**Universitetet i Bergen**
Det matematisk-naturvitenskaplige fakultet
Eksamen i emnet Inf102 - Algoritmer, programmering og datastrukturer
Fredag 16. desember 2005, kl. 09-14
Bokmålstekst

Tillatte hjelpemidler: Kalkulator.

## Oppgave 1

Vi sier at en sorteringsalgoritme er *stabil* dersom den bevarer den relative rekkefølgen mellom element med samme verdi. Dvs. at dersom verdien $v$ forekommer to ganger i inndata så skal den verdien av $v$ som kommer først i inndata også komme først i det sorterte resultatet.

Avgjør hvilke av følgende sorteringsalgoritmer som allerede er eller som enkelt kan gjøres stabile.

a) Utvalgsortering (en. selection sort)

b) Insettingsortering (en. insertion sort)

c) Flettesortering (en. merge sort)

d) Quicksort

e) Heapsort

f) Shellsort

## Oppgave 2

Bruk definisjonen for $O()$-notasjon i følgende oppgaver.

a) Vis at $2n^3 + n^2 + 7$ er $O(n^3)$.

b) Vis at $4n^2 + 2^n$ er $O(2^n)$.

c) Vis at $\log_b n = O(\log_a n)$ der $a$ og $b$ er basen for logaritmen.

d) Vis at $7n^2 + 5n$ *ikke* er $O(n)$.

**Oppgave 3**
For hver deloppgave blir det gitt to påstander. Avgjør hvilke som er sanne og hvilke som er falske. Motiver svarene.

a) Gitt $f(n) = 3n^3 + 2n^2 + n$

i) $f(n) = O(n^3)$
ii) $f(n) = O(n^4)$

b) La $T$ være et balansert binært søketre med $n$ noder.

i) Høyden til $T$ er aldri mer enn $\log n + 1$.
ii) Det trengs aldri mer enn $\log n$ operasjoner for å slette et element i $T$.

c) La $G$ være en sammenhengende graf med $n$ noder og $m$ kanter.

i) $n = O(m)$
ii) $m = O(n^2)$

d) La $G$ være en vektet graf. Anta at den korteste stien mellom noder $i$ og $j$ går gjennom node $k$ og at kostnaden til delstien fra $i$ til $k$ er $s$.

i)Alle korteste stier mellom $i$ og $j$ går gjennom $k$.
ii)Alle korteste stier mellom $i$ og $k$ har kostnad $s$.

**Oppgave 4**
a) Tegn det binære søketreet du får hvis du starter med et tomt tre og legger til følgende verdier: 10,5,12,2,7,8,6.

b) Tegn treet du får etter å ha slettet verdien 5 i treet fra oppgave a).

c) Tegn det binære treet som ved postorder traversering gir E,N,A,K,M,S,E og som ved inorder traversering gir E, A, N, K, E, S, M.

d) Skriv resultatet av en nivå traversering av treet fra oppgave c).

e) Gi et rekursivt uttrykk for det *minste* antall noder i et balansert binært søketre av høyde $h$.

f) Bruk løsningen fra e) for å vise at $h = O(logn)$ for et balansert binært søketre. (Hint, Fibonacci sekvensen har løsning slik at $F(k) > c^k$ for konstant $c > 1$).

**Oppgave 5**

I denne oppgaven skal vi se på algoritmer og datastrukturer for å hådtere *mengder* av tall (ikke nødvendigvis heltall). Merk at i en mengde forekommer hvert element høyst en gang og elementene ligger ikke nødvendigvis i en bestemt rekkefølge. Operasjonene vi vil se på er som følger.

`erMed(`$e, S$`)`, returnerer sann hvis tallet $e$ er med i mengden $S$ ellers falsk.

`leggTil(`$e, S$`)`, legger tallet $e$ til mengden $S$. Hvis $e$ allerede er med i $S$ skjer ingenting.

`union(`$S_1, S_2$`)`, lager og returnerer en peker til en ny mengde bestående av elementene i mengdene $S_1$ og $S_2$.

`snitt(`$S_1, S_2$`)`, lager og returnerer en peker til en ny mengde bestående av elementene som er i både $S_1$ og $S_2$.

Dersom $S_1 = \{1, 3, 4.5\}$ og $S_2 = \{1, 2, 5\}$ så returnerer `erMed(`$2, S_1$`)` falsk og `erMed(`$2, S_2$`)` sann. `union(`$S_1, S_2$`)` returnerer en peker til mengden $\{1, 2, 3, 4.5, 5\}$ mens `snitt(`$S_1, S_2$`)` returnerer en peker til mengden $\{1\}$. Etter et kall til `leggTil(2,`$S_1$`)` vil $S_1$ inneholde $\{1, 2, 3, 4.5\}$. Merk, du skal ikke bruke en *hash* tabell i noen av de påfølgende oppgavene.

a) Skisser datastruktur og algoritmer slik at både operasjonene `erMed` og `leggTil` kan implementeres mest mulig effektivt.

b) Bruk datastrukturen fra oppgave a) og beskriv algoritmer og gi kjøretid for algoritmer for å utføre `union` og `snitt`.

c) Se vekk fra løsningene for a) og b) og ta nå utgangspunkt i `union` og `snitt` og beskriv og analyser datastrukturer og algoritmer slik at disse operasjonene går fortest mulig.

d) Bruk datastrukturen fra oppgave c) og beskriv algoritmer og gi kjøretid for algoritmer for å utføre `erMed` og `leggTil`.

Fredrik Manne　　　　Fedor Fomin