DM534 — Øvelser Uge 39

Introduktion til Datalogi, Efterår 2021

Jonas Vistrup

---

1 I

1.1

Er nedenstående en algoritme?

```
i=0
Sålænge i!=5
    i=i+2
```

SVAR: Nej. Det er instruktioner som er unambiguous og executable, men det er ikke en terminating process.

1.2

Betragt listen $L = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]$. I nedenstående spørgsmål tæller vi sammenligninger, som involverer elementer i listen.

(a) Hvor mange sammenligninger foretages der med SequentialSearch($L, 7$)?

(b) Hvor mange sammenligninger foretages der med BinarySearch($L, 7$)?

Antag nu, at L indeholder 10.000 elementer.

(c) Hvor mange sammenligninger foretager man i værste tilfælde med en sekventiel søgning i L?

(d) Hvor mange sammenligninger foretager man i værste tilfælde med en binær søgning i L?

SVAR a: 1,2,3,4,5,6,7 så 7 sammenligninger.

SVAR b: 10,5,7 så 3 sammenligninger.

SVAR c: 10.000, hvis det er det sidste element, eller elementet er ikke i listen.

SVAR d: $\lceil \log_2(10.000) \rceil = 14$.

1.3

Udfyld de manglende felter (undtagen dem i øverste række) i tabellen på side 13 i slides fra forelæsningen om algoritmer.

SVAR:

	1 s	1 min	1 døgn	1 år
n	10^9	$6 \cdot 10^{10}$	$9 \cdot 10^{13}$	$3 \cdot 10^{16}$
$n \log_2 n$	$4 \cdot 10^7$	$2 \cdot 10^9$	$2 \cdot 10^{12}$	$5,5 \cdot 10^{16}$
n^2	$3 \cdot 10^4$	$2 \cdot 10^5$	$9 \cdot 10^6$	$2 \cdot 10^8$
n^3	10^3	$4 \cdot 10^3$	$4 \cdot 10^4$	$3 \cdot 10^5$
2^n	30	35	46	55

1.4

Hvilke af følgende udsagn er sande?

(a) $n \in O(n)$. **SVAR:** True

(b) $2n + 5 \in O(n)$. **SVAR:** True

(c) $\sqrt{n} - \log(n) \in O(n)$. **SVAR:** True

(d) $(\log(n)^2) \in O(n \log(n))$. **SVAR:** True

(e) $n^2 \in O(n)$. **SVAR:** False

(f) $n \in O(n^2)$. **SVAR:** True

(g) $n \log(n) \in O(n^2)$. **SVAR:** True

(h) $n \log(n) \in O(n)$. **SVAR:** False

(i) $3n^2 + 2n + 1 \in O(n^2)$. **SVAR:** True

(j) $3n^2 + 2n + 1 \in O(n)$. **SVAR:** False

1.5

Angiv for hver af følgende algoritmer deres asymptotiske køretid i O-notation som funktion af n.

```
ALGORITME1(n)
    s = 0
    for i = 1 to n
        s = s + 1
    return s
```

```
ALGORITME2(n)
    s = 0
    for i = 1 to n
        for j = 1 to n
            s = s + 1
    return s
```

```
ALGORITME3(n)
    s = 0
    for i = 1 to n
        for j = i to n
            s = s + 1
    return s
```

```
ALGORITME4(n)
    s = 0
    for i = 1 to n
        for j = 1 to n
            if i == j
                for k = 1 to n
                    s = s + 1
    return s
```

SVAR: $O(n)$, $O(n^2)$, $O(n^2)$, $O(n^2)$

1.6

Betragt følgende algoritme til at finde det mindste tal i listen L.

```
MIN(L)
n = L.length
min = L[1]
For i = 2 to n
    If L[i] < min
        min = L[i]
Return min
```

(a) Hvad er algoritmens køretid?

(b) Opskriv en løkke-invariant for algoritmen, og bevis, at den altid finder det mindste element i L.

(c) Bemærk, at algoritmen er iterativ. Skriv en rekursiv version af algoritmen.

SVAR a: $O(n)$

SVAR b: min er det mindste element i $L[1...i]$.

Initialization: min er det mindste i $L[1..1]$ da der kun er et element og min er dét element.

Induktionsantagelse: min er det mindste element i $L[1...i]$.

Induktionsskridt: min er det mindste element i $L[1...i]$. $i' = i + 1$ og min' er det mindste af min og $L[i']$, derfor er min' det mindste af $L[1...i]$ og $L[i + 1]$, som betyder at det er det mindste af $L[1..i + 1] = L[1..i']$.

SVAR c:

```
Min(L, min, i)
if index > L.length
   Return min
if L[i] < min
   min = L[i]
Min(L, min, i+1)
```

2 II

2.1

Hvilke af følgende udsagn er sande?

(a) $n \in O(n^3)$. **SVAR:** True

(b) $n^3 \in O(n^2)$. **SVAR:** False

(c) $\log(n) \in O(n)$. **SVAR:** True

(d) $n \in O(n \log(n))$. **SVAR:** True

(e) $0.1n^2 + n + 10 \in O(n)$. **SVAR:** False

(f) $0.1n^2 + n + 10 \in O(n^2)$. **SVAR:** True

(g) $0.1n^2 + n + 10 \in O(n^3)$. **SVAR:** True

(h) $n^2 \log(n) \in O(n^2)$. **SVAR:** False

2.2

Angiv for følgende algoritme dens asymptotiske køretid i O-notation som funktion af n.

```
ALGORITME1(n)
    s = 0
    for i = 1 to n
        for j = i to n
            for k = i to j
                s = s + 1
    return s
```

SVAR: $O(n^3)$

2.3

Husk på algoritmerne til, ciffer for ciffer, at addere eller gange to tal i hånden (også nævnt på slides om repræsentation af tal). Hvis du ikke helt kan huske algoritmerne, er her et eksempel:

$$
\begin{array}{rr}
 & 321 \\
+ & 281 \\
\hline
 & 602
\end{array}
\qquad\qquad
\begin{array}{r}
321 \times 281 \\
\hline
281 \\
562 \\
843 \\
\hline
90201
\end{array}
$$

(a) Hvad er køretiden for at addere to tal med n cifre hver? Hvad er den karakteristiske operation?

(b) Hvad er køretiden for at gange to tal med n cifre hver? Hvad er den karakteristiske operation?

SVAR a: Læg to chiffer sammen og huske carry bit'en. Det tager $O(n)$.

SVAR b: Gange alle kombinationer af tal. Læg dem sammen til sidst. $O(n^2 + n^2) = O(n^2)$