

## Universitetet i Bergen

Det matematisk-naturvitenskapelige fakultet

Eksamen i emnet INF 102 - Algoritmer, datastrukturer og programmering

Fredag 30. november 2012, kl 09:00 - 12:00

Bokmål

Ingen tillatte hjelpemidler.

Du trenger ikke å skrive java-kode for noen av oppgavene. Det holder med pseudokode eller en beskrivelse av algoritmen.

**Alle svar må begrunnes!**

### Oppgave 1

Gi kjøretiden til hvert av programmene som en funksjon av  $n$ , der  $n$  er et positivt heltall. Kjøretiden skal gies som "størrelseorden" (eng. *order of growth*).

<p>a)</p> <pre>for(i = 1; i &lt; n; i = i * 2)   for(j = 0; j &lt; n; j++)     for(k = 0; k &lt; j; k++)       sum += A[j][k] * i;</pre>	<p>b)</p> <pre>i = 0; while (i &lt; n)   if (i % 2 == 0)     i = i + 3;   else     i = i - 1;</pre>
--	---

I oppgave c) og d) skal du bestemme kjøretiden til  $F(n)$ .

<p>c)</p> <pre>F(i) {     if (i == 1)       return i;     return(F(i/2) + F(i/2)); }</pre>	<p>d)</p> <pre>F(i) {     if (i == 1)       return i;     return(F(i-2) + F(i-2)); }</pre>
--	--

### Oppgave 2

a) Demonstrer "nedenfra og opp" flettesortering (eng. *bottom-up mergesort*) på følgende tallsekvens:

34, 12, 89, 8, 4, 45, 41, 9, 32, 75, 10, 34, 3, 21, 55, 43.

b) Utled kjøretiden til flettesortering.

c) Hvilken sorteringsalgoritme vil du bruke for å sortere en tabell:

1. hvor hvert element ikke er mer enn 10 plasser fra sin endelige plass?
2. hvor hvert element er et heltall mellom 1 og 1000?
3. som er raskest i de fleste tilfeller?
4. som har best mulig kjøretid i verste tilfelle, samtidig som den ikke bruker mer enn et par variabler ekstra plass?
5. som er raskest mulig, samtidig som den er stabil?
6. hvor du flytter minst mulig rundt på elementene?

### Oppgave 3

Vi skal nå se på symboltabeller (eng. *symbol tables*).

**NB!** Du skal ikke bruke *hashing* i denne oppgaven.

a) Beskriv og analyser datastrukturer og algoritmer slik at den langsomste av operasjonene `put(Key key, Value val)` og `contains(Key key)` går så raskt som mulig.

b) Unionen av to symboltabeller  $S_1$  og  $S_2$  inneholder alle element som enten er med i  $S_1$  eller i  $S_2$ , mens snittet av  $S_1$  og  $S_2$  inneholder alle element som *både* er med i  $S_1$  og i  $S_2$ . Dersom  $S_1$  inneholder nøkkelverdier  $\{A, C, D\}$  og  $S_2$  inneholder  $\{C, B, D\}$  så vil unionen av  $S_1$  og  $S_2$  inneholde  $\{A, B, C, D\}$  og snittet vil inneholde  $\{C, D\}$ .

Anta at  $S_1$  og  $S_2$  er gitt ved datastrukturene du beskrev i oppgave a). Beskriv og gi kjøretiden til to metoder som implementerer henholdsvis union- og snitt-operasjonene.

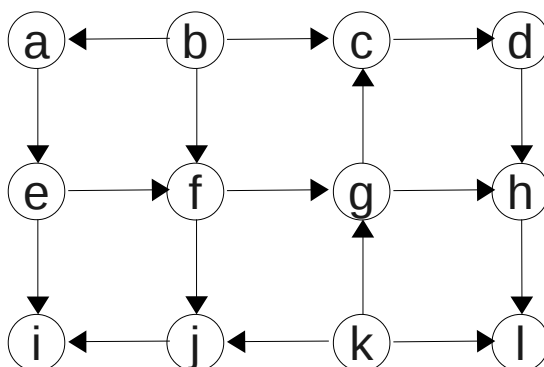
c) Se nå vekk fra løsningene du gav i oppgave a) og b) og ta utgangspunkt i union- og snitt-operasjonene. Beskriv og analyser datastrukturer og algoritmer slik at disse to går forrest mulig.

d) Anta at  $S_1$  og  $S_2$  er gitt ved datastrukturene du beskrev i oppgave c). Beskriv og analyser to algoritmer som implementerer henholdsvis `put()` og `contains()`.

### Oppgave 4

a) Beskriv og analyser en effektiv algoritme for å avgjøre om en rettet graf er asyklisk.

b) Demonstrer bruk av algoritmen fra a) på følgende graf. Der hvor du har mulighet, prosesser alltid nodene i leksikografisk ordning (dvs. i alfabetisk rekkefølge).



c) Bruk algoritmen fra a) til å gi en topologisk sortering av nodene i grafen fra b). Igjen, prosesser nodene i leksikografisk ordning.

d) Beskriv og analyser en effektiv algoritme som avgjør om en rettet asyklisk graf har en sti som går gjennom alle nodene.