EKSAMEN I FAG 45011
ALGORITMER OG DATASTRUKTURER

Tirsdag 12. januar 1993 kl. 0900-1300

Faglig kontakt under eksamen: Bård Kjos, tlf. 3470

Alle trykte og håndskrevne hjelpemidler tillatt.
Godkjent lommekalkulator tillatt.

Merk: Alle svar skal skrives i de angitte feltene på oppgavearket. Korte konsise svar honoreres best.

Oppgave 1 (20%)

a) (4%)

Vis at $(n+2)^2 = O(n^2)$ ved å spesifisere konstantene n_0 og c i definisjonen av O-notasjonen.

b) (4%)

Anta at du skal sortere n tall $x_1, ..., x_n$ der enhver x_i enten er 5 eller 120. Hvilken metode vil du benytte, og hva er kompleksiteten på kjøretiden (verste tilfelle)?

c) (4%)

Hvilke viktigste fordeler har et 2-3-tre i forhold til et ordinært binært søketre?

d) (4%)

Hvilke viktigste fordeler har Quicksort i forhold til Radix-sortering?

e) (4%)

Hva er de viktigste forskjeller og likheter mellom søking i binærtre og binær søking i tabell?

Oppgave 2 (20%)

Betrakt prosedyren BuildHeap som bygger en haug fra en tabell A[1..n]:

Procedure BuildHeap(A,n);
  begin
    for i := $\lfloor n/2 \rfloor$ **downto** 1 **do**
      Heapify(A,i);
  end;

a) (10%)

Skriv om BuildHeap til en rekursiv splitt-og-hersk prosedyre BuildHeap(A,i,n) der A[i] er roten til en del-haug. Hele haugen skal lages ved kallet BuildHeap(A,1,n).

b) (5%)

Finn rekurrensformelen for verste-tilfellet tidsforbruk for den rekursive prosedyren du fant i a).

c) (5%)

Løs rekurrensligningen under b) ved hjelp av "Master-metoden":

Oppgave 3 (20%)

Vi skal utføre matrisekjedemultiplikasjonen $A_1 \cdot A_2 \cdots A_5$, der matrisene har dimensjonene hhv. $(7 \times 3), (3 \times 12), (12 \times 8), (8 \times 10), (10 \times 4)$. Vis, ved å fylle ut tabellene m og s, fremgangsmåten for å finne optimal parentessetting ved dynamisk programmering. $m[i,j]$ er det minimale antall skalare multiplikasjoner for å beregne $A_i \cdots A_j$, mens $s[i,j]$er en k-verdi slik at $m[i,j] = m[i,k] + m[k+1,j] + p_{i-1}p_kp_j$. En matrise A_i har dimensjon $p_{i-1} \times p_i$.

Fyll ut parentessettingen: $A_1 \quad A_2 \quad A_3 \quad A_4 \quad A_5$

Oppgave 4 (20%)

La G=(V,E) være et flyt-nettverk der hver linje har kapasitet 1. s og t er hhv. kilde og sluk.

Anta, for både oppgave a) og b), at maksimal flyt allerede er funnet ved bruk av en flytalgoritme A(s,t).

a) (8%)

Anta at vi legger til en linje med kapasitet 1 mellom to vilkårlige noder u og v i G. Hvordan kan vi mest effektivt finne ny (muligens høyere) flyt i G? Merk at det er ny flyt i *alle* kanter som skal finnes.

b) (12%)

Anta at vi fjerner en kant mellom to vilkårlige noder u og v i G. Hvordan kan vi nå mest effektivt finne ny (muligens lavere) flyt i G? Også her er det ny flyt i alle kanter som skal finnes.

Oppgave 5 (20%)

Betrakt et uordnet sett S av $n \geq 2$ *forskjellige* tall. De tre deloppgavene ber deg foreslå en algoritme for å finne to tall x og y i S som tilfredsstiller en gitt betingelse. Med så få ord som mulig skal du beskrive dine algoritmer, evt. med henvisning til kjente algoritmer.

a) (4%)

Finn, ubetinget i $O(n)$ tid, x og y slik at $|x - y| \geq |w - z|$ for alle $w, z \in S$.

b) (4%)

Finn, ubetinget i $O(n \log n)$ tid, x og y slik at $|x - y| \leq |w - z|$ for alle $w, z \in S$.

c) (12%)

Finn, ubetinget i $O(n)$ tid, x og y slik at

$$|x - y| \leq \frac{1}{n-1}(\max_{z \in S} z - \min_{z \in S} z)$$

Du skal utvikle en rekursiv algoritme Find(...) med kjøretid $T(n) = T(n/2) + \Theta(n)$. Vi antar her at S er lagret i tabellen A.