

Øvelsestimer DM573 Uge 40/41

Husk at læse de relevante sider i slides før du/I forsøger at løse en opgave.

I: Løses i løbet af øvelsestimerne i uge 40

1. Er nedenstående en algoritme?

$$\begin{array}{l} i = 0 \\ \textbf{sålænge } i \neq 5 \\ \quad i = i + 2 \end{array}$$

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 $\text{SequentialSearch}(L, 7)$?
- (b) Hvor mange sammenligninger foretages der med $\text{BinarySearch}(L, 7)$?

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 ?
3. I tabellen på side 16 i slides fra Lenes forelæsning: udfyld de manglende felter i kolonnen for *1 minut* og kolonnen for *1 år* (i begge dog undtaget felterne i den øverste række).
 4. Hvilke af følgende udsagn er sande?

- (a) $n \in O(n)$
- (b) $2n + 5 \in O(n)$
- (c) $\sqrt{n} - \log(n) \in O(n)$
- (d) $(\log(n))^2 \in O(n \log n)$
- (e) $n^2 \in O(n)$
- (f) $n \in O(n^2)$
- (g) $n \log(n) \in O(n^2)$
- (h) $n \log(n) \in O(n)$
- (i) $3n^2 + 2n + 1 \in O(n^2)$
- (j) $3n^2 + 2n + 1 \in O(n)$

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

```

ALGORITME1( $n$ )
   $s = 0$ 
  for  $i = 1$  til  $n$ 
     $s = s + 1$ 
  returner  $s$ 

```

```

ALGORITME2( $n$ )
   $s = 0$ 
  for  $i = 1$  til  $n$ 
    for  $j = 1$  til  $n$ 
       $s = s + 1$ 
  returner  $s$ 

```

```

ALGORITME3( $n$ )
   $s = 0$ 
  for  $i = 1$  til  $n$ 
    for  $j = i$  til  $n$ 
       $s = s + 1$ 
  returner  $s$ 

```

```

ALGORITME4( $n$ )
   $s = 0$ 
  for  $i = 1$  til  $n$ 
    for  $j = 1$  til  $n$ 
      hvis  $i == j$ 
        for  $k = 1$  til  $n$ 
           $s = s + 1$ 
  returner  $s$ 

```

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

```

MINIMUM( $L$ )
 $n = L.length$ 
 $min = L[1]$ 
for  $i = 2$  til  $n$ 
  hvis  $L[i] < min$ 
     $min = L[i]$ 
returner  $min$ 

```

- (a) Hvad er algoritmens køretid?
- (b) Opskriv en 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.

II: Løses hjemme inden øvelsestimerne i uge 41

1. Hvilke af følgende udsagn er sande?
 - (a) $n \in O(n^3)$
 - (b) $n^3 \in O(n^2)$
 - (c) $\log(n) \in O(n)$
 - (d) $n \in O(n \log(n))$
 - (e) $0.1n^2 + n + 10 \in O(n)$
 - (f) $0.1n^2 + n + 10 \in O(n^2)$
 - (g) $0.1n^2 + n + 10 \in O(n^3)$
 - (h) $n^2 \log(n) \in O(n^2)$
2. Angiv for følgende algoritme dens asymptotiske køretid i O -notation som funktion af n .

```

ALGORITME1( $n$ )
   $s = 0$ 
  for  $i = 1$  til  $n$ 
    for  $j = i$  til  $n$ 
      for  $k = i$  til  $j$ 
         $s = s + 1$ 
  returner  $s$ 

```

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:

		321 × 281
		281
	321	562
+	281	843
	602	90201

- (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?