

45011 Algoritmer og datastrukturer

Løsningsforslag eksamen 17. august 1994

Oppgave 1

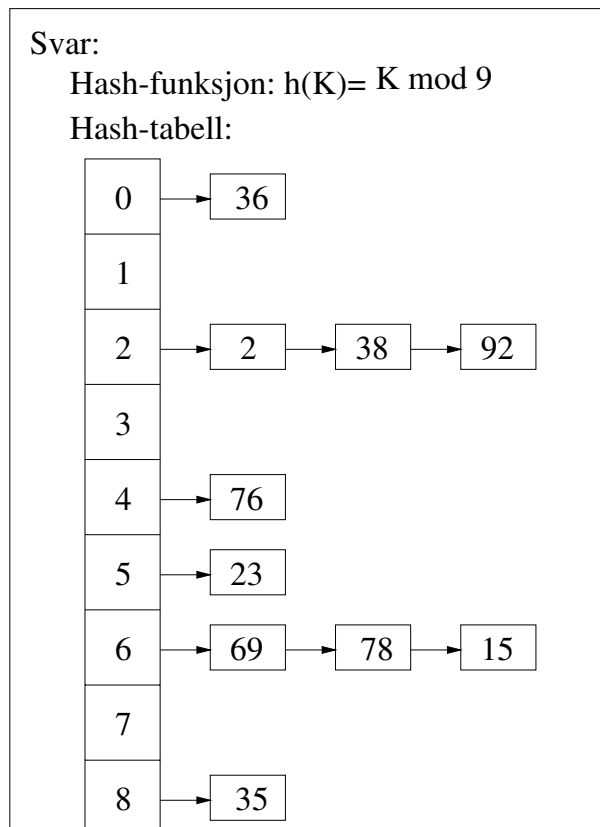
- a) $O(n^3)$ b) $O(n^3)$ c) $O(n^2)$ d) $O(2^n)$

Oppgave 2

a)

```
PROCEDURE SkrivUt (Rot: ^Node)
BEGIN
  IF Rot <> NIL THEN
  BEGIN
    SkrivUt(Rot^.Barn);
    SkrivUt(Rot^.Sosken);
    WritLn(Rot^.Verdi);
  END;
END;
```

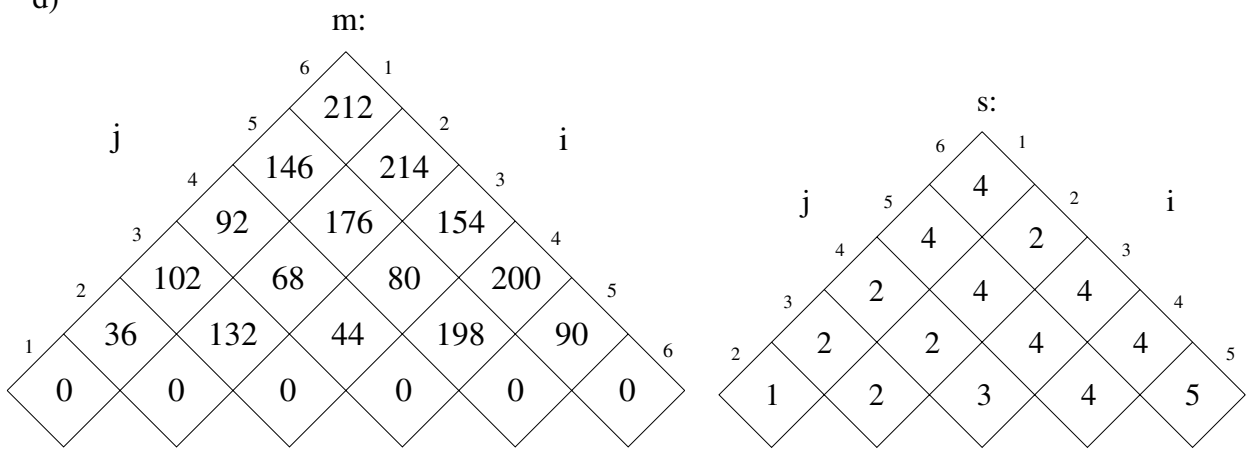
b)



c)

		Mulig	Umulig
a	710, 415, 90, 107, 232, 140	×	
b	783, 710, 694, 506, 415, 250, 140	×	
c	90, 783, 107, 661, 232, 120, 250, 140		×
d	2, 874, 170, 64, 163, 140	×	
e	874, 2, 798, 743, 170, 729, 447, 203, 140		×

d)



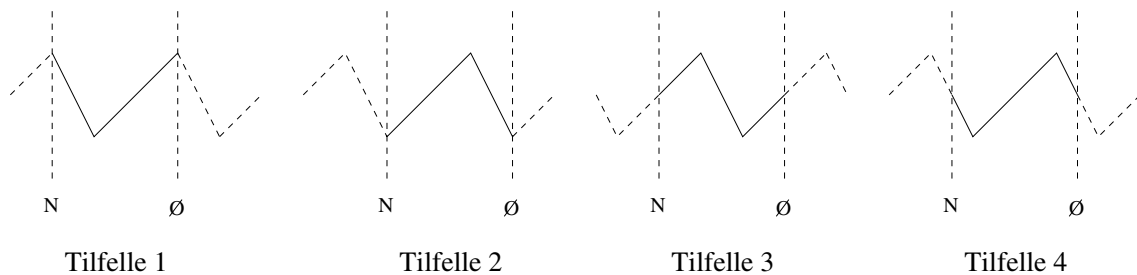
Parentessetting: $((A_1 \cdot A_2) \cdot (A_3 \cdot A_4)) \cdot (A_5 \cdot A_6)$

e)

a: 100 b: 101 c: 00 d: 01 e: 11
 Sparer: ca 24,67%

Oppgave 3

Vi har her tre mulige tilfeller (Se også eksamen 10. august 1992, oppgave 4):



Figur1: Unimodale sekvenser med ulike t -verdier

```

PROGRAM Stoerste;
VAR A:ARRAY [0..N-1] OF INTEGER;
Maks:INTEGER;

FUNCTION case1(M,N:INTEGER):INTEGER;
VAR I:INTEGER;
BEGIN
  IF N-M<2 THEN case1:=Max(A[M],A[N])
  ELSE BEGIN
    I:=M+(N-M) DIV 2;
    IF A[I]<A[I+1] THEN case1:=case1(I+1,N)
    ELSE case1:=case1(M,I);
  END;
END;

FUNCTION case2(M,N:INTEGER):INTEGER;
BEGIN
  case2:=Max(A[M],A[N]);
END;

FUNCTION case3(M,N:INTEGER):INTEGER;
VAR I:INTEGER;
BEGIN
  I:=M+(N-M) DIV 2;
  IF A[I]<A[I+1] THEN case3:=case1(I+1,N)
  ELSE IF A[I]<A[M] THEN case3:=case3(I+1,N)
  ELSE case3:=case3(M,I);
END;

FUNCTION case4(M,N:INTEGER):INTEGER;
VAR I:INTEGER;
BEGIN
  I:=M+(N-M) DIV 2;
  IF A[I]>A[I+1] THEN case4:=case1(M,I)
  ELSE IF A[I]>A[M] THEN case4:=case4(I+1,N)
  ELSE case4:=case4(M,I);
END

```

```

BEGIN
  {Initialiserer A}
  IF (A[0]<A[1]) THEN
    IF (A[N-2]>A[N-1]) THEN Maks:=case1(0,N-1)
    ELSE Maks:=case4(0,N-1)
  ELSE
    IF (A[N-2]>A[N-1]) THEN Maks:=case3(0,N-1)
    ELSE Maks:=case2(0,N-1);
  WriteLn(Maks);
  END.
  Kjøretid:  $\Theta(\lg n)$ 

```

Oppgave 4

Svar:

- Lag en graf $G^* = (V^*, E^*)$ der
 - $V^* = F \cup \{s, m\}$
 - $\delta(u, v)$ er korteste distanse fra $u \in V^*$ til $v \in V^*$ i G (Bruker Dijkstra en gang per node i F).
 - $E^* = \{(u, v) \text{ slik at } \delta(u, v) \leq t \text{ dersom } u = s \text{ eller } \delta(u, v) \leq d \text{ dersom } u \neq s\}$
- Finn korteste vei p^* fra s til m i G^* (Bruk Dijkstra).
- Konstruer veien i G vha. p^* : Erstatt alle kanter i p^* med den tilsvarende veien i G . (Disse veiene er allerede kjent fra kalkulasjonen av vektene i G^*).

Kjøretid:

- $O(f)$ for å konstruere V^* .
 $O(f \cdot (E + V \log V))$ for å konstruere E^* (Dijkstra f ganger).
- $O(E^* + V^* \log V^*) = (f^2 + f \log f) = O(f^2)$ for å utføre Dijkstra på G^* .
- $O(f \cdot V)$ for å erstatte $O(f)$ kanter med en $O(V)$ lang vei.

Total Kjøretid: $O(f \cdot E + f \cdot V \log V)$