Sidste gang

Algoritmer

Søgning

- Sekventiel søgning
- Binær søgning

Køretid

- Karakteristisk operation
- O-notation – recap.

Korrekthed

- Løkke-invarianter – afslutning

I dag:

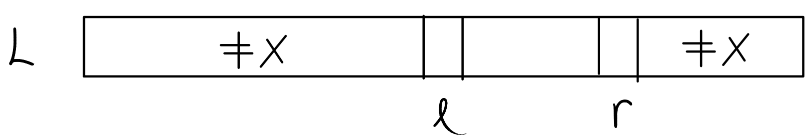
- Rekursive alg.

Binary Search (L, x)

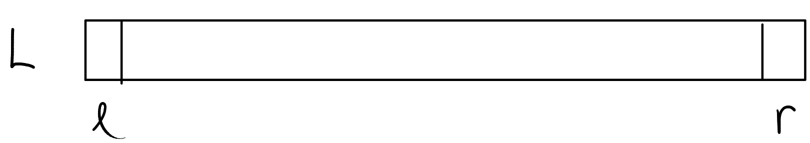
```
l = 1,   r = L.length,   m = ⌊(l+r)/2⌋
While *I*  l ≤ r  and  L[m] ≠ x
    If  x < L[m]
        r = m-1
    Else
        l = m+1
    m = ⌊(l+r)/2⌋
If  x == L[m]
    Return  m
Else
    Return „Not found"
```

Invariant I:

Hvis x findes i L, findes den i $L[l..r]$:

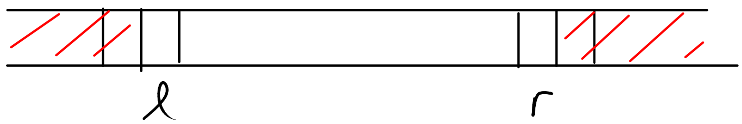


Første gang vi kommer til *I*, er det trivielt sandt:

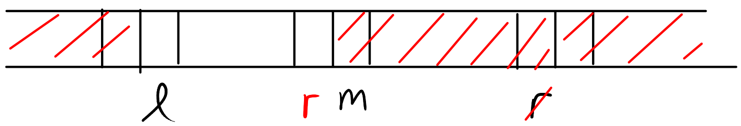


Generelt:

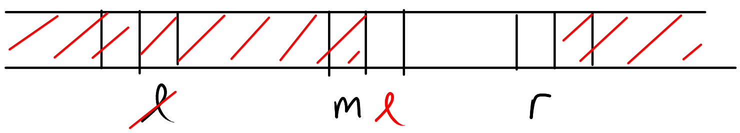
Lige inden vi checker $l \le r$ and $L[m] \neq x$, gælder I:



Efter gennemløb af løkken gælder I stadig:



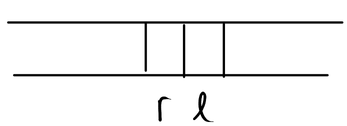
eller



Når vi til sidst forlader while-løkken, skyldes det en af to ting:

- $L[m] = x$
  I dette tilfælde returneres i, og det er indeks til en plads i L, hvor man kan finde x. D.v.s. korrekt output i dette tilfælde.

- $l > r$
  I dette tilfælde returneres „Not found". Da $l > r$, er $L[l..r]$ tomt:



Dermed følger det af I, at x ikke findes i L. D.v.s. også i dette tilfælde korrekt output.

Køretid:

Det vigtige er, om køretiden er proportional med $\log n$, n, n^2 eller...

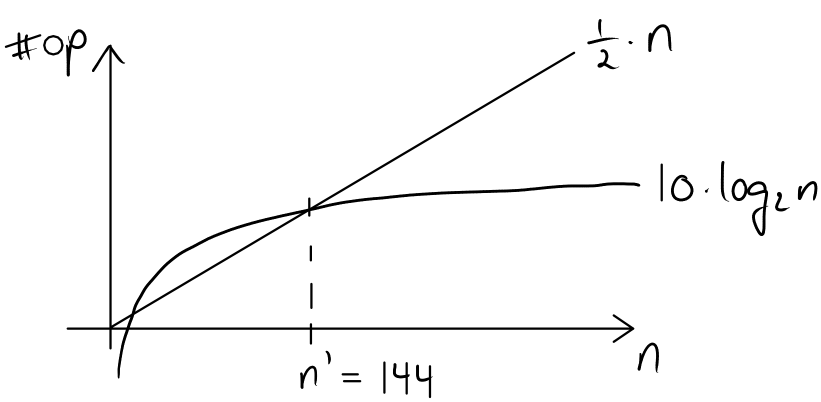
Eks:

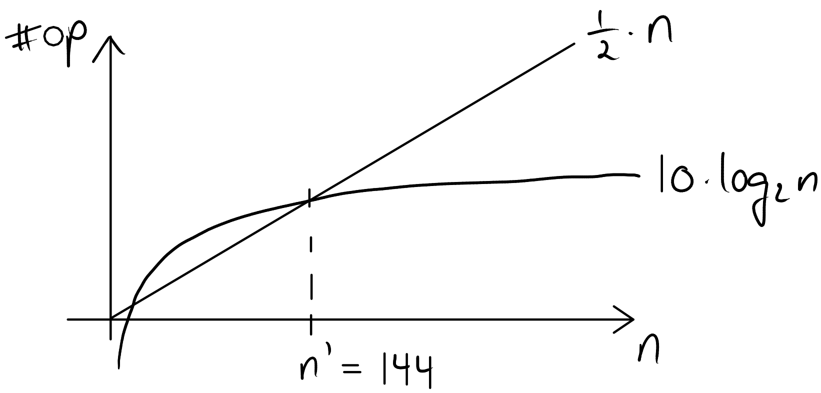
#op. i sek: 10^9

#op	$n=100$		$n=10^6$		$n=10^9$	
	#op.	tid	#op	tid	#op.	tid
$10 \cdot \lceil \log_2 n \rceil$	70	70 ns	200	200 ns	$\boxed{300}$	300 ns
$\frac{1}{2} n$	50	50 ns	$5 \cdot 10^5$	500 μs	$\boxed{10^9}$	1 s
$\frac{1}{100} n^2$	100	100 ns	10^{10}	10 s	10^{18}	116 døgn

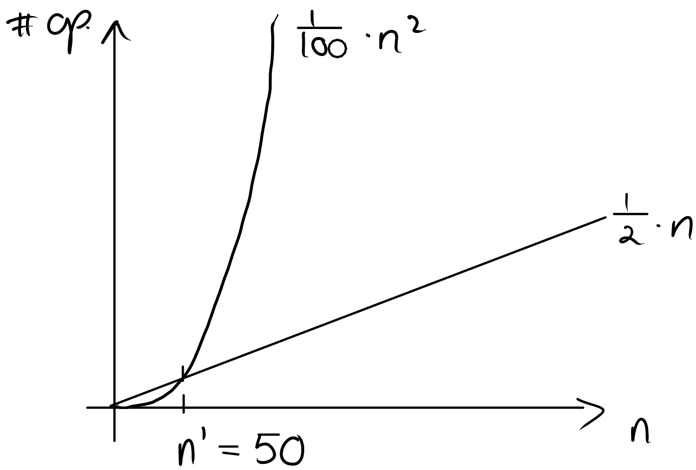
$10^9 \approx 3 \cdot 10^6 \cdot 300$

D.v.s. om køretiden er $\frac{1}{2} \cdot n$ eller $10 \cdot n$ er relativt uvigtigt. Det, der betyder noget, er, om den er proportional med n eller $\log n$.

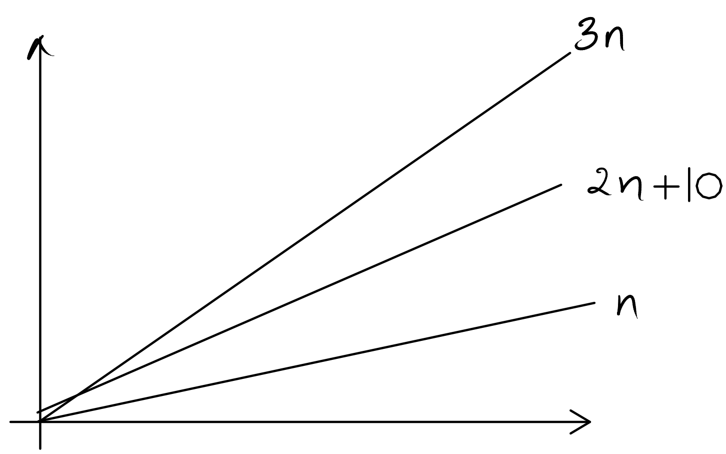




#op
$\frac{1}{2} \cdot n$
$10 \cdot \log_2 n$
$n' = 144$
n



#op.
$\frac{1}{100} \cdot n^2$
$\frac{1}{2} \cdot n$
$n' = 50$
n



3n
2n+10
n

$2n+10 \in O(n)$:

$$\frac{2n+10}{n} = 2+\frac{10}{n} \leq 3, \quad \text{for} \quad n \geq 10$$

$2n+10 \in O(n^2)$:

$$\frac{2n+10}{n^2} = \frac{2}{n} + \frac{10}{n^2} \leq 1, \quad \text{for} \quad n \geq 5$$

$10n\log_2 n + 3n \in O(n\log n)$:

$$\frac{10n\log_2 n + 3n}{n\log_2 n} = 10 + \frac{3}{\log_2 n} \leq 11, \quad \text{for} \quad n \geq 8$$

$n^2 \notin O(n)$:

$$\frac{n^2}{n} = n \longrightarrow \infty \quad \text{for} \quad n \to \infty$$