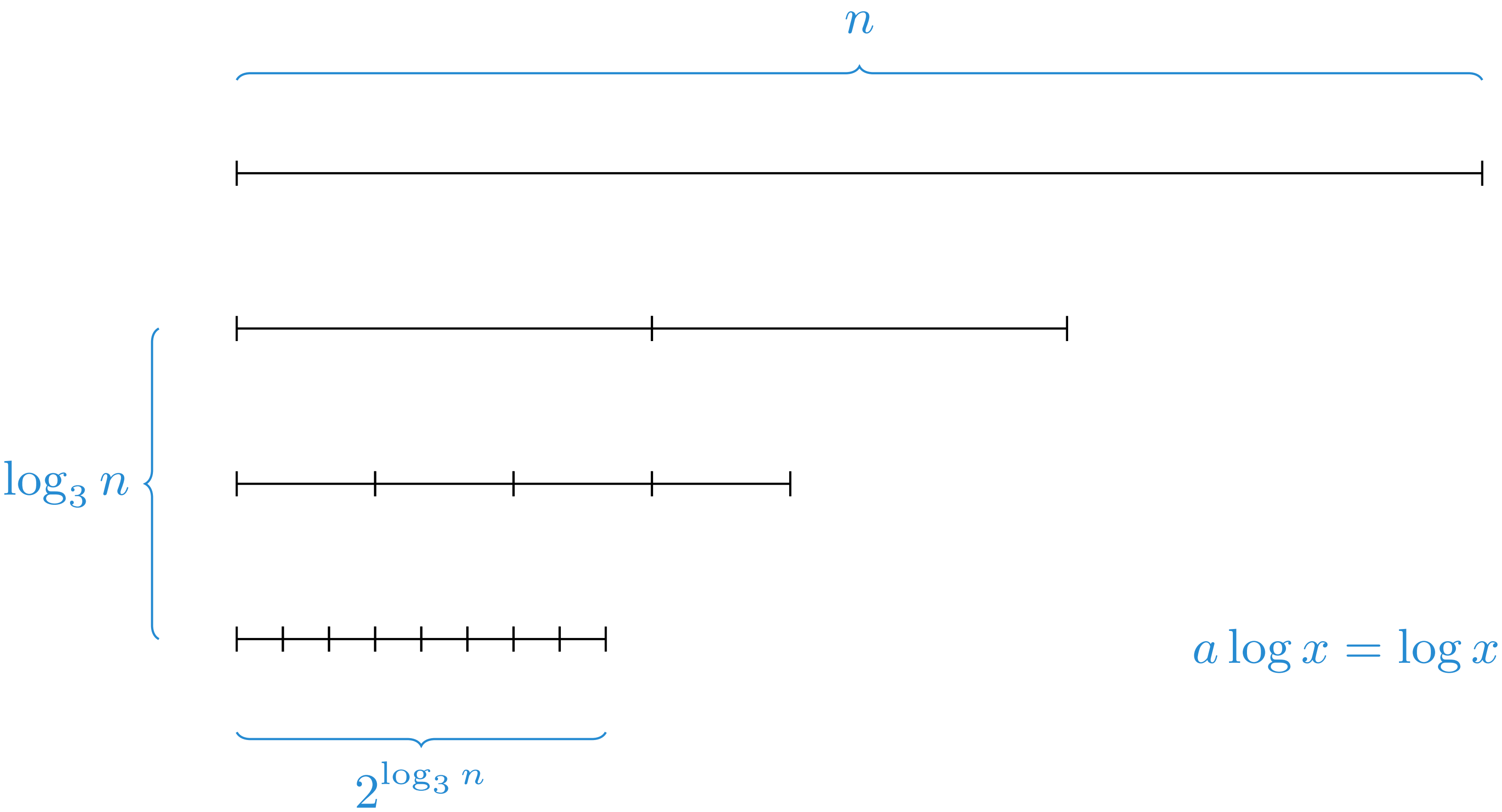



n
$\log_3 n$

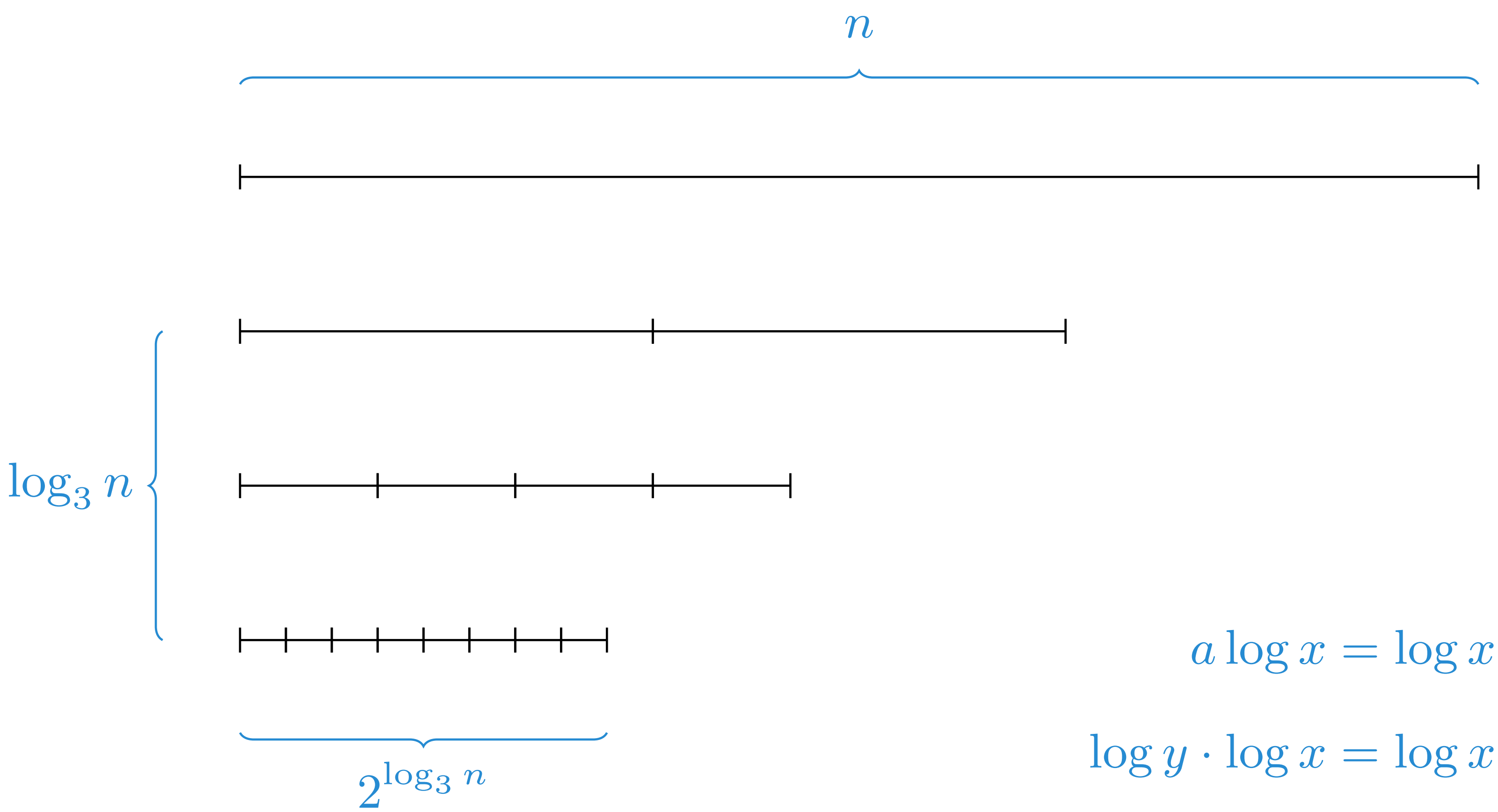

n
$\log_3 n$


n
$\log_3 n$
?


n
$\log_3 n$
$2^{\log_3 n}$

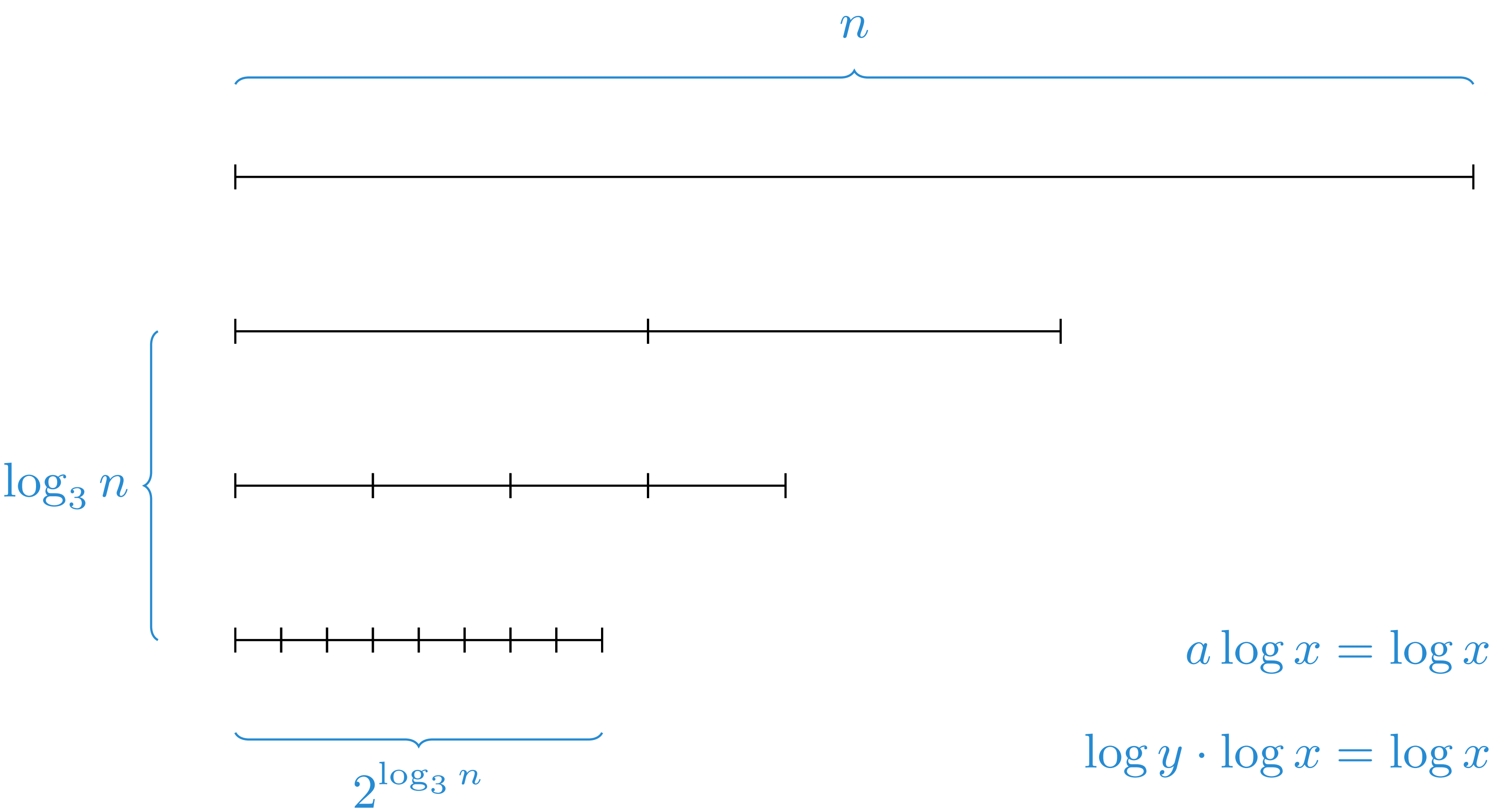


$$a \log x = \log x^a$$



$$a \log x = \log x^a$$

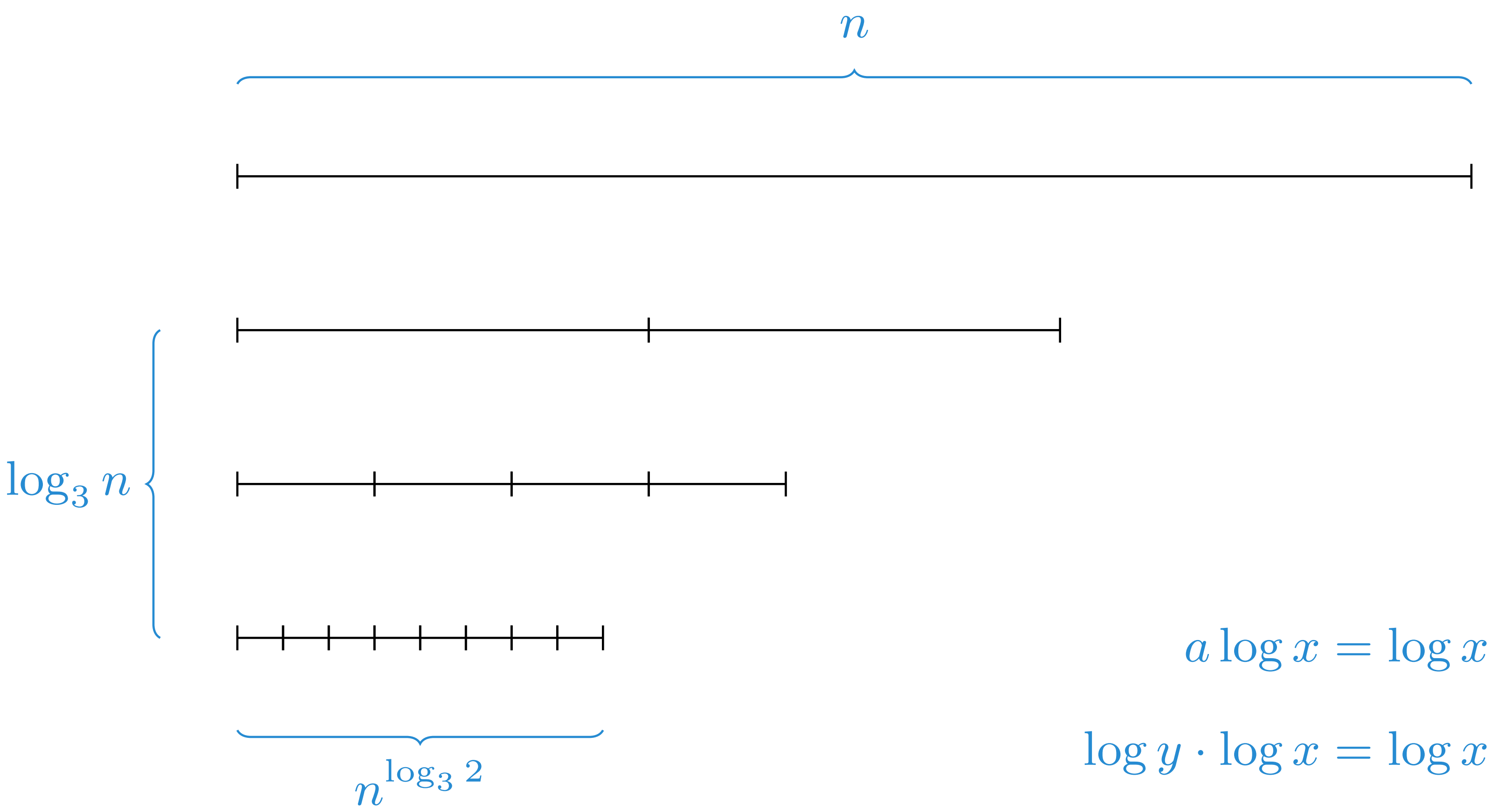
$$\log y \cdot \log x = \log x \cdot \log y$$



$$a \log x = \log x^a$$

$$\log y \cdot \log x = \log x \cdot \log y$$

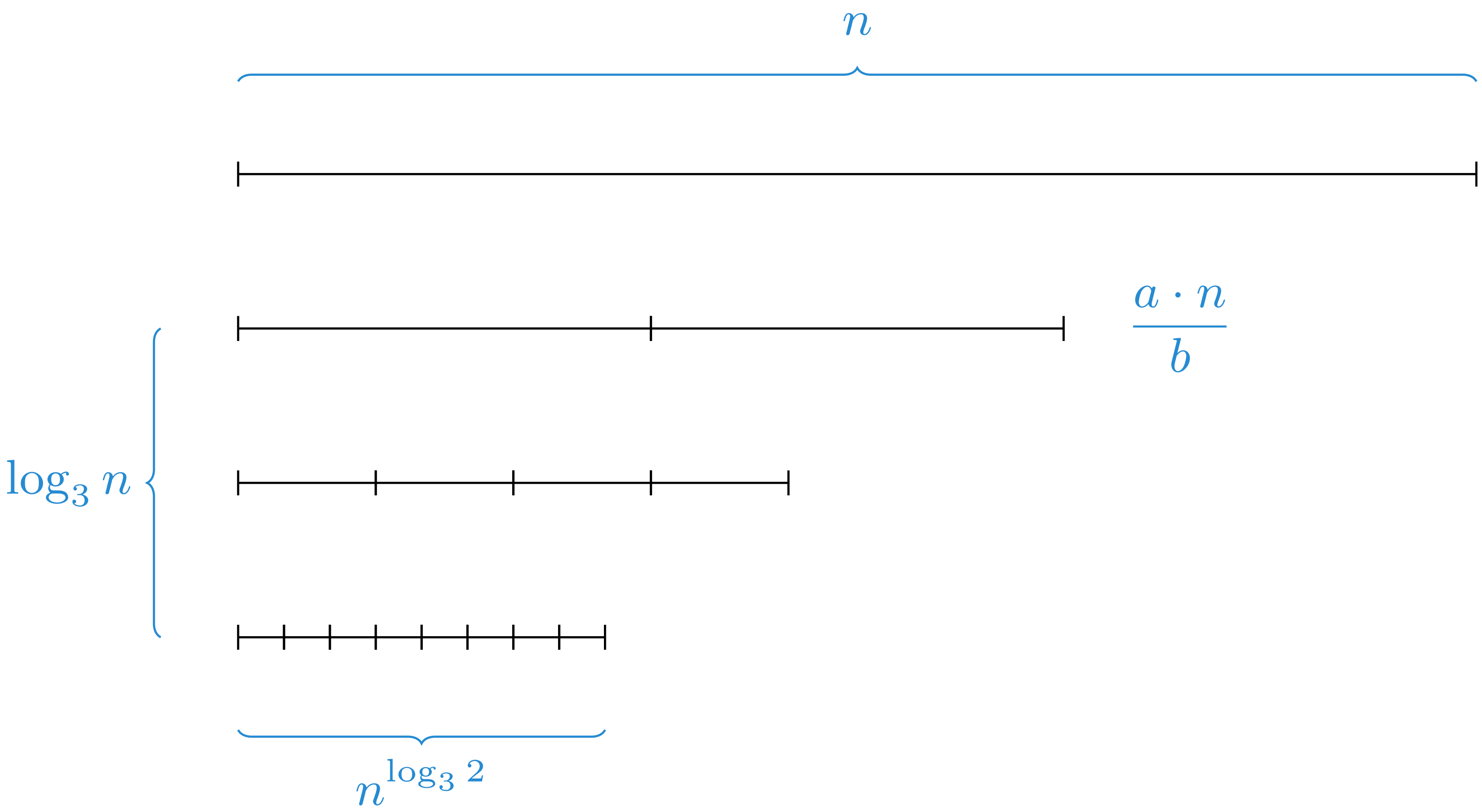
$$\log\left(x^{\log y}\right) = \log\left(y^{\log x}\right)$$

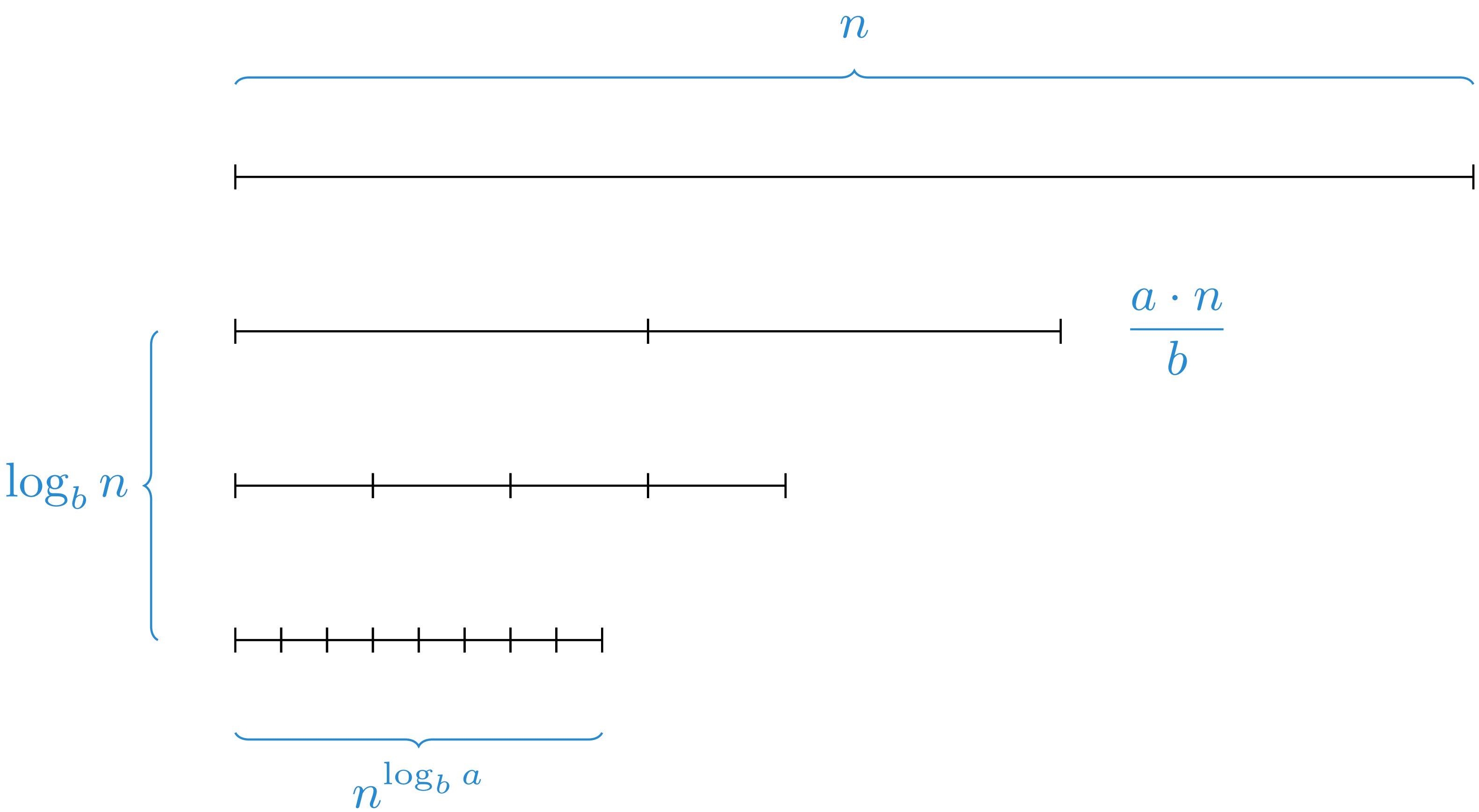


$$a \log x = \log x^a$$

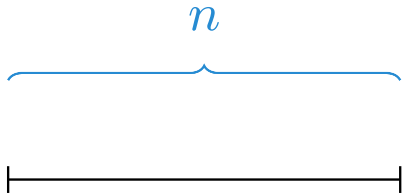
$$\log y \cdot \log x = \log x \cdot \log y$$

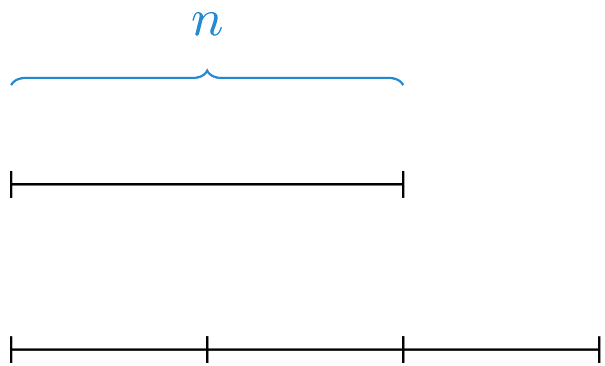
$$\log\left(x^{\log y}\right) = \log\left(y^{\log x}\right)$$

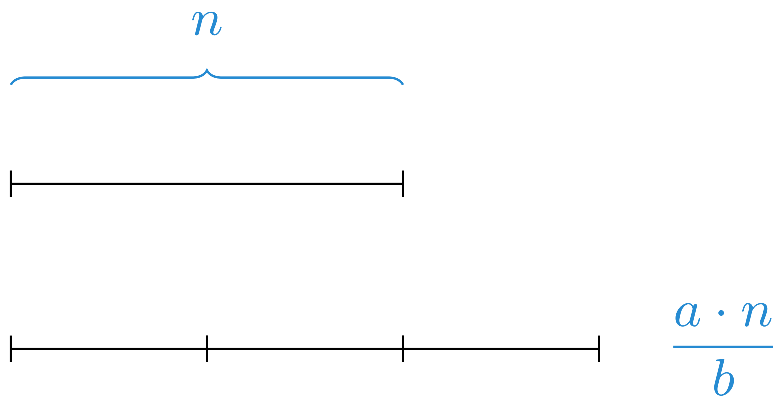

n
$\frac{a \cdot n}{b}$
$\log_3 n$
$n^{\log_3 2}$

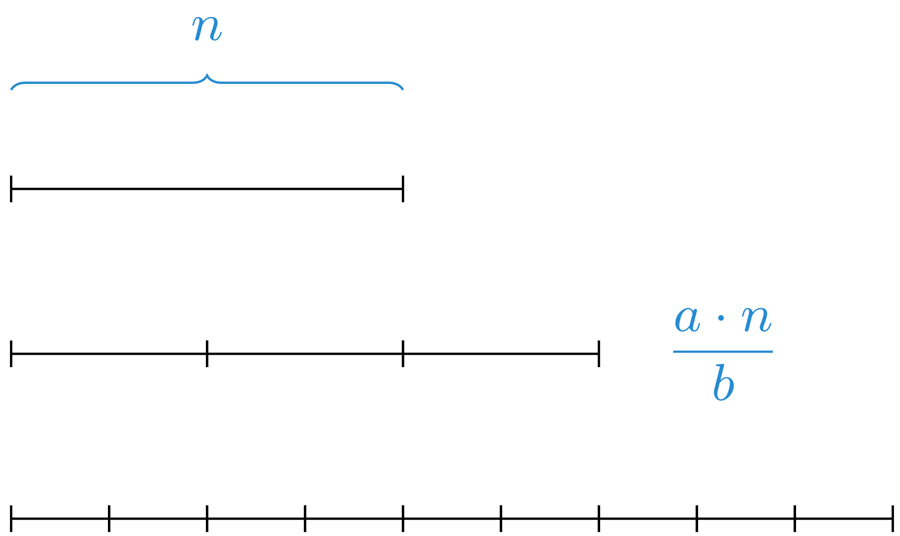


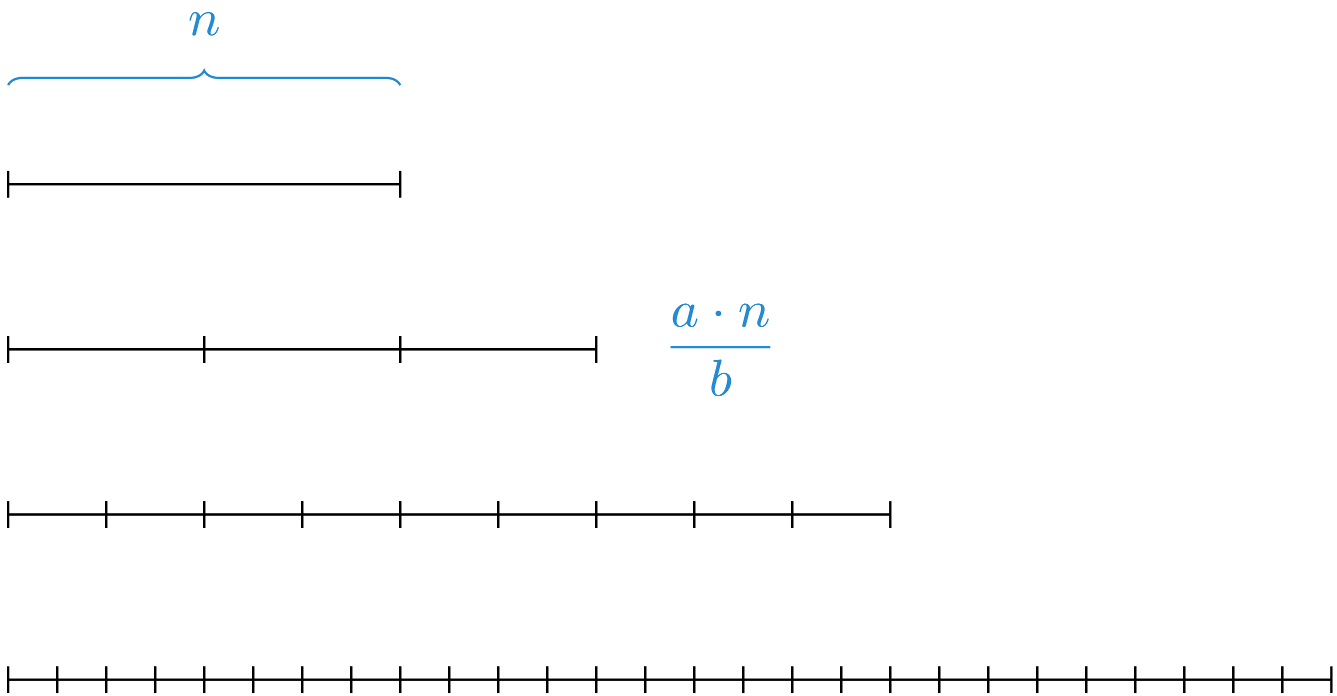
n
$\frac{a \cdot n}{b}$
$\log_b n$
$n^{\log_b a}$

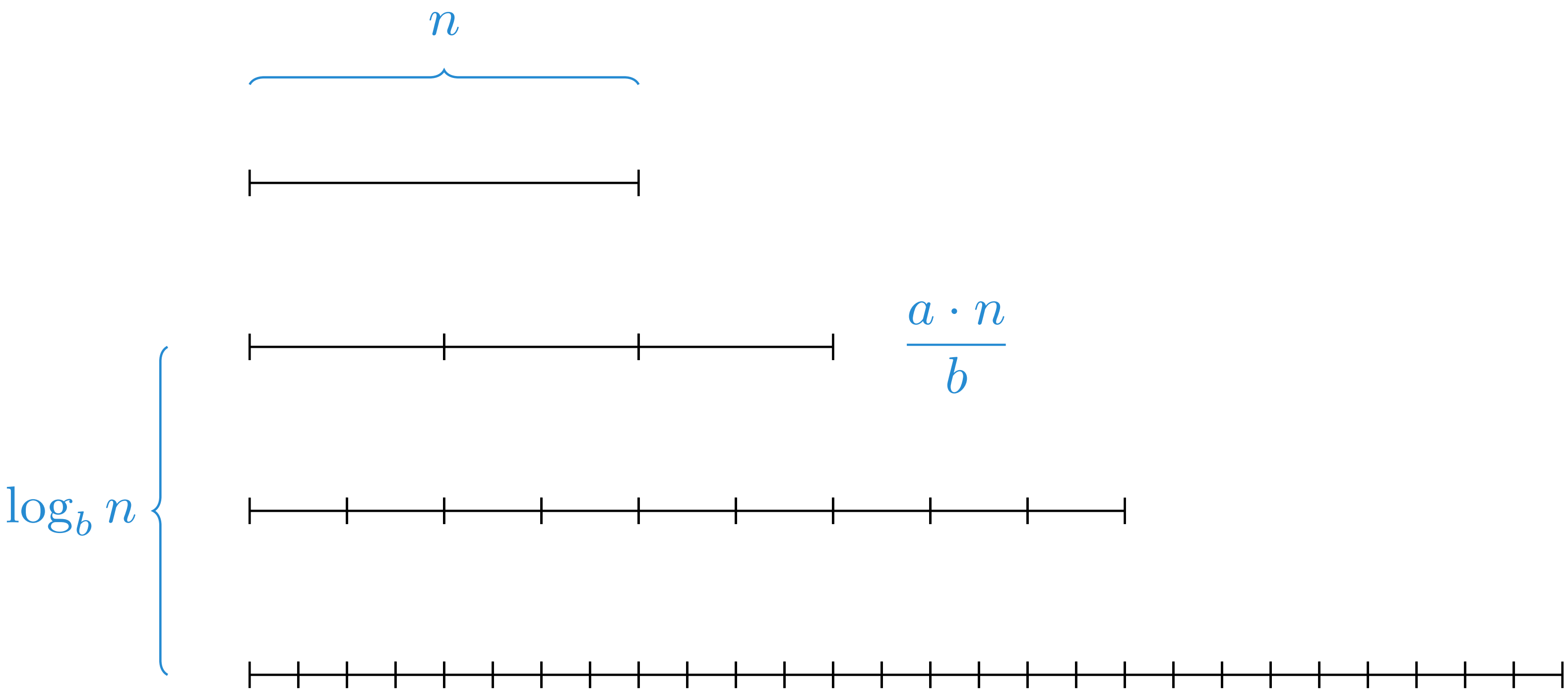

n

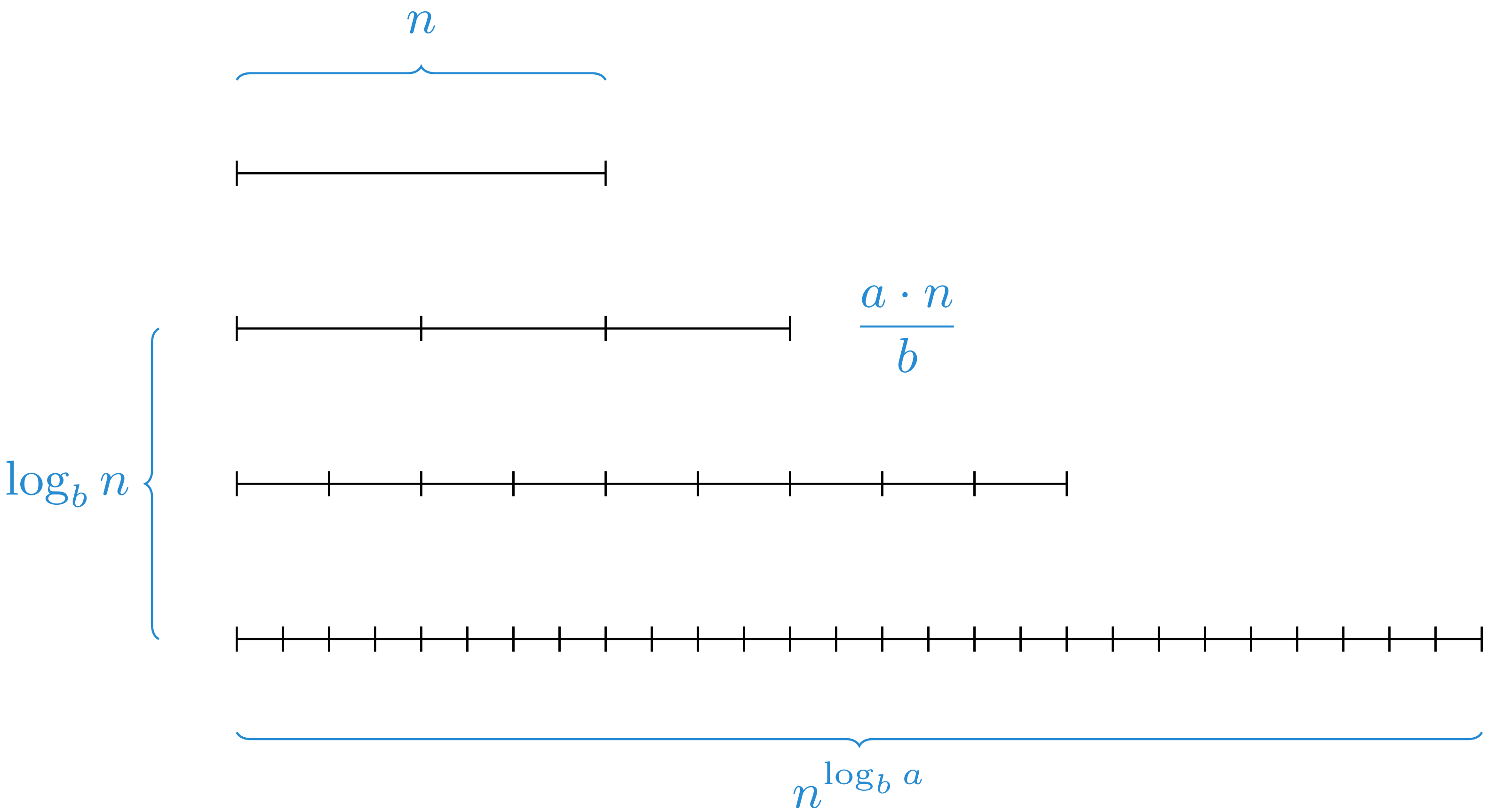

n

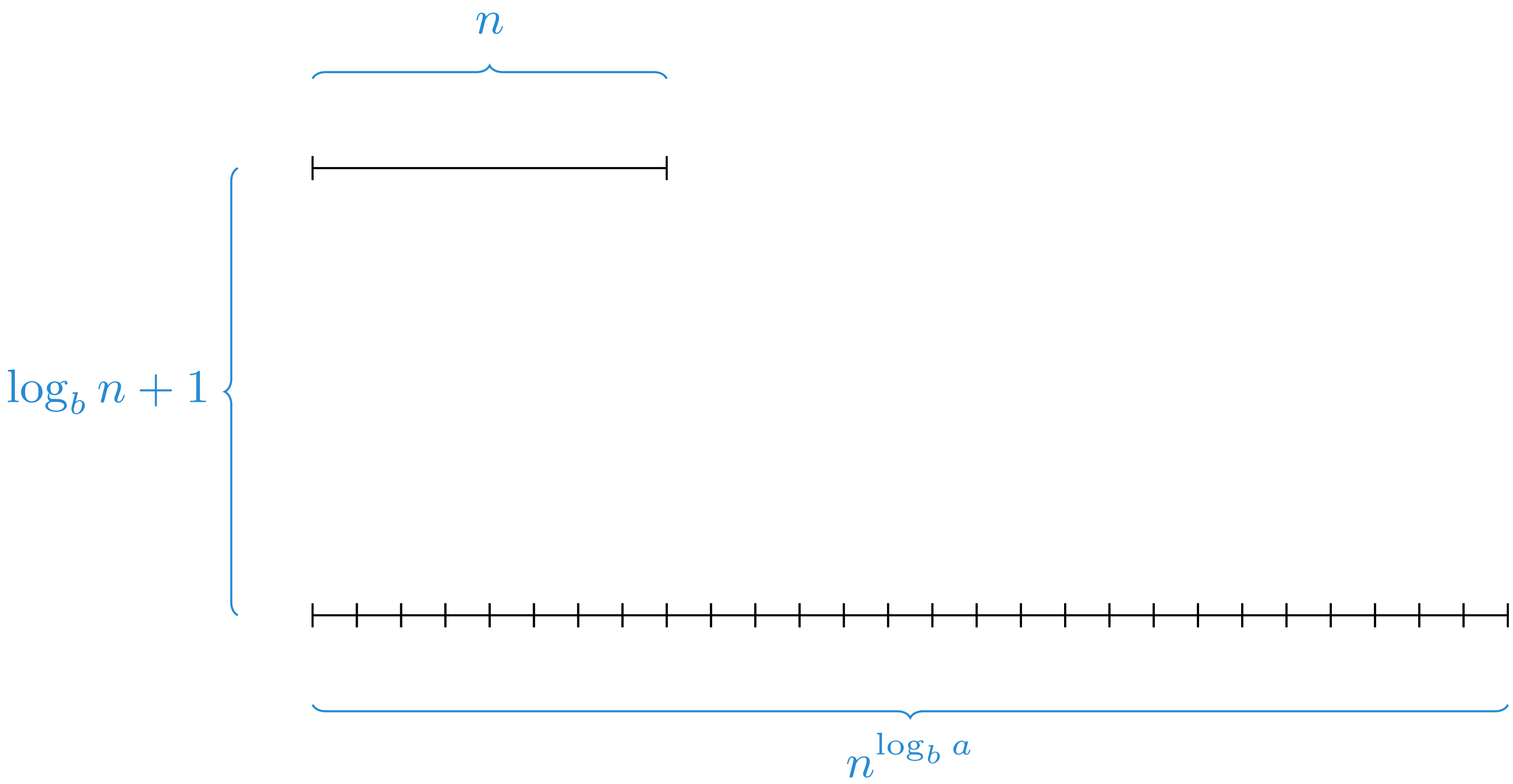

n
$\frac{a \cdot n}{b}$

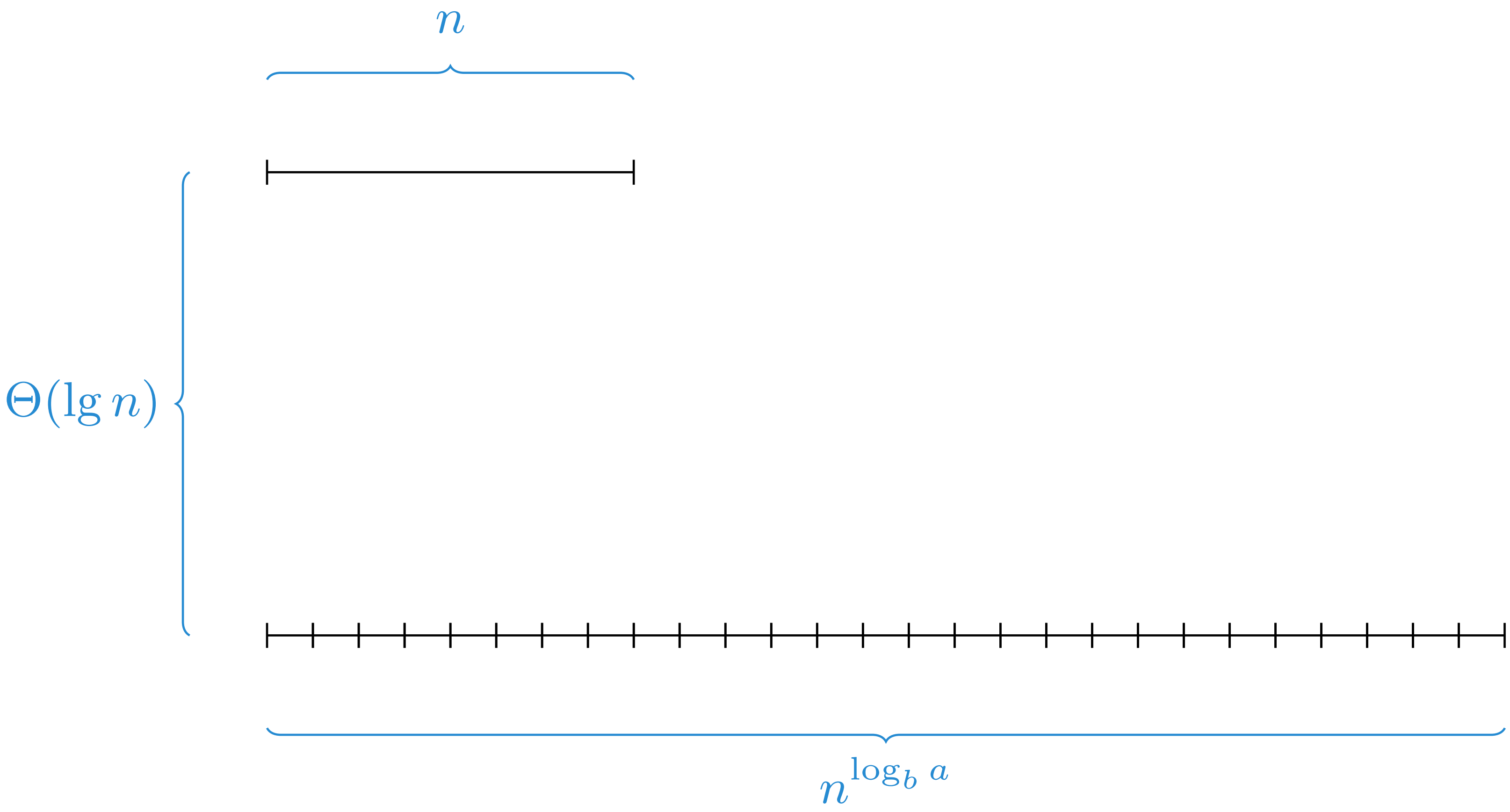

n
$\frac{a \cdot n}{b}$


n
$\frac{a \cdot n}{b}$

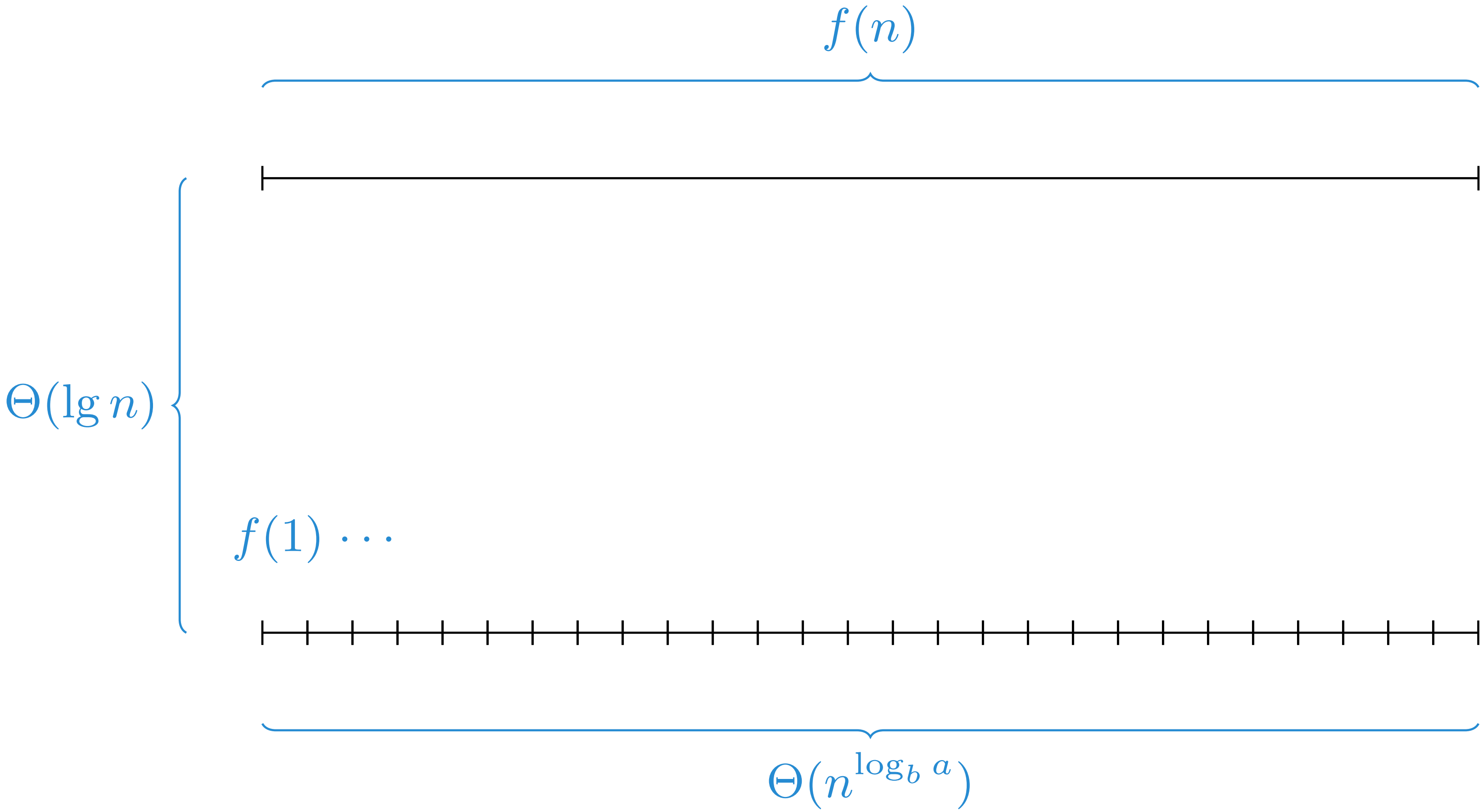

n
$\frac{a \cdot n}{b}$
$\log_b n$

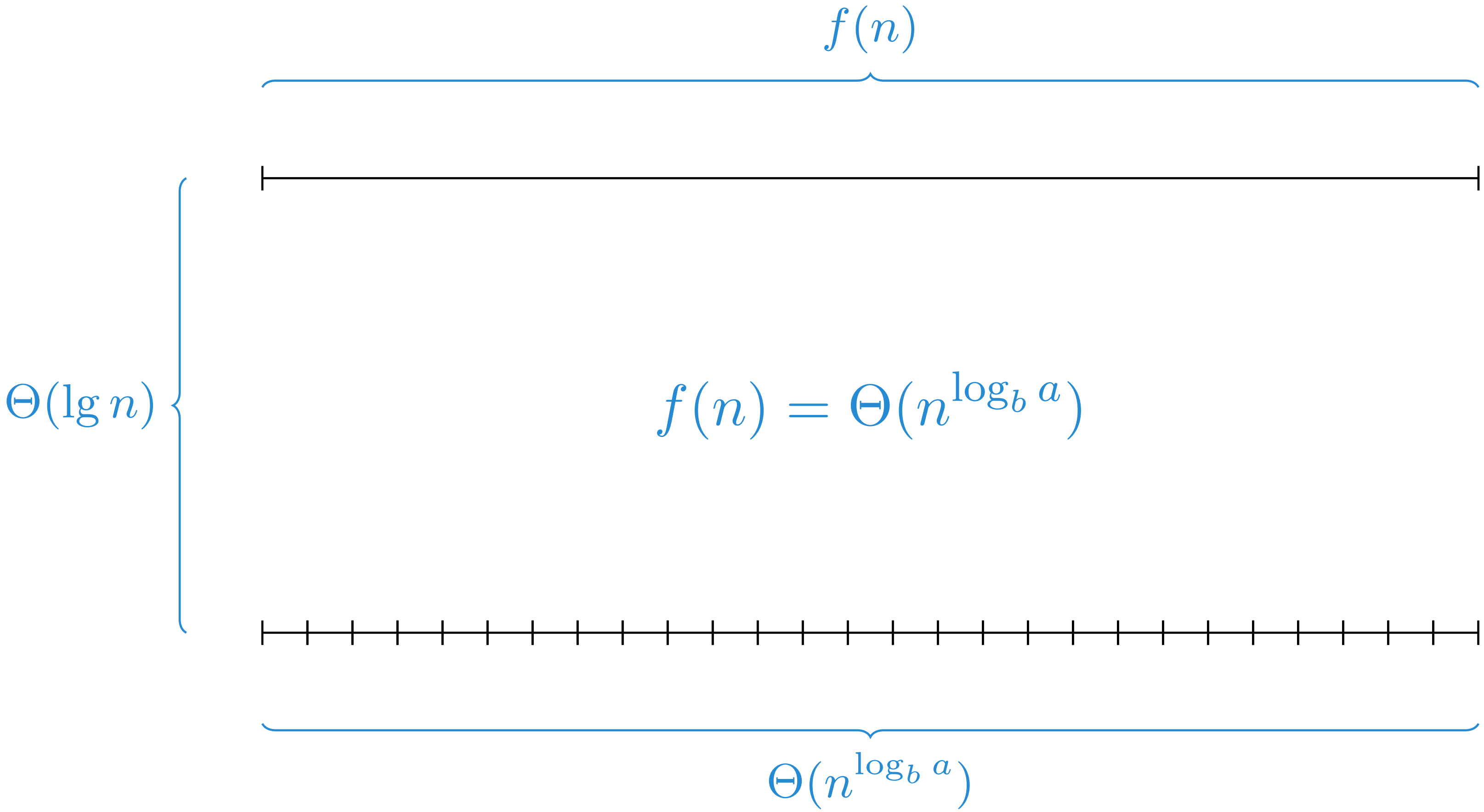

n
$\frac{a \cdot n}{b}$
$\log_b n$
$n^{\log_b a}$


n
$\log_b n + 1$
$n^{\log_b a}$

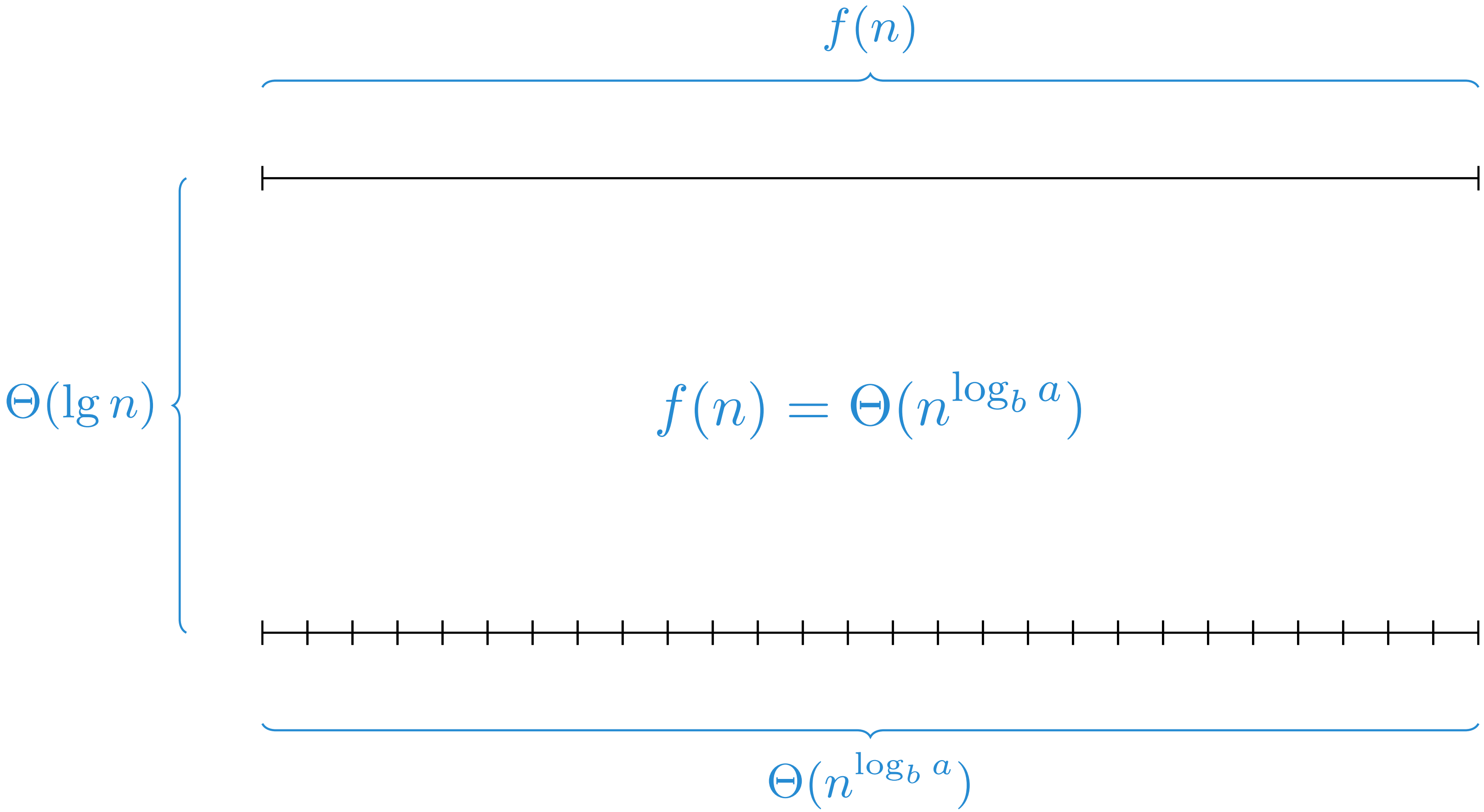


n
$\Theta(\lg n)$
$n^{\log_b a}$

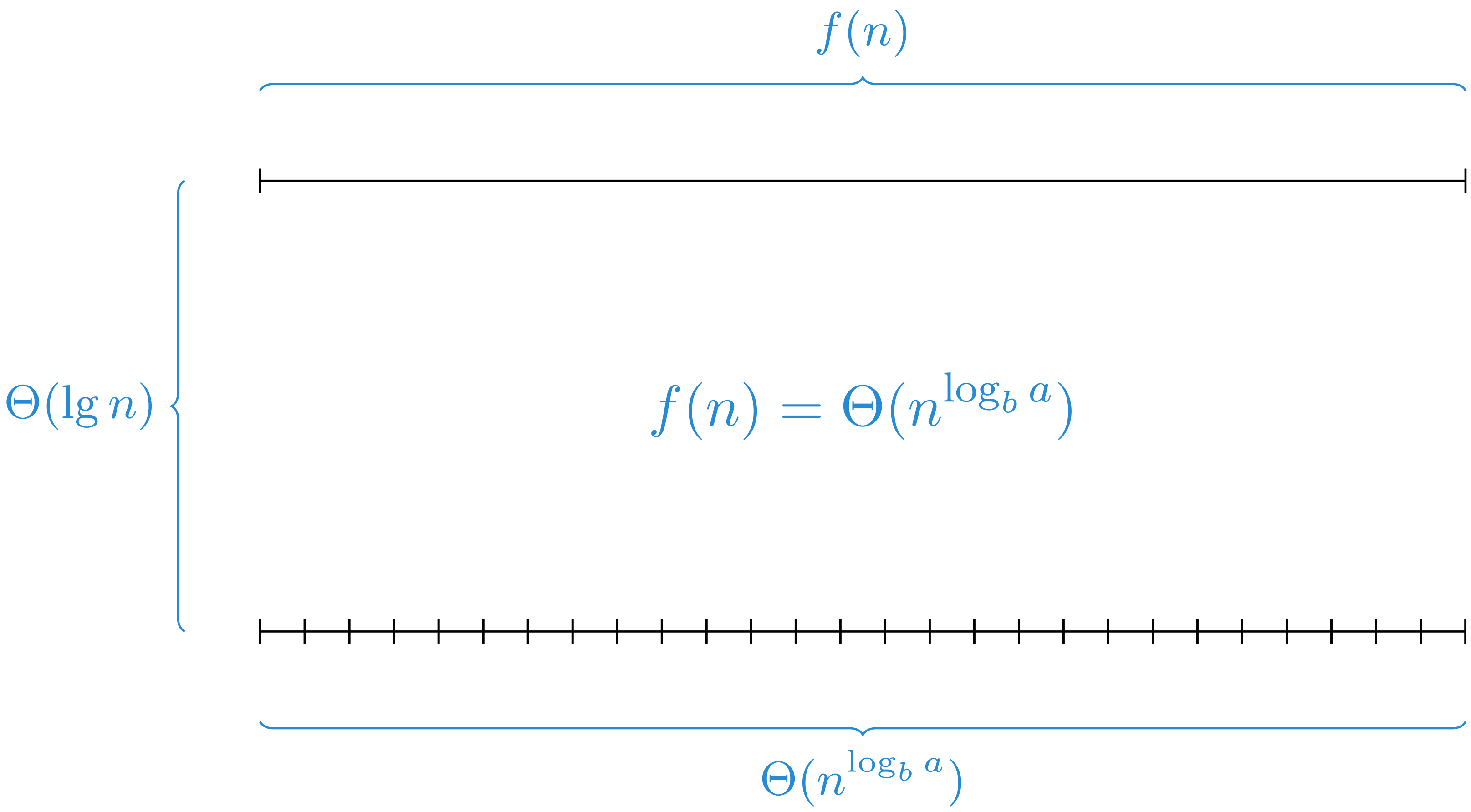

$f(n)$
$\Theta(\lg n)$
$f(1) \cdots$
$\Theta(n^{\log_b a})$



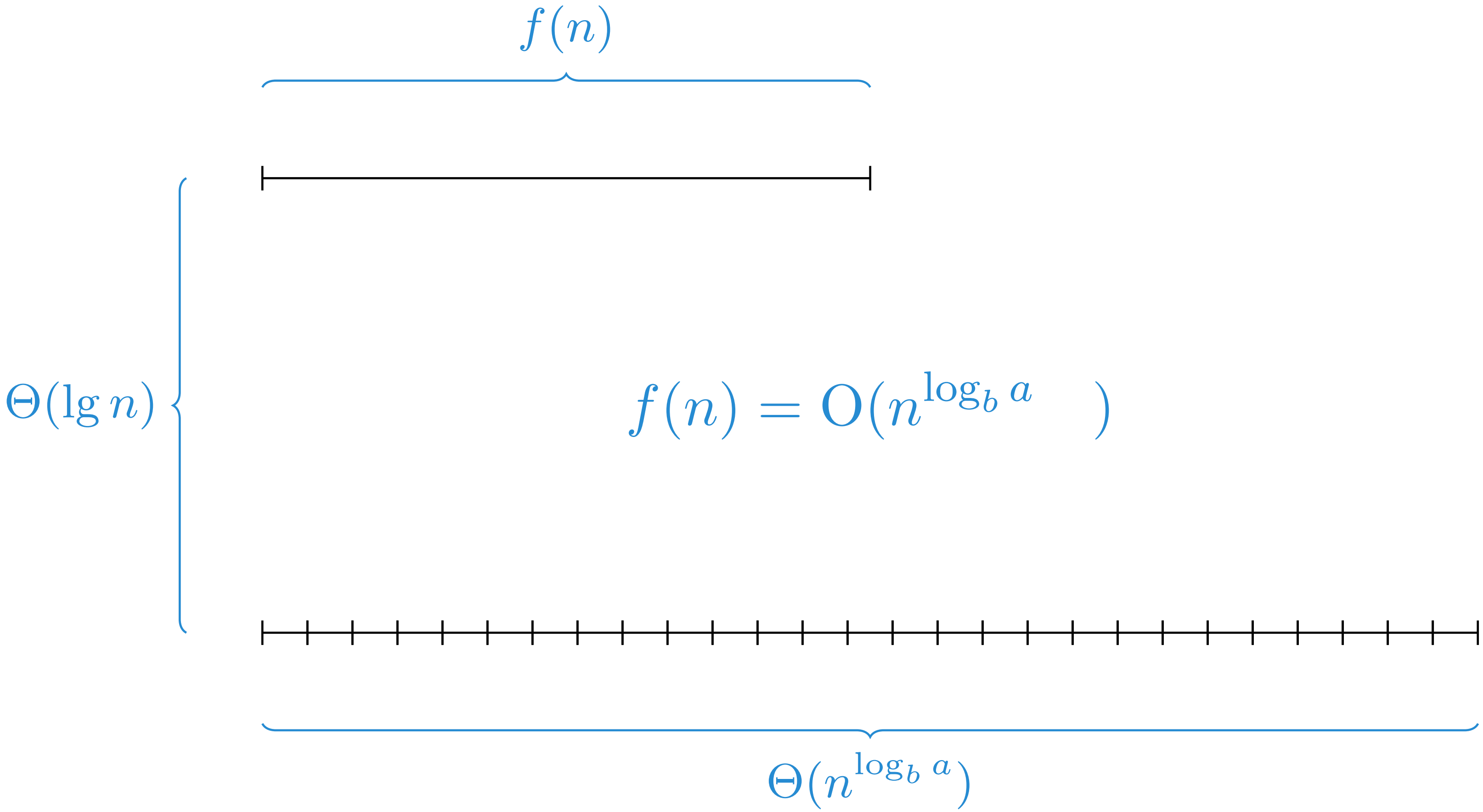
$f(n)$
$\Theta(\lg n)$
$f(n) = \Theta(n^{\log_b a})$
$\Theta(n^{\log_b a})$



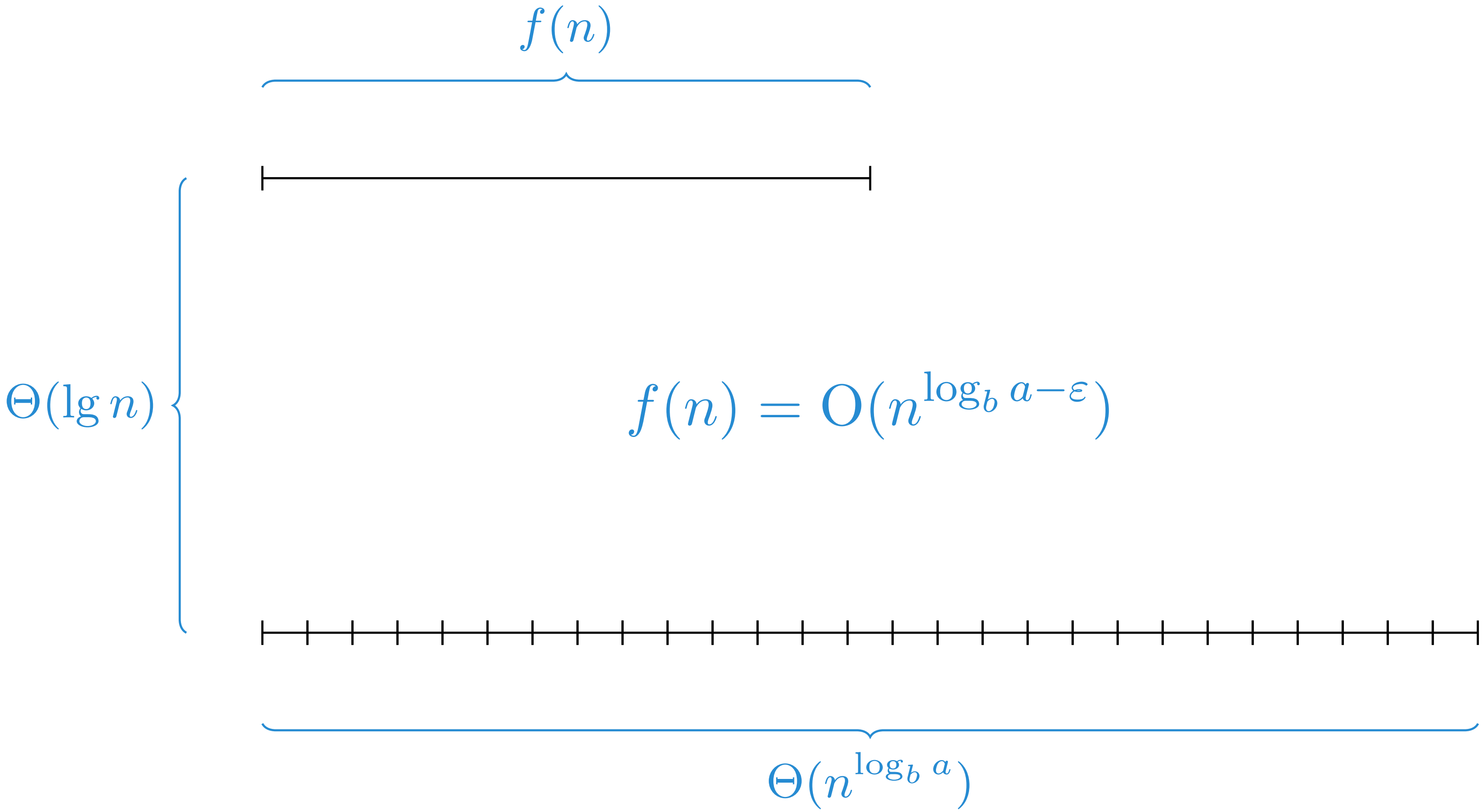
Totalt:



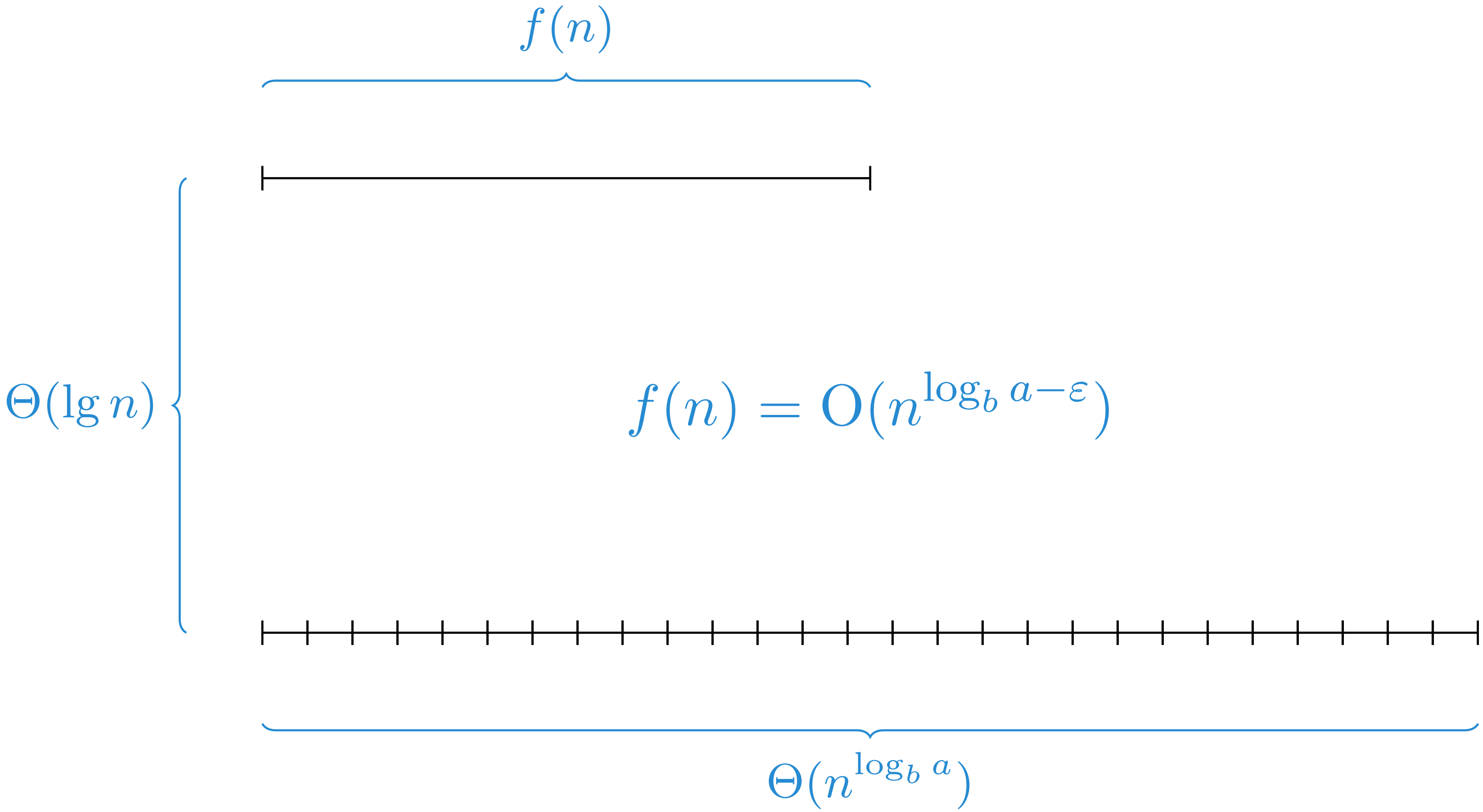
Totalt: $\Theta(n^{\log_b a} \lg n)$



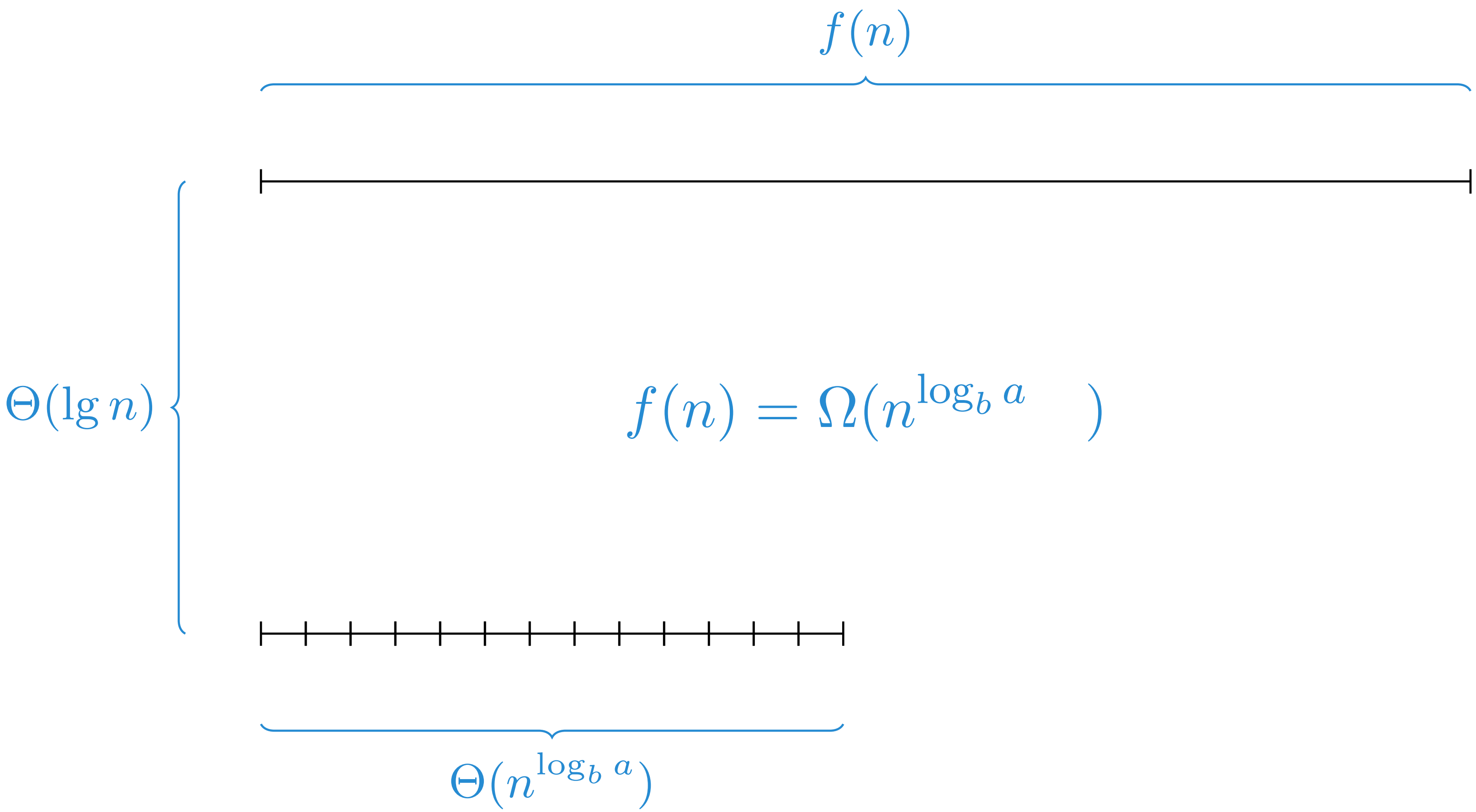
Totalt:



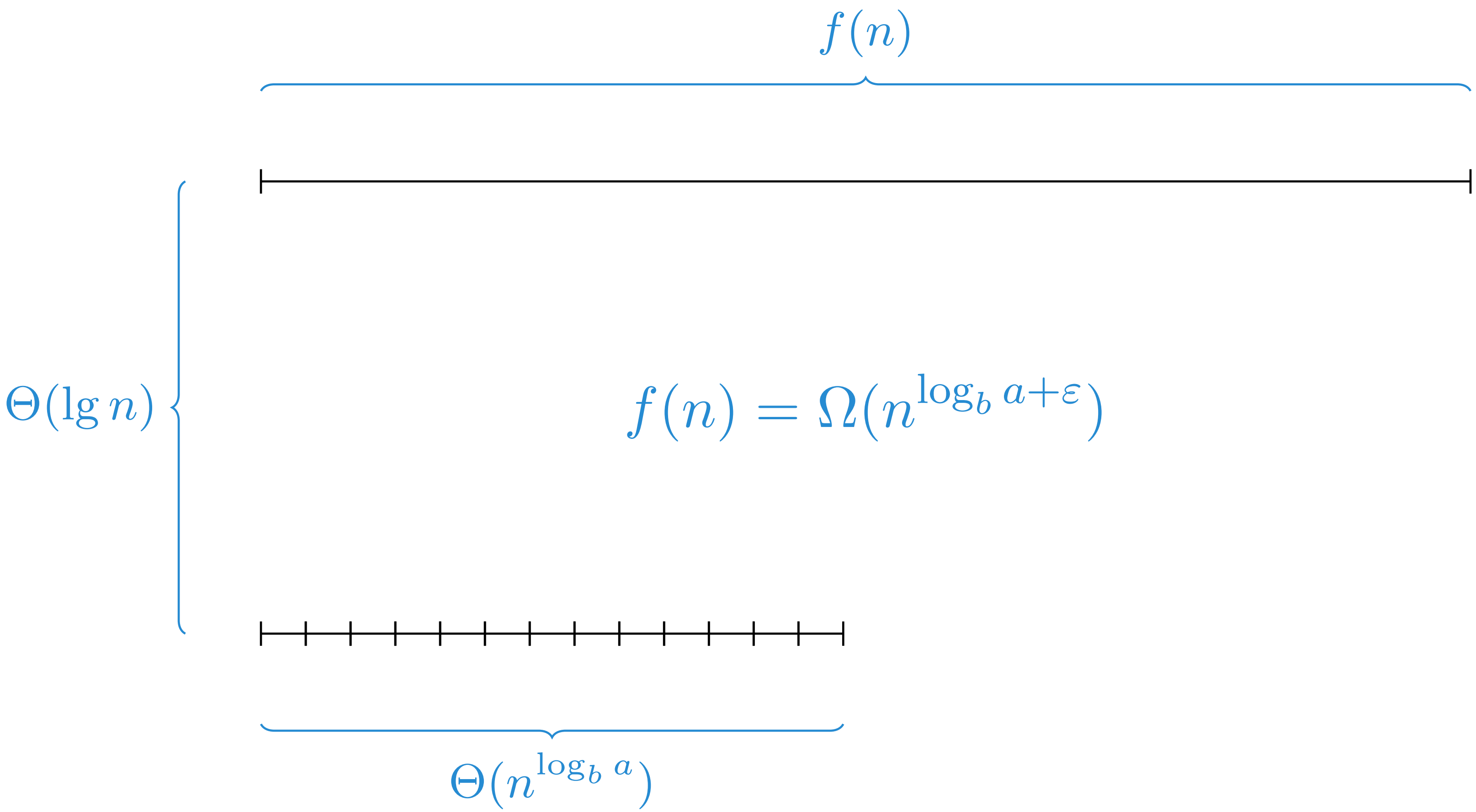
Totalt:



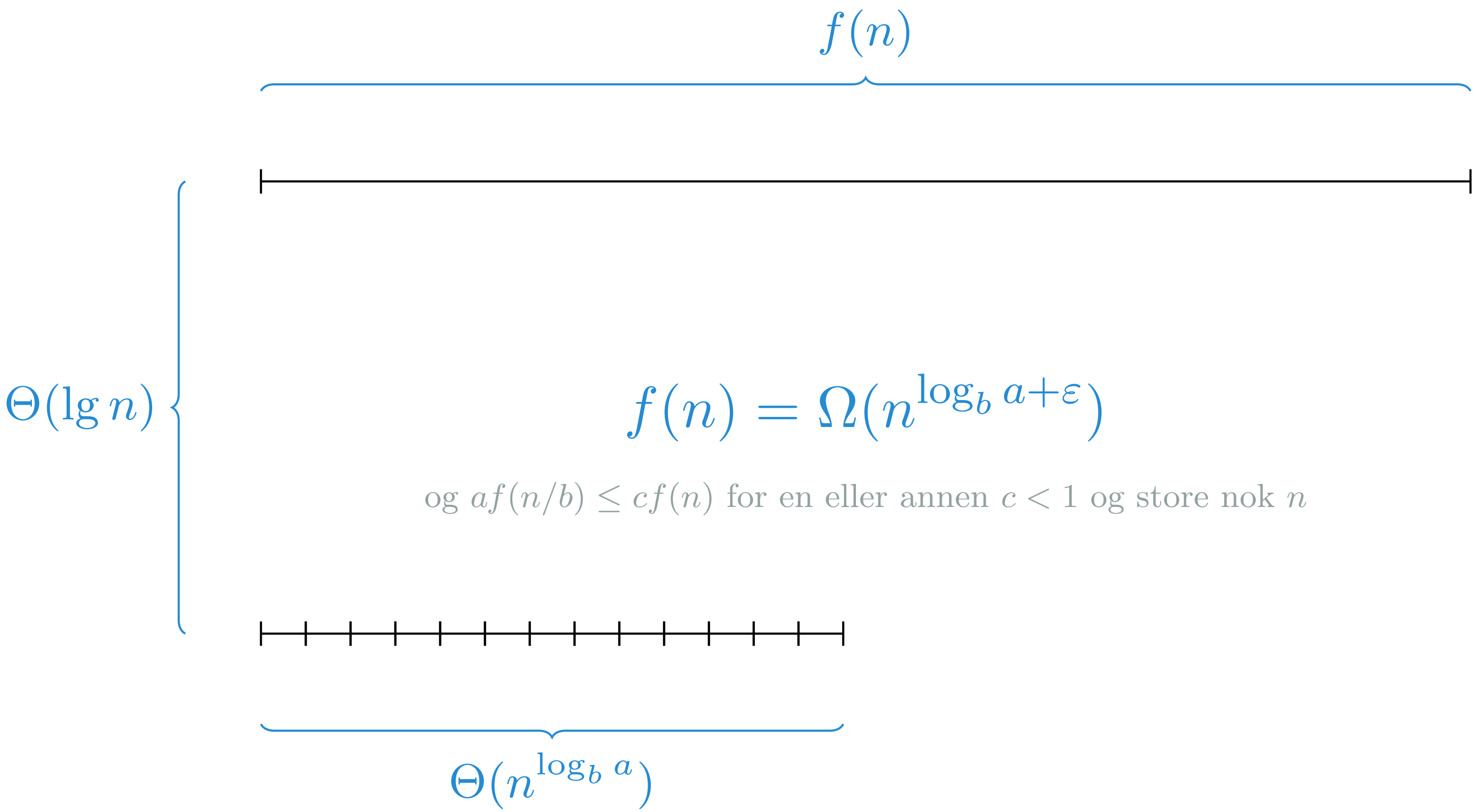
Totalt: $\Theta(n^{\log_b a})$



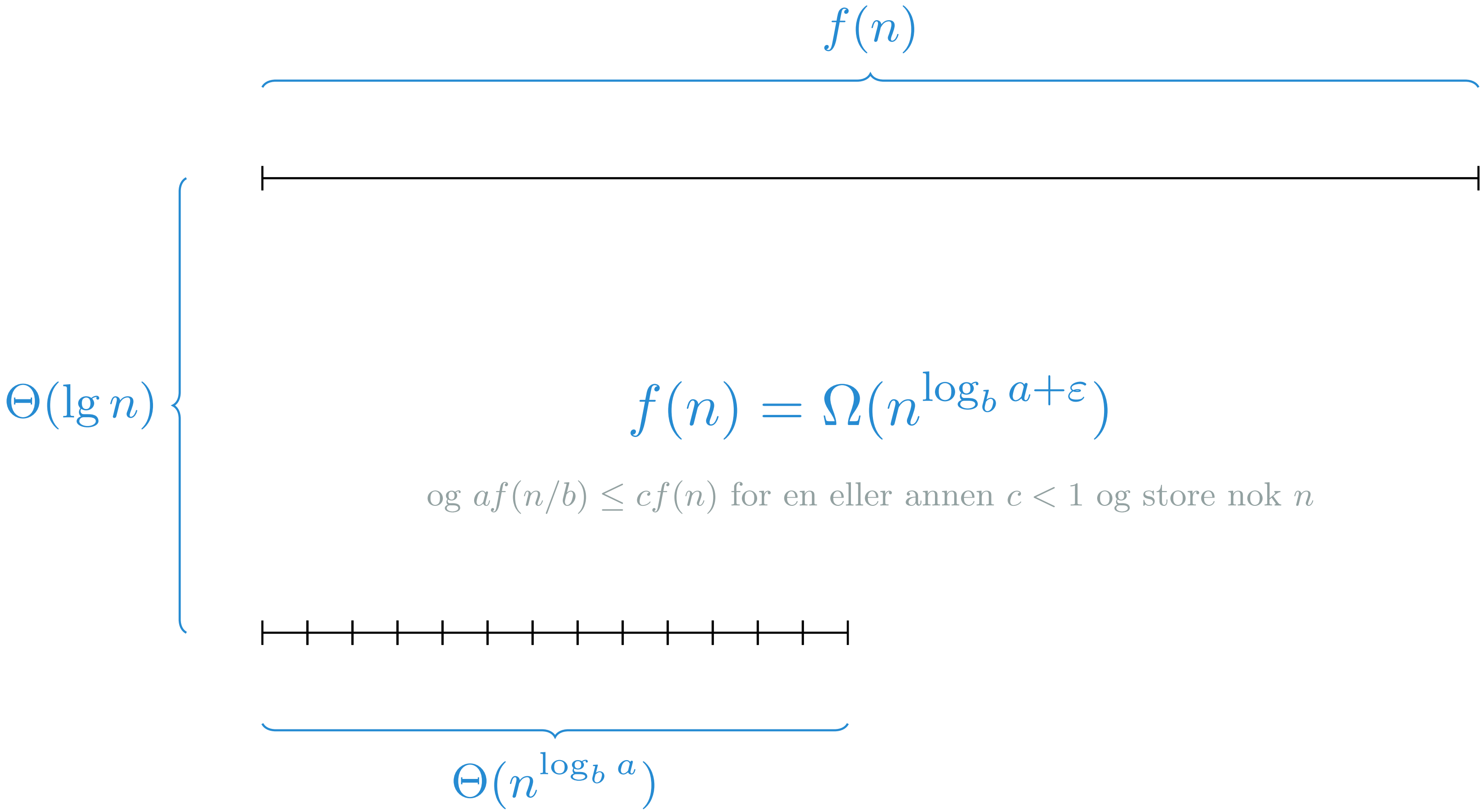
Totalt:



Totalt:


$f(n)$
$\Theta(\lg n)$
$f(n) = \Omega(n^{\log_b a + \varepsilon})$
og $af(n/b) \leq cf(n)$ for en eller annen $c < 1$ og store nok n
$\Theta(n^{\log_b a})$

Totalt:



Totalt: $\Theta(f(n))$

Rekurrenser

En rekurrens er en rekursiv ligning, der en funksjon defineres delvis ved hjelp av seg selv.

$$f(x) = 1 + f(x)$$

$$\boxed{f(x)} = \boxed{1 + f(x)}$$

$$\boxed{f(x)} = \boxed{1 + \boxed{f(x)}}$$

$$\boxed{f(x)} = \boxed{1 + \boxed{1 + f(x)}}$$

$$\boxed{f(x)} = \boxed{1 + \boxed{1 + \boxed{f(x)}}}$$

$$\boxed{f(x)} = \boxed{1 + \boxed{1 + \boxed{1 + f(x)}}}$$

$$\boxed{f(x)} = \boxed{1 + \boxed{1 + \boxed{1 + \boxed{f(x)}}}}$$

$$\boxed{f(x)} = \boxed{1 + \boxed{1 + \boxed{1 + \boxed{\cdots}}}}$$

Dette konvergerer ikke ... rekursjonen terminerer aldri. Vi trenger et grunntilfelle!

$$f(x) = 1 + f(x-1)$$

$$\boxed{f(x)} = \boxed{1 + f(x-1)}$$

$$\boxed{f(x)} = \boxed{1 + \boxed{f(x-1)}}$$

$$\boxed{f(x)} = \boxed{1 + \boxed{1 + f(x - 2)}}$$

$$\boxed{f(x)} = \boxed{1 + \boxed{1 + \boxed{f(x-2)}}}$$

$$\boxed{f(x)} = \boxed{1 + \boxed{1 + \boxed{1 + f(x-3)}}}$$

$$\boxed{f(x)} = \boxed{1 + \boxed{1 + \boxed{1 + \cdots + \boxed{f(1)}}}}$$

$$\boxed{f(x)} = \boxed{1 + \boxed{1 + \boxed{1 + \cdots + \boxed{1}}}}$$

$$f(n) = n + f(n-1)$$

$$f(n) = n + (n-1) + f(n-1)$$

$$f(n) = n + (n-1) + (n-2) + f(n-2)$$

$$f(n) = n + (n-1) + (n-2) + \cdots + f(1)$$

$$f(n) = n + (n-1) + (n-2) + \cdots + 1$$

Gjentatt innsetting

Ekvivalent med rekursjonstrærne i boka

$$T(n) = 1 + \boxed{T(n-1)}$$

$$i = 0$$

$$\boxed{T(1)} \to 1$$

$$\boxed{T(n-i)} \to 1 + \boxed{T(n-i-1)}$$

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

$$i = 1$$

$$\boxed{T(1)} \rightarrow 1$$

$$\boxed{T(n-i)} \rightarrow 1 + \boxed{T(n-i-1)}$$

$$T(n) = 1 + 1 + 1 + \boxed{T(n-3)}$$

$$i = 2$$

$$\boxed{T(1)} \to 1$$

$$\boxed{T(n-i)} \to 1 + \boxed{T(n-i-1)}$$

$$T(n) = 1 + 1 + 1 + 1 + \cdots + \boxed{T(1)}$$

$$i = n - 2$$

$$\boxed{T(1)} \rightarrow 1$$

$$\boxed{T(n-i)} \rightarrow 1 + \boxed{T(n-i-1)}$$

Lineært!

$$T(n) = 1 + 1 + 1 + 1 + \cdots + 1$$

$$i = n - 2$$

$$\boxed{T(1)} \to 1$$

$$\boxed{T(n-i)} \to 1 + \boxed{T(n-i-1)}$$

$$T(n) = 1 + \boxed{T(n/2^1)}$$

$$i = 0$$

$$\boxed{T(1)} \to 1$$

$$\boxed{T(n/2^i)} \to 1 + \boxed{T(n/2^{i+1})}$$

$$T(n) = 1 + 1 + \boxed{T(n/2^2)}$$

$$i = 1$$

$$\boxed{T(1)} \to 1$$

$$\boxed{T(n/2^i)} \to 1 + \boxed{T(n/2^{i+1})}$$

$$T(n) = 1 + 1 + 1 + \boxed{T(n/2^3)}$$

$$i = 2$$

$$\boxed{T(1)} \to 1$$

$$\boxed{T(n/2^i)} \to 1 + \boxed{T(n/2^{i+1})}$$

$$T(n) = 1 + 1 + 1 + 1 + \cdots + \boxed{T(1)}$$

$$i = \lg n - 1$$

$$\boxed{T(1)} \to 1$$

$$\boxed{T(n/2^i)} \to 1 + \boxed{T(n/2^{i+1})}$$

Logaritmisk!

$$T(n) = 1 + 1 + 1 + 1 + \cdots + 1$$

$$i = \lg n - 1$$

$$\boxed{T(1)} \to 1$$

$$\boxed{T(n/2^i)} \to 1 + \boxed{T(n/2^{i+1})}$$

$$T(n) = n + \boxed{T(n/2^1)}$$

$$i = 1$$

$$\boxed{T(1)} \to 1$$

$$\boxed{T(n/2^i)} \to n/2^i + \boxed{T(n/2^{i+1})}$$

$$T(n) = n + n/2^1 + \boxed{T(n/2^2)}$$

$$i = 2$$

$$\boxed{T(1)} \rightarrow 1$$

$$\boxed{T(n/2^i)} \rightarrow n/2^i + \boxed{T(n/2^{i+1})}$$

$$T(n) = n + n/2^1 + n/2^2 + \boxed{T(n/2^3)}$$

$$i = 3$$

$$\boxed{T(1)} \to 1$$

$$\boxed{T(n/2^i)} \to n/2^i + \boxed{T(n/2^{i+1})}$$

Lineært!

$$T(n) = n + n/2^1 + n/2^2 + n/2^3 + \cdots + 1$$

$$i = \lg n$$

$$\boxed{T(1)} \to 1$$

$$\boxed{T(n/2^i)} \to n/2^i + \boxed{T(n/2^{i+1})}$$

$$T(n) = n + \boxed{T(n-1)}$$

$$i = 0$$

$$\boxed{T(1)} \rightarrow 1$$

$$\boxed{T(n-i)} \rightarrow (n-i) + \boxed{T(n-i-1)}$$

$$T(n) = n + (n-2) + \boxed{T(n-2)}$$

$$i = 1$$

$$\boxed{T(1)} \rightarrow 1$$

$$\boxed{T(n-i)} \rightarrow (n-i) + \boxed{T(n-i-1)}$$

$$T(n) = n + (n-2) + (n-3) + \boxed{T(n-3)}$$

$$i = 2$$

$$\boxed{T(1)} \rightarrow 1$$

$$\boxed{T(n-i)} \rightarrow (n-i) + \boxed{T(n-i-1)}$$

$$T(n) = n + (n-2) + (n-3) + (n-4) + \cdots + \boxed{T(1)}$$

$$i = n - 2$$

$$\boxed{T(1)} \to 1$$

$$\boxed{T(n-i)} \to (n-i) + \boxed{T(n-i-1)}$$

Kvadratisk!

$$T(n) = n + (n-2) + (n-3) + (n-4) + \cdots + 1$$

$$i = n - 2$$

$$\boxed{T(1)} \rightarrow 1$$

$$\boxed{T(n-i)} \rightarrow (n-i) + \boxed{T(n-i-1)}$$

Rekurrenstræer

$T(4)$

$T(4)$

$$T(1) = \boxed{1}$$

$$T(n) = \boxed{n + 2 \cdot T(n/2)}$$

$T(4)$

$T(4) =$

$T(1) = \boxed{1}$

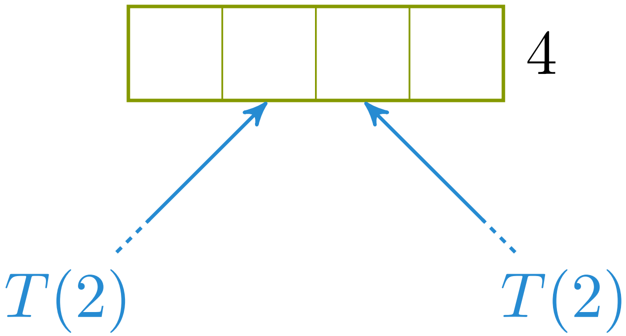
$T(n) = \boxed{n + 2 \cdot T(n/2)}$

$T(4)$

$T(4) =$

$T(1) = \boxed{1}$

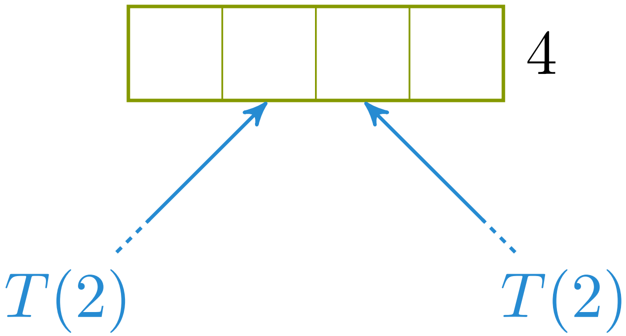
$T(n) = \boxed{n + 2 \cdot T(n/2)}$



$$T(4) = \boxed{4 + 2 \cdot T(4/2)}$$

$$T(1) = \boxed{1}$$

$$T(n) = \boxed{n + 2 \cdot T(n/2)}$$

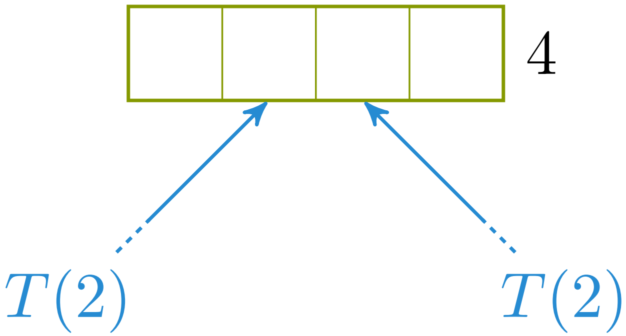


$$T(4) = \boxed{4 + 2 \cdot T(4/2)}$$

$$=$$

$$T(1) = \boxed{1}$$

$$T(n) = \boxed{n + 2 \cdot T(n/2)}$$

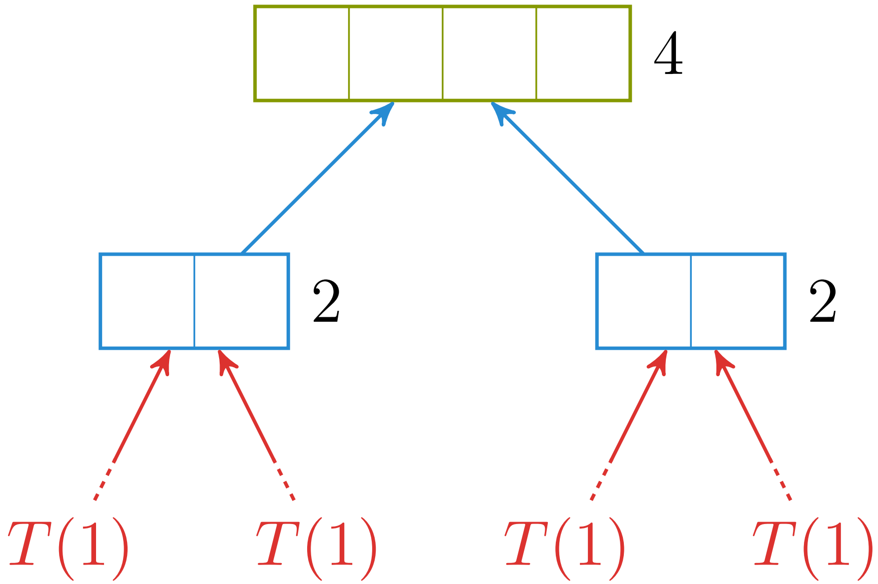


$$T(4) = \boxed{4 + 2 \cdot T(4/2)}$$

$$=$$

$$T(1) = \boxed{1}$$

$$T(n) = \boxed{n + 2 \cdot T(n/2)}$$

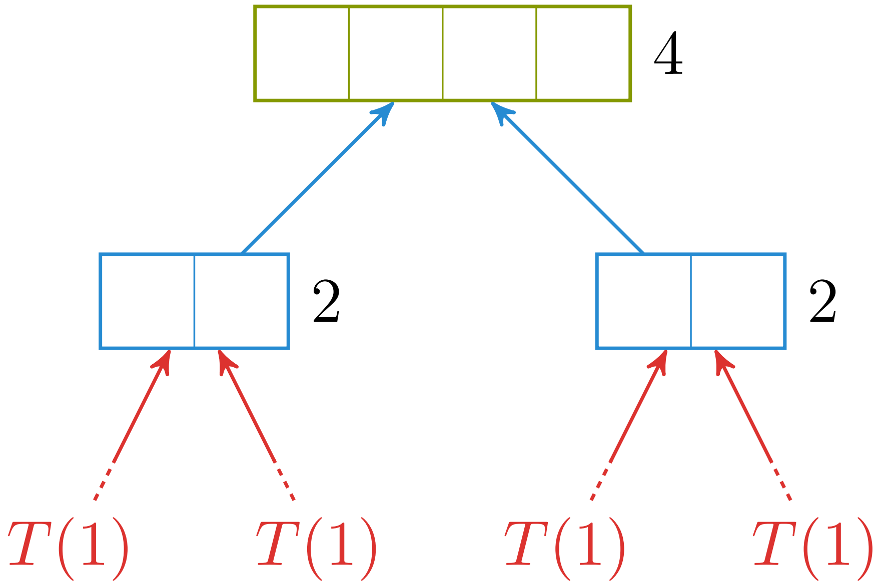


$$T(4) = \boxed{4 + 2 \cdot T(4/2)}$$

$$= \boxed{4 + 2 \cdot \boxed{4/2 + 2 \cdot T(4/4)}}$$

$$T(1) = \boxed{1}$$

$$T(n) = \boxed{n + 2 \cdot T(n/2)}$$



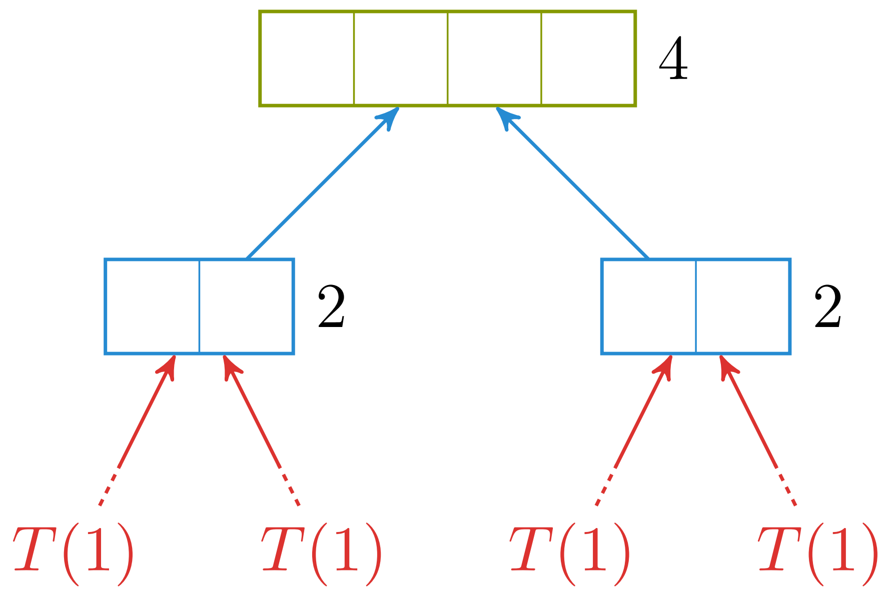
$$T(4) = \boxed{4 + 2 \cdot T(4/2)}$$

$$= \boxed{4 + 2 \cdot \boxed{4/2 + 2 \cdot T(4/4)}}$$

$$=$$

$$T(1) = \boxed{1}$$

$$T(n) = \boxed{n + 2 \cdot T(n/2)}$$



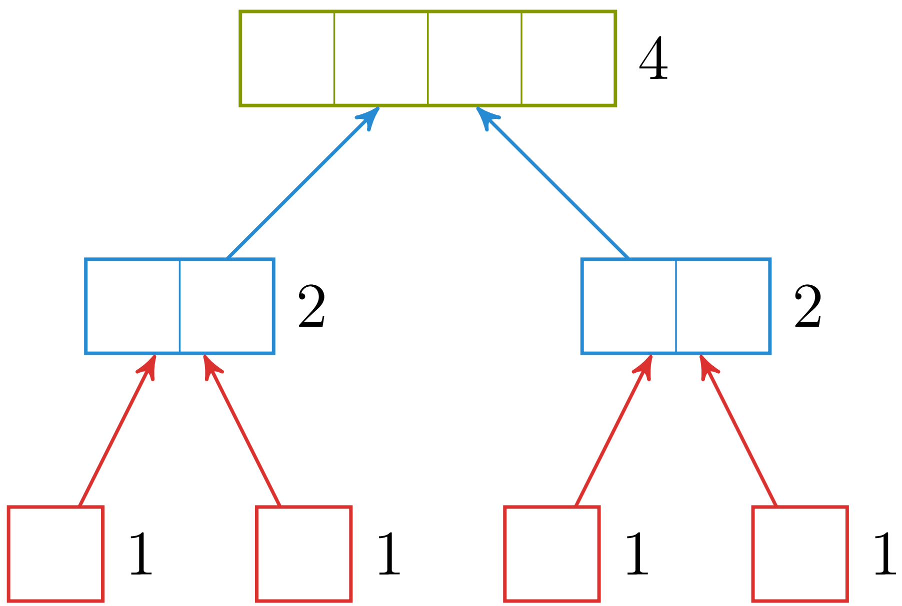
$$T(4) = \boxed{4 + 2 \cdot T(4/2)}$$

$$= \boxed{4 + 2 \cdot \boxed{4/2 + 2 \cdot T(4/4)}}$$

$$=$$

$$T(1) = \boxed{1}$$

$$T(n) = \boxed{n + 2 \cdot T(n/2)}$$



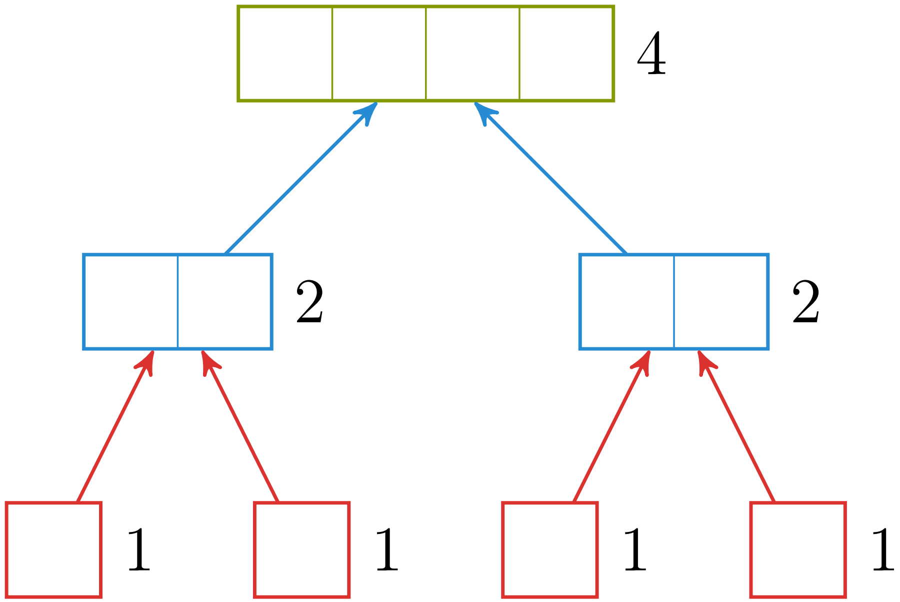
$$T(4) = \boxed{4 + 2 \cdot T(4/2)}$$

$$= \boxed{4 + 2 \cdot \boxed{4/2 + 2 \cdot T(4/4)}}$$

$$= \boxed{4 + 2 \cdot \boxed{4/2 + 2 \cdot \boxed{1}}}$$

$$T(1) = \boxed{1}$$

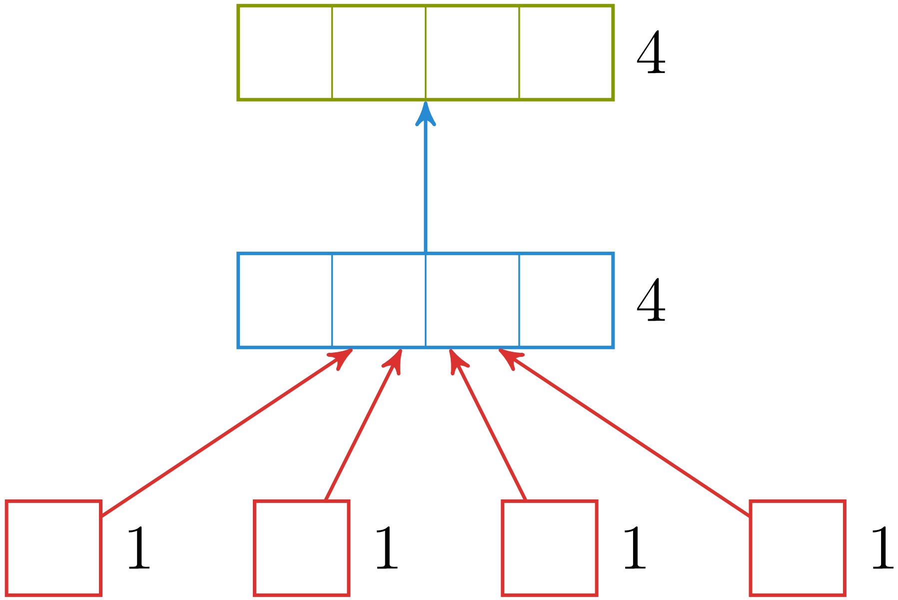
$$T(n) = \boxed{n + 2 \cdot T(n/2)}$$



$$T(4) = \boxed{4 + 2 \cdot T(4/2)}$$

$$= \boxed{4 + 2 \cdot \boxed{4/2 + 2 \cdot T(4/4)}}$$

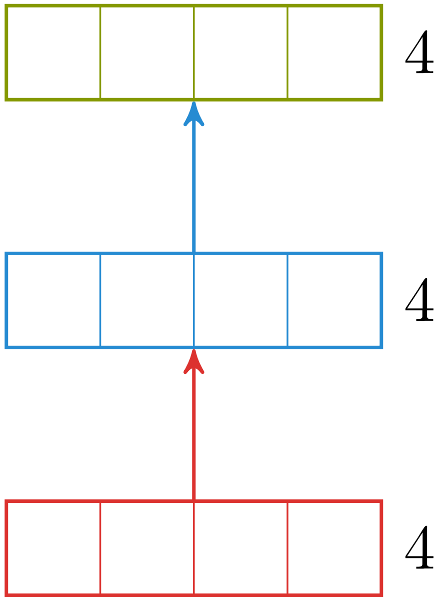
$$= \boxed{4 + 2 \cdot \boxed{4/2 + 2 \cdot \boxed{1}}}$$



$$T(4) = \boxed{4 + 2 \cdot T(4/2)}$$

$$= \boxed{4 + \boxed{4 + 4 \cdot T(4/4)}}$$

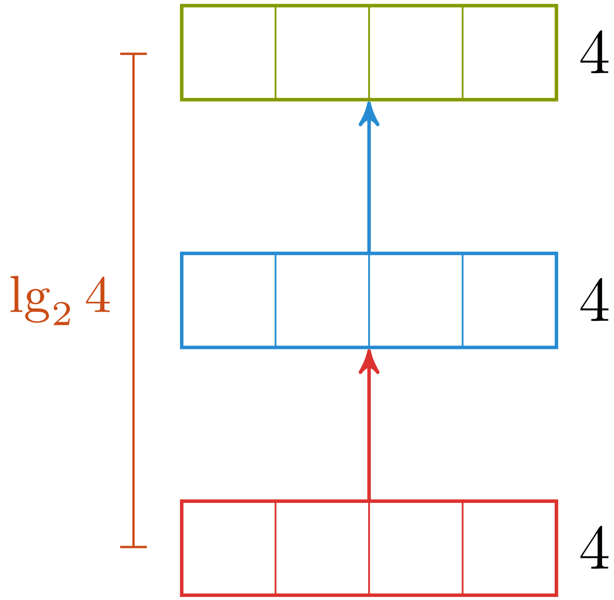
$$= \boxed{4 + \boxed{4 + 4 \cdot \boxed{1}}}$$



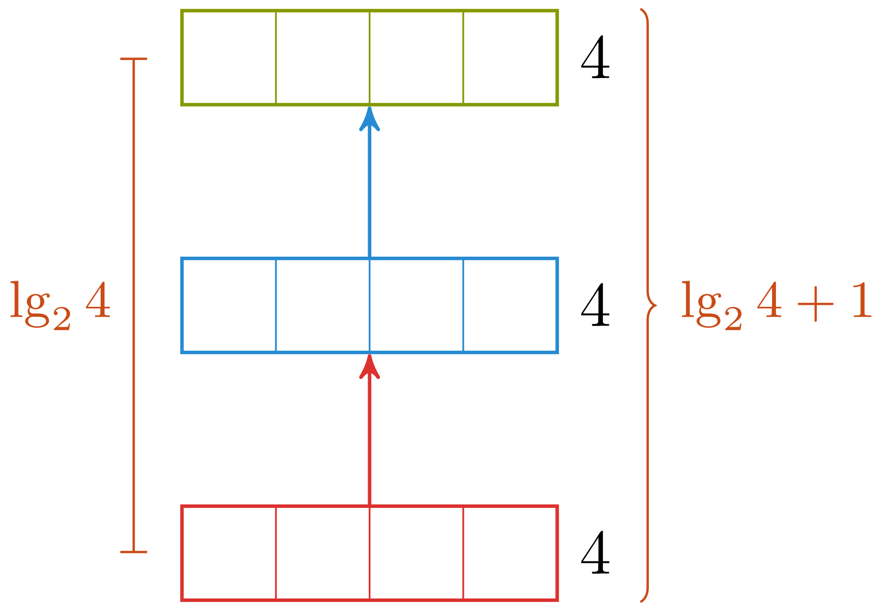
$$T(4) = \boxed{4 + 2 \cdot T(4/2)}$$

$$= \boxed{4 + \boxed{4 + 4 \cdot T(4/4)}}$$

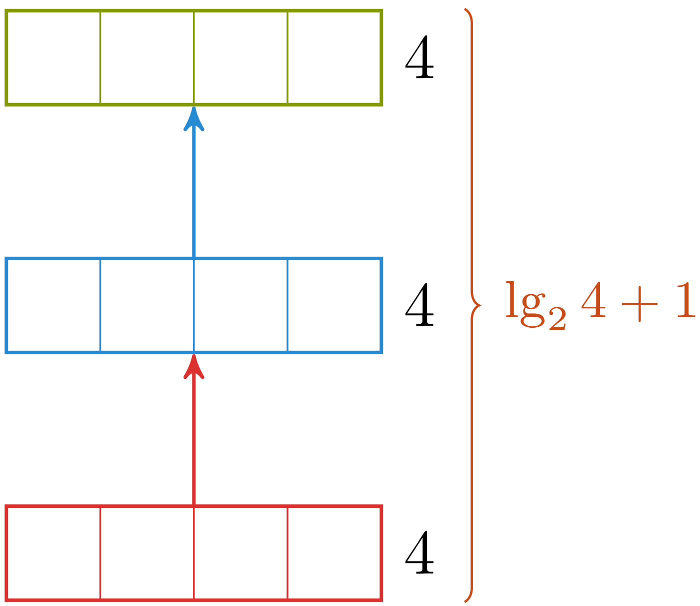
$$= \boxed{4 + \boxed{4 + \boxed{4}}}$$



$$
\begin{aligned}
T(4) &= \boxed{4 + 2 \cdot T(4/2)} \\
&= \boxed{4 + \boxed{4 + 4 \cdot T(4/4)}} \\
&= \boxed{4 + \boxed{4 + \boxed{4}}}
\end{aligned}
$$



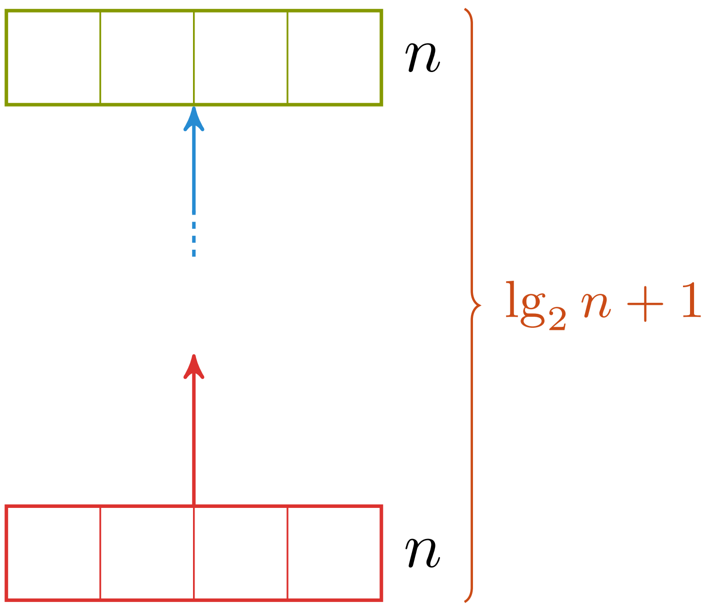
$$
\begin{aligned}
T(4) &= \boxed{4 + 2 \cdot T(4/2)} \\
&= \boxed{4 + \boxed{4 + 4 \cdot T(4/4)}} \\
&= \boxed{4 + \boxed{4 + \boxed{4}}}
\end{aligned}
$$



$$T(4) = \boxed{4 + 2 \cdot T(4/2)}$$

$$= \boxed{4 + \boxed{4 + 4 \cdot T(4/4)}}$$

$$= \underbrace{\boxed{4 + \boxed{4 + \boxed{4}}}}_{\lg_2 4 + 1}$$



$$T(n) = \boxed{n + 2 \cdot T(n/2)}$$

$$\vdots$$

$$= \underbrace{\boxed{n + \boxed{\cdots + \boxed{n}}}}_{\lg_2 n + 1}$$

Substitusjon

(Dvs. induksjon...)

- **Vi tar ofte snarveier**
- **F.eks.: Vi ignorerer avrundingsfeil**
- **Sjekk løsningen med induksjon!**
- **Evt. gjett løsning og sjekk den**

$$T(1) = 1$$

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

$$T(1) = 1$$

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

$n > 1$

$$T(1) = 1$$

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

$$T(n) = \frac{n(n+1)}{2}$$

$T(n)$

$$\frac{n(n+1)}{2}$$

$$T(n) \quad \bigg| \quad \frac{n(n+1)}{2}$$

$$T(1) = 1$$

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

$n = 1$

$$T(n) \quad \Big| \quad \frac{n(n+1)}{2}$$

$T(1) = 1$

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

$n = 1$

$T(n)$ | $\frac{n(n+1)}{2}$

1

$T(1) = 1$

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

$n = 1$

$$T(n) \quad \Bigg| \quad \frac{n(n+1)}{2}$$

$$1 \qquad \frac{1(1+1)}{2} = 1$$

$T(1) = 1$

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

$n > 1$

$$T(n) \quad \Big| \quad \frac{n(n+1)}{2}$$

H: $T(n-1) = \dfrac{(n-1)n}{2}$

$T(1) = 1$

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

$n > 1$

$T(n)$

$$\frac{n(n+1)}{2}$$

$$\frac{n(n+1)}{2}$$

$T(1) = 1$

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

H: $T(n-1) = \dfrac{(n-1)n}{2}$

$n > 1$

$T(n)$ | $\dfrac{n(n+1)}{2}$

$T(n) =$

$$\frac{n(n+1)}{2}$$

H: $T(n-1) = \dfrac{(n-1)n}{2}$

$T(1) = 1$

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

$n > 1$

$T(n)$ $\frac{n(n+1)}{2}$

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

$$\frac{n(n+1)}{2}$$

H: $T(n-1) = \frac{(n-1)n}{2}$

$T(1) = 1$

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

$n > 1$

$T(n)$ | $\dfrac{n(n+1)}{2}$

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

$$\frac{n(n+1)}{2}$$

H: $T(n-1) = \dfrac{(n-1)n}{2}$

$T(1) = 1$

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

$n > 1$

$T(n)$ | $\dfrac{n(n+1)}{2}$

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

$$= \frac{(n-1)n}{2} + n$$

$$\frac{n(n+1)}{2}$$

H: $T(n-1) = \dfrac{(n-1)n}{2}$

$T(1) = 1$

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

$n > 1$

$T(n)$

$$\frac{n(n+1)}{2}$$

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

$$= \frac{(n-1)n}{2} + n$$

$$= \frac{n^2 - n + 2n}{2}$$

$$\frac{n(n+1)}{2}$$

H: $T(n-1) = \frac{(n-1)n}{2}$

$T(1) = 1$

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

$n > 1$

$T(n)$ | $\frac{n(n+1)}{2}$

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

$$= \frac{(n-1)n}{2} + n$$

$$= \frac{n^2 - n + 2n}{2}$$

$$= \frac{n(n+1)}{2}$$

$$\frac{n(n+1)}{2}$$

H: $T(n-1) = \frac{(n-1)n}{2}$

$T(1) = 1$

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

Masterteoremet

$$T(n) = aT(n/b) + f(n)$$

$$T(n) = aT(n/b) + f(n)$$

$$f(n) = \mathrm{O}(n^{\log_b a - \varepsilon})$$

$$T(n) = aT(n/b) + f(n)$$

$$f(n) = \mathrm{O}(n^{\log_b a - \varepsilon}) \implies T(n) = \Theta(n^{\log_b a})$$

$$T(n) = aT(n/b) + f(n)$$

$$f(n) = \mathrm{O}(n^{\log_b a - \varepsilon}) \implies T(n) = \Theta(n^{\log_b a})$$

$$f(n) = \Theta(n^{\log_b a})$$

$$T(n) = aT(n/b) + f(n)$$

$$f(n) = \mathrm{O}(n^{\log_b a - \varepsilon}) \implies T(n) = \Theta(n^{\log_b a})$$

$$f(n) = \Theta(n^{\log_b a}) \implies T(n) = \Theta(n^{\log_b a} \lg n)$$

$$T(n) = aT(n/b) + f(n)$$

$$f(n) = \mathrm{O}(n^{\log_b a - \varepsilon}) \implies T(n) = \Theta(n^{\log_b a})$$

$$f(n) = \Theta(n^{\log_b a}) \implies T(n) = \Theta(n^{\log_b a} \lg n)$$

$$f(n) = \Omega(n^{\log_b a + \varepsilon})$$

$$T(n) = aT(n/b) + f(n)$$

$$f(n) = \mathrm{O}(n^{\log_b a - \varepsilon}) \implies T(n) = \Theta(n^{\log_b a})$$

$$f(n) = \Theta(n^{\log_b a}) \implies T(n) = \Theta(n^{\log_b a} \lg n)$$

$$f(n) = \Omega(n^{\log_b a + \varepsilon}) \implies T(n) = \Theta(f(n))$$

$$T(n) = aT(n/b) + f(n)$$

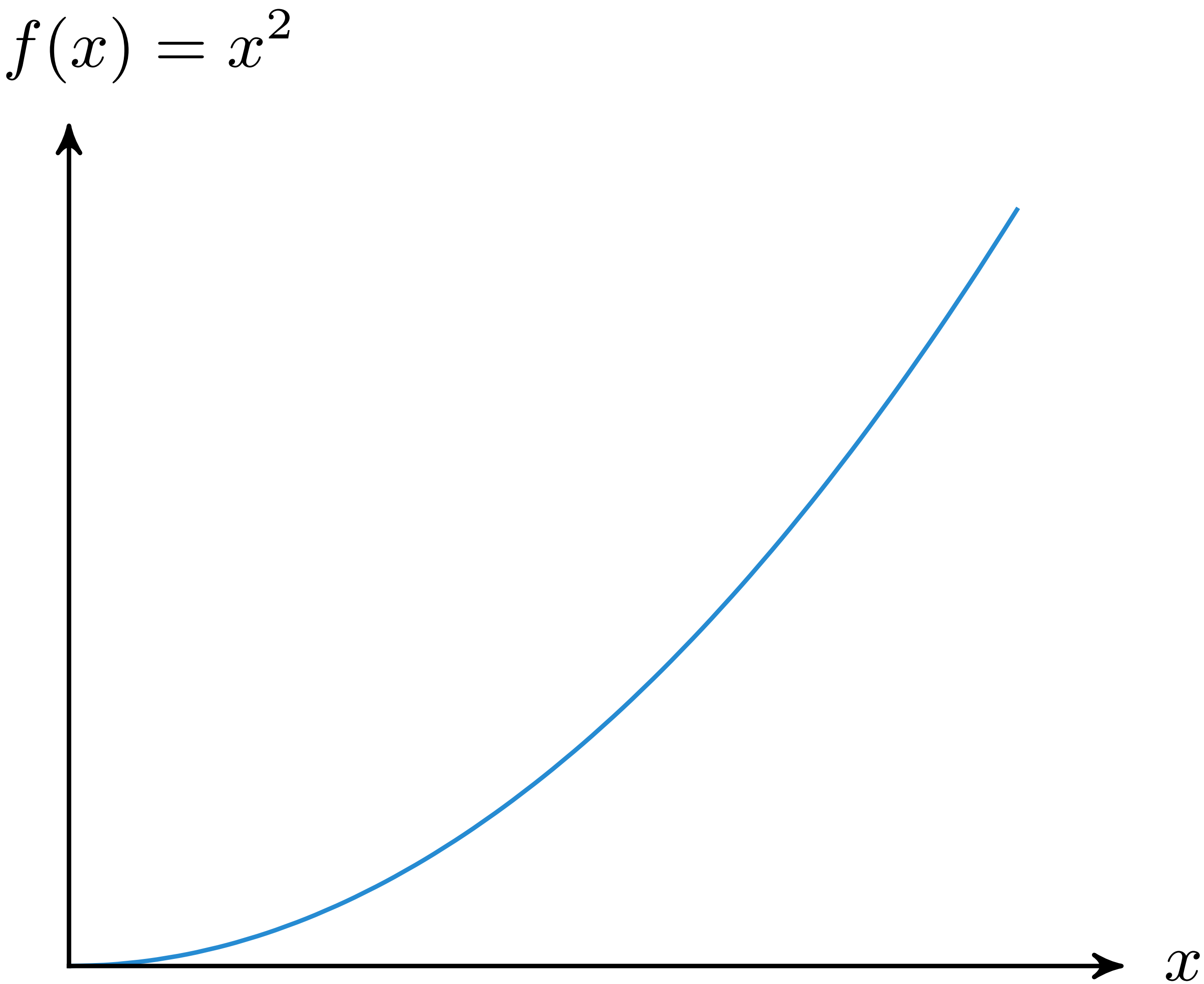
$$f(n) = O(n^{\log_b a - \varepsilon}) \implies T(n) = \Theta(n^{\log_b a})$$

$$f(n) = \Theta(n^{\log_b a}) \implies T(n) = \Theta(n^{\log_b a} \lg n)$$

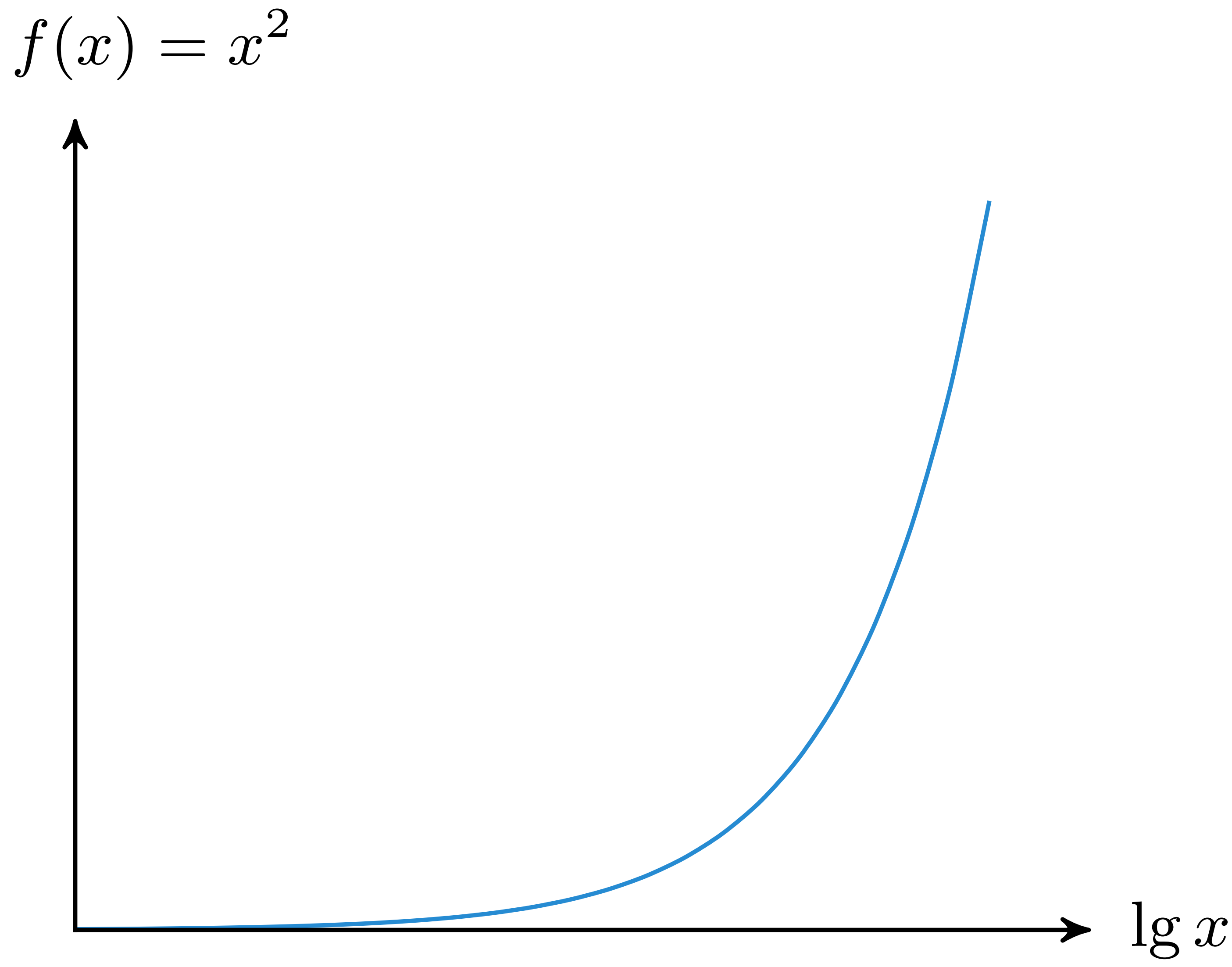
$$f(n) = \Omega(n^{\log_b a + \varepsilon}) \implies T(n) = \Theta(f(n))^*$$

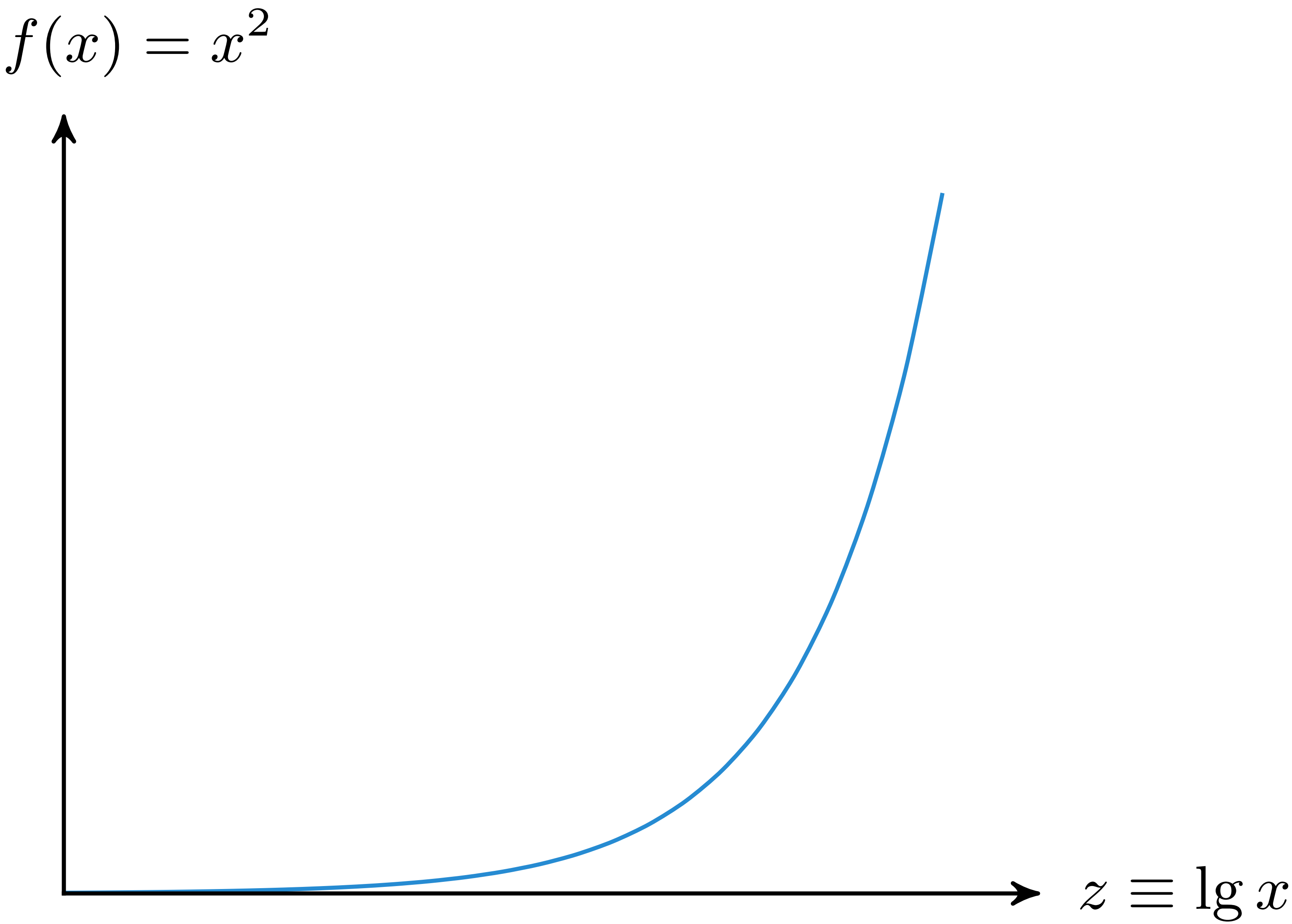
hvis $af(n/b) \leq cf(n)$ for en eller annen $c < 1$ og store nok n

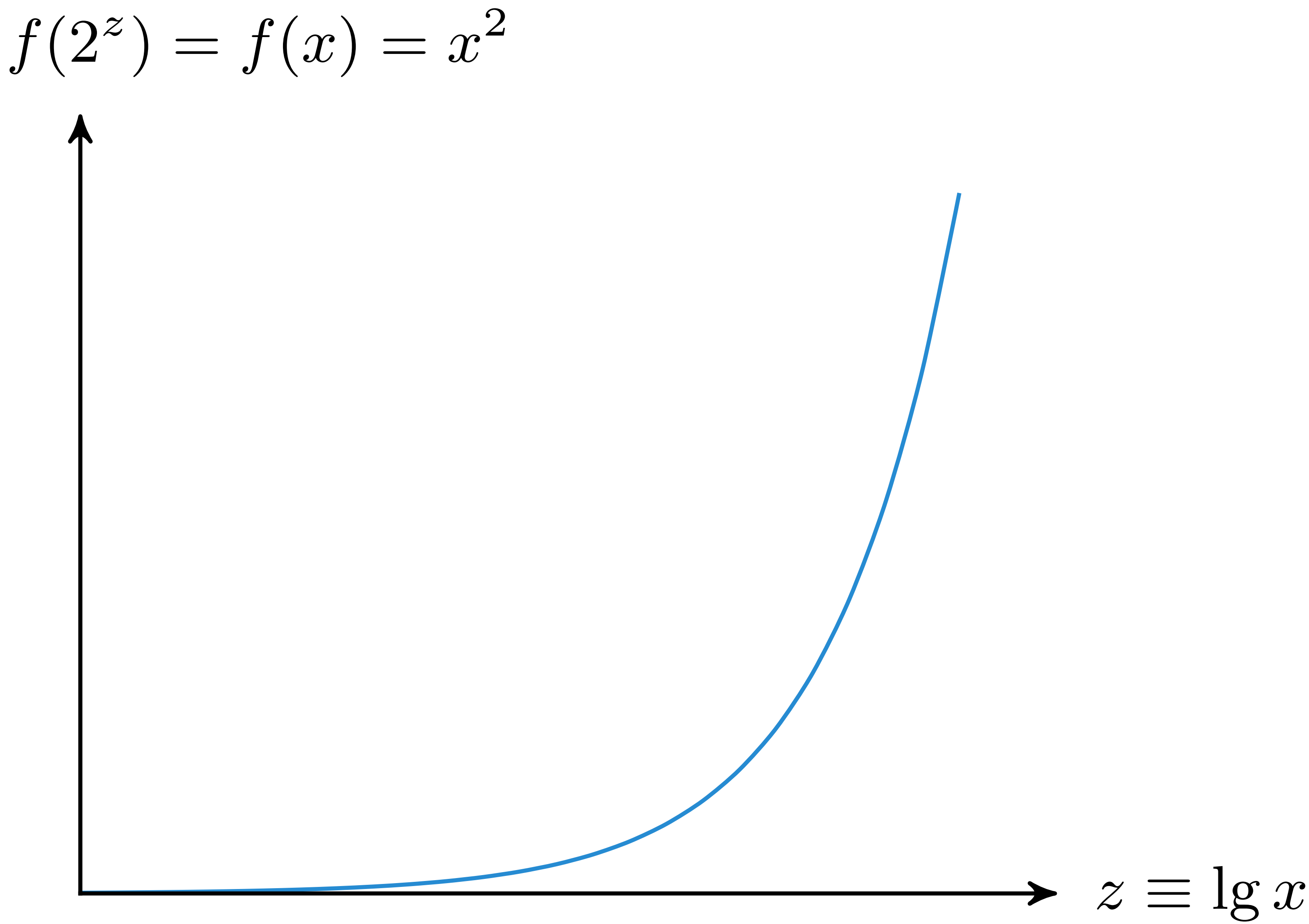
Variabelskifte

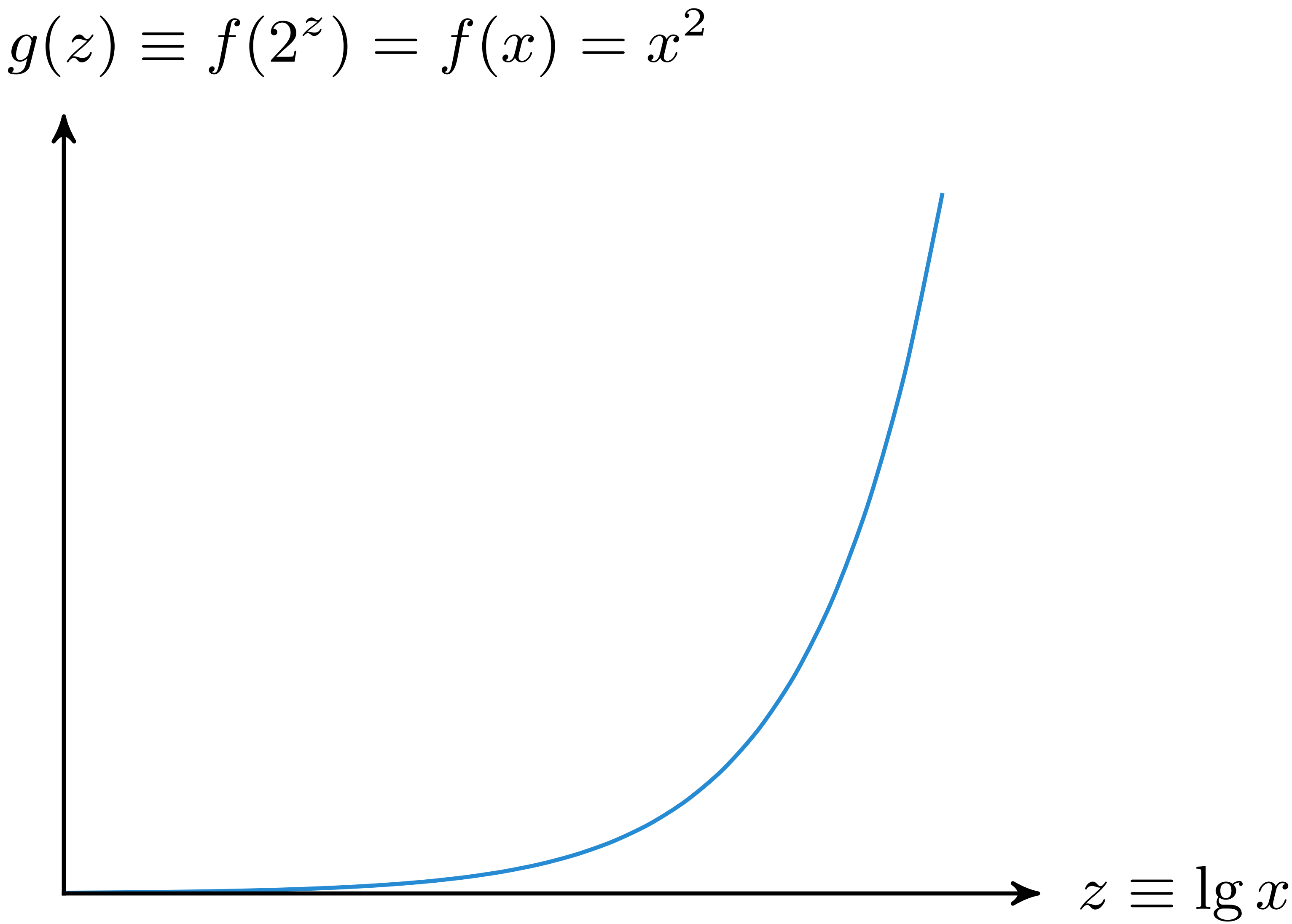


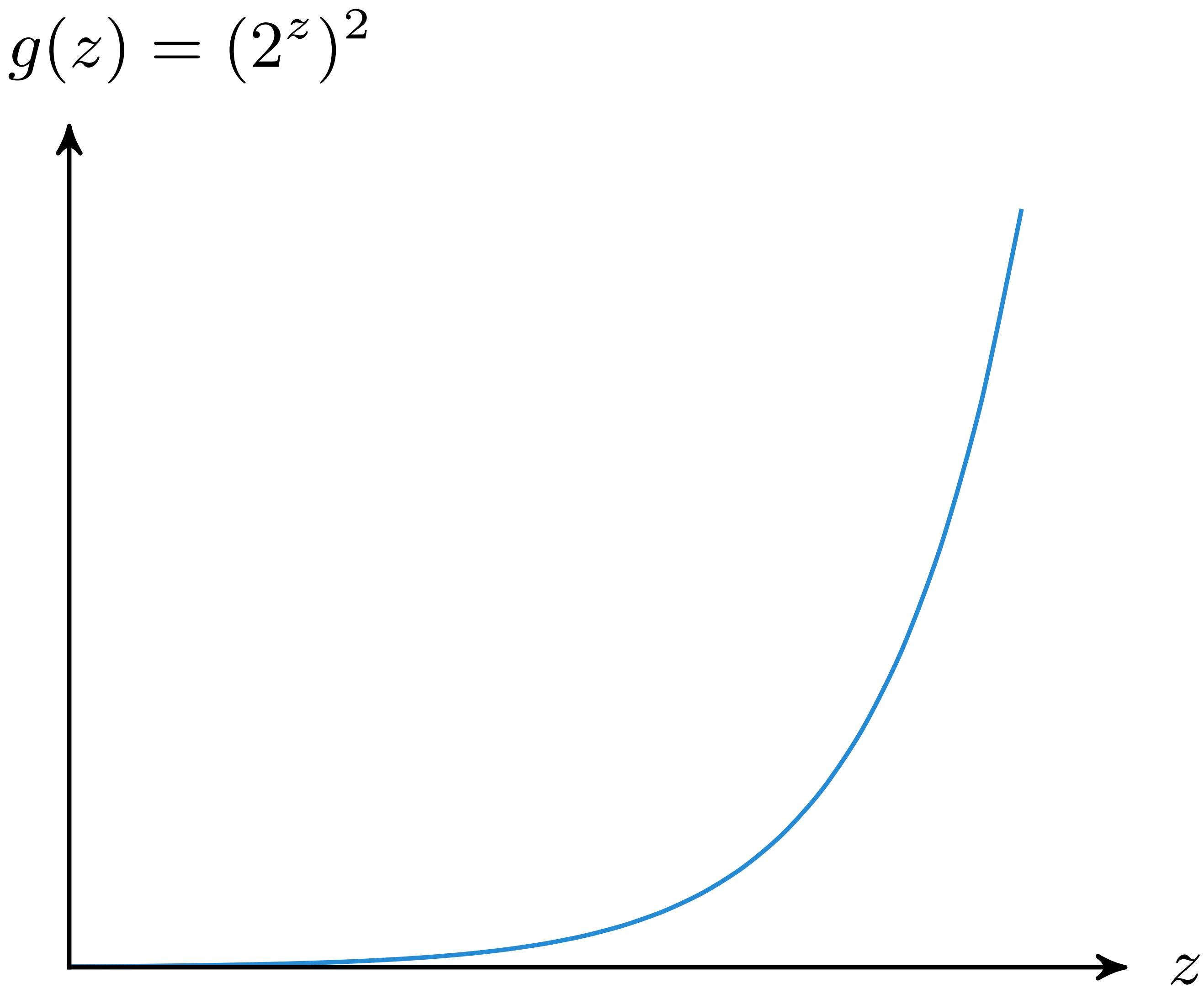
$f(x) = x^2$
x


$f(x) = x^2$
$\lg x$

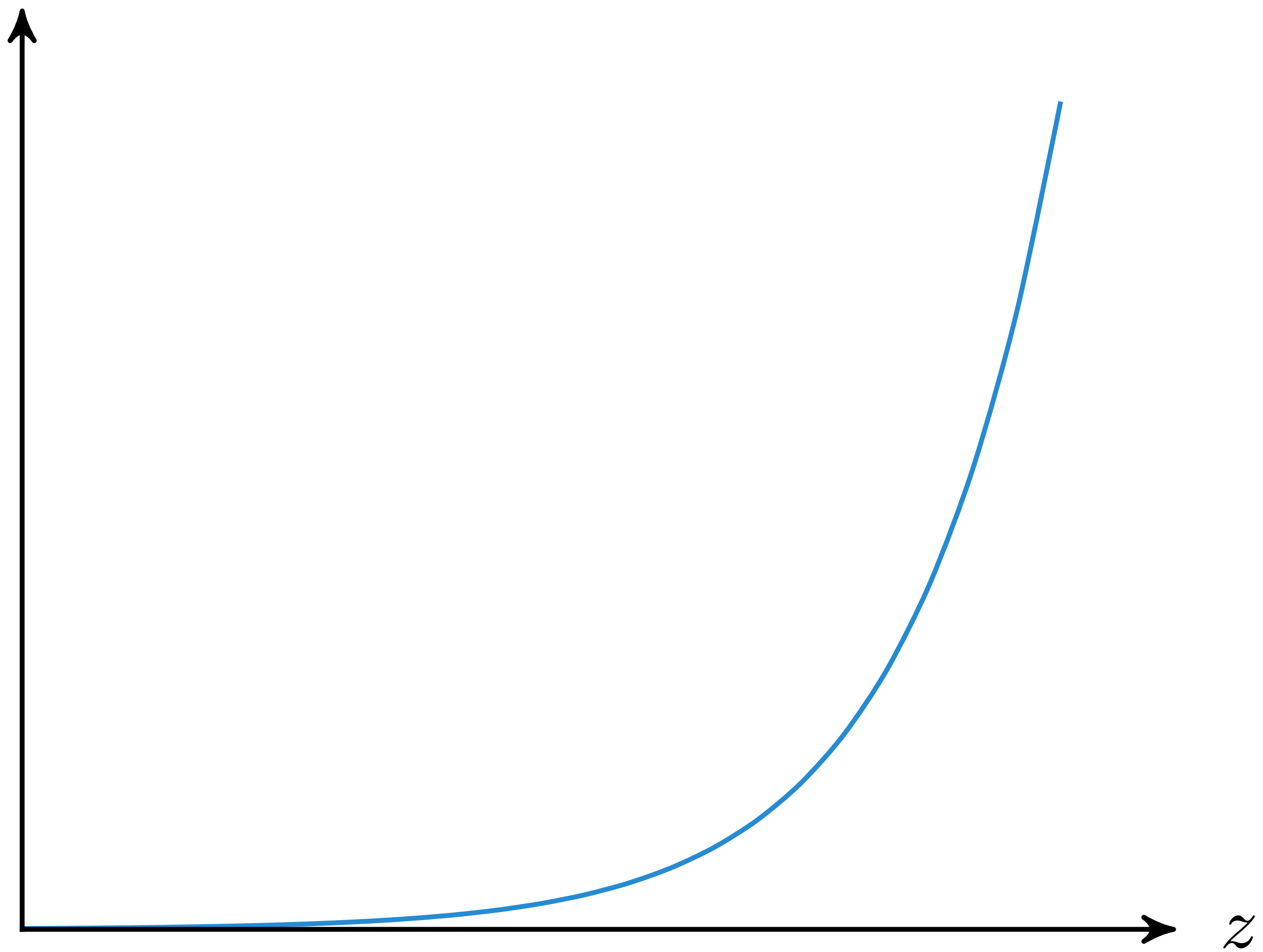

$f(x) = x^2$
$z \equiv \lg x$


$f(2^z) = f(x) = x^2$
$z \equiv \lg x$

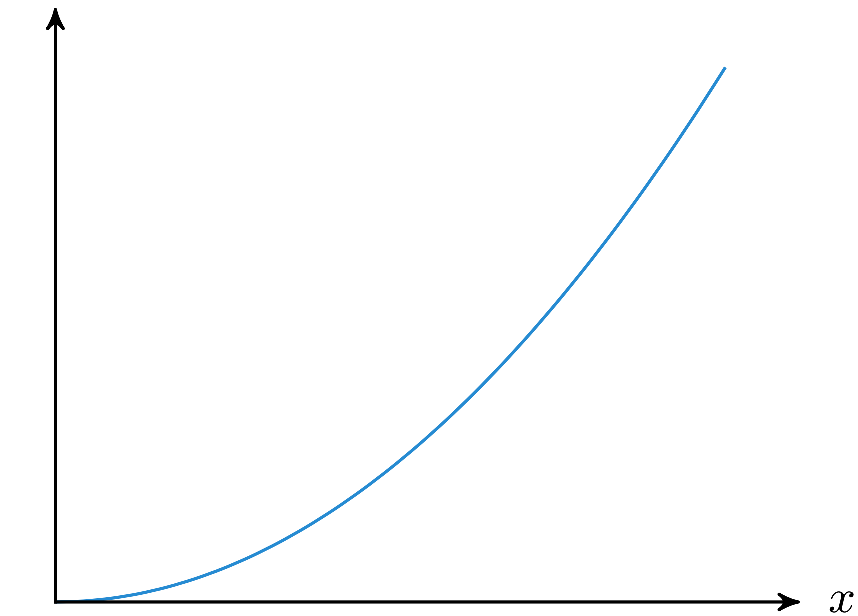

$g(z) \equiv f(2^z) = f(x) = x^2$
$z \equiv \lg x$


$g(z) = (2^z)^2$
z

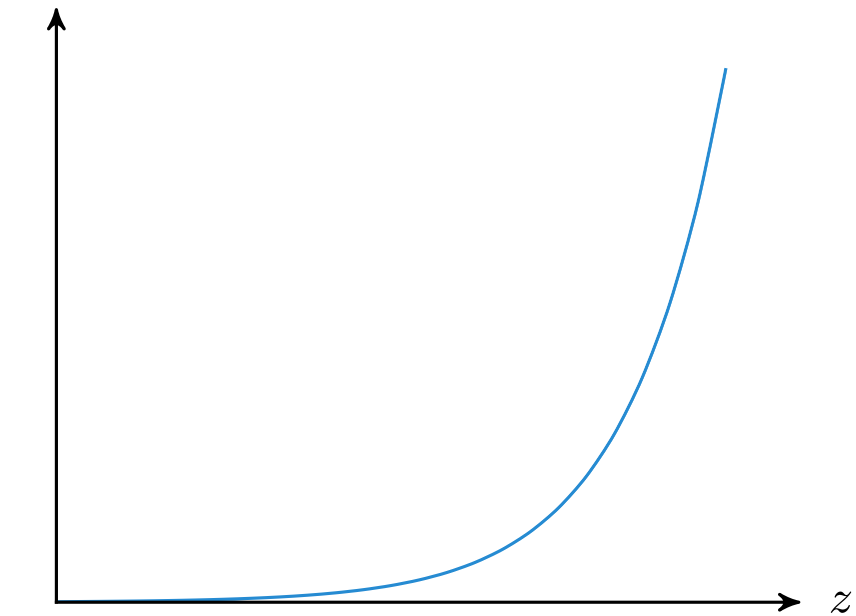
$$g(z) = 2^{2z}$$



$f(x) = x^2$



$g(z) = 2^{2z}$



Bytt ut variabelen – gjerne med logaritmen til den opprinnelige variabelen – og bytt funksjonen med en annen funksjon som oppveier for transformasjonen.

$$z \equiv \lg x$$

$$g(z) \equiv f(x)$$

$$T(n) = 2T(\sqrt{n}) + \lg n$$

$$m \equiv \lg n$$

$$T(n) = 2T(\sqrt{n}) + \lg n$$

$$
\begin{aligned}
m &\equiv \lg n \\
T(n) &= 2T(\sqrt{n}) + \lg n \\
&= 2T(\sqrt{2^m}) + m
\end{aligned}
$$

$$
\begin{aligned}
m &\equiv \lg n \\
T(n) &= 2T(\sqrt{n}) + \lg n \\
&= 2T(\sqrt{2^m}) + m = 2T(2^{m/2}) + m
\end{aligned}
$$

$$m \equiv \lg n$$

$$\begin{aligned} S(m) \equiv T(2^m) = T(n) &= 2T(\sqrt{n}) + \lg n \\ &= 2T(\sqrt{2^m}) + m = 2T(2^{m/2}) + m \end{aligned}$$

Vi bytter ut n med m og T med S slik at S(m) = T(n), og ender med en enklere, men ekvivalent, rekurrens.

$$m \equiv \lg n$$

$$S(m) \equiv T(2^m) = T(n) = 2T(\sqrt{n}) + \lg n$$

$$= 2T(\sqrt{2^m}) + m = 2T(2^{m/2}) + m$$

$$= 2S(m/2) + m$$

$$S(m) = 2S(m/2) + m$$

$$
\begin{aligned}
S(m) &= 2S(m/2) + m \\
&= \Theta(m \lg m)
\end{aligned}
$$

$$
\begin{aligned}
m &\equiv \lg n \\
S(m) &= 2S(m/2) + m \\
&= \Theta(m \lg m)
\end{aligned}
$$

$$
\begin{aligned}
m &\equiv \lg n \\
S(m) &= 2S(m/2) + m \\
&= \Theta(m \lg m) \\
&= \Theta(\lg n \lg \lg n)
\end{aligned}
$$

$$m \equiv \lg n$$

$$\begin{aligned} T(n) \equiv S(m) &= 2S(m/2) + m \\ &= \Theta(m \lg m) \\ &= \Theta(\lg n \lg \lg n) \end{aligned}$$

Vi løser den enklere rekurrensen, og bytter så m med lg n og får så svaret vi er ute etter.

$$m \equiv \lg n$$

$$\begin{aligned} T(n) \equiv S(m) &= 2S(m/2) + m \\ &= \Theta(m \lg m) \\ &= \Theta(\lg n \lg \lg n) \end{aligned}$$

D&C

D&C

D&C