

## Avsluttende eksamen i TDT4120 Algoritmer og datastrukturer

|                              |  |
|------------------------------|--|
| <b>Eksamensdato</b>          | 5. august 2008                                     |
| <b>Eksamenstid</b>           | 0900–1300  |
| <b>Sensurdato</b>            | 26. august   |
| <b>Språk/målform</b>         | Bokmål   |
| <b>Kontakt under eksamen</b> | Magnus Lie Hetland (tlf. 91851949)                 |
| <b>Tillatte hjelpemidler</b> | Alle trykte/håndskrevne; bestemt, enkel kalkulator |

Vennligst les hele oppgavesettet før du begynner, disponer tiden og forbered evt. spørsmål til faglærer kommer til eksamenslokalet. Gjør antagelser der det er nødvendig. Skriv kort og konsist, fortrinnsvis på angitt sted. Lange forklaringer og utledninger som ikke direkte besvarer oppgaven tillegges liten eller ingen vekt. Ved sensur vektlegges oppgavene som angitt. Innen hver oppgave teller alle deloppgaver (nummerert med bokstaver) like mye.

### Oppgave 1 (25%)

Anta at du kjører dybde-først-søk (DFS) i en urettet graf.

a. Hva slags kanter kan du ende opp med i DFS-treet?

. . .

Du kommer over et program som søker i binære søketrær. Du er usikker på om programmet bruker en eller annen form for balansering (som i *red-black*-trær) eller ikke. Du skal likevel forsøke å si noe om kjøretiden til ett enkelt søk med dette programmet.

b. Fyll inn de fire ulike kjøretidsbeskrivelsene i tabellen nedenfor.

|                   |                 |            |
|-------------------|-----------------|------------|
| <b>Best-case</b>  | $\Omega(\quad)$ | $O(\quad)$ |
| <b>Worst-case</b> | $\Omega(\quad)$ | $O(\quad)$ |

. . .

Anta at du skal implementere en prioritetskø, og vurderer om du skal bruke en sortert liste, en usortert liste, eller en haug (*heap*).

c. For hver løsning, hva blir kjøretiden for å sette inn ett element, for å ta ut ett element, og for å sette inn og ta ut  $n$  elementer? (For det siste tilfellet, anta at operasjonene gjøres for eksempel annenhver gang, så køens størrelse holdes på

$\Theta(n)$ .) Siden det er snakk om en prioritetskø er det alltid det minste elementet som skal tas ut. Fyll inn tabellen nedenfor.

| Metode         | Én innsetting | Ett uttak  | $n$ inn + $n$ ut |
|----------------|---------------|------------|------------------|
| Sortert liste  | $O(\quad)$    | $O(\quad)$ | $O(\quad)$       |
| Usortert liste | $O(\quad)$    | $O(\quad)$ | $O(\quad)$       |
| Haug (heap)    | $O(\quad)$    | $O(\quad)$ | $O(\quad)$       |

...

d. Løs rekurensen  $T(n) = 5T(n^{0.2}) + \lg n$ . Oppgi svaret i  $\Theta$ -notasjon. Vis hovedtrekkene i utregningen.

## Oppgave 2 (25%)

a. Forklar kort hvordan hauger (heaps) kan brukes til konstruksjon av Huffman-koder.

...

Betrakt følgende algoritme, BIDIRECTIONAL-DIJKSTRA, som er ment å finne den korteste avstanden  $d(s, t)$  mellom nodene  $s$  og  $t$  i en urettet graf med positive kantvekter:

La  $A$  og  $B$  være to instanser (kjøringer) av Dijkstras algoritme som starter i henholdsvis  $s$  og  $t$ .  $A$  og  $B$  alternerer, så etter hver iterasjon i  $A$  (dvs. hver node som farges svart) kjøres en iterasjon i  $B$ , og omvendt. Når  $A$  og  $B$  møtes i en node  $x$  (dvs. begge har farget  $x$  svart) vil  $d(s, t) = d(s, x) + d(x, t)$ .

b. Er algoritmen BIDIRECTIONAL-DIJKSTRA korrekt? Begrunn svært kort.

### Oppgave 3 (25%)

Det følgende er en del av en kjent algoritme (med endrede variabelnavn):

```

while  $A \neq B$ 
   $C \leftarrow \text{EXTRACT-MIN}(A)$ 
  for each  $D \in \text{Adj}[C]$ 
    if  $D \in A$  and  $w(C, D) < E[D]$ 
       $E[D] \leftarrow w(C, D)$ 

```

a. Hvilken algoritme er det snakk om?

. . .

Betrakt følgende algoritme, KNOTTY. Anta at første element i sekvensen  $a$  er  $a[0]$  og at sekvensen inneholder  $n$  elementer.

```

1   $i \leftarrow 1$ 
2   $j \leftarrow 2$ 
3  while  $i \leq n$ 
4    if  $a[i - 1] \geq a[i]$ :
5       $i \leftarrow j$ 
6       $j \leftarrow j + 1$ 
7    else
8       $a[i] \leftrightarrow a[i - 1]$ 
9       $i \leftarrow i - 1$ 
10   if  $i = 0$ 
11      $i \leftarrow 1$ 

```

b. Algoritmen KNOTTY er ment å sortere sekvensen  $a$  i stigende rekkefølge, men to av linjene i pseudokoden er gale. Oppgi (i tabellen nedenfor) hvilke linjer det gjelder, og hvordan de kan korrigeres.

(Du kan anta at indenteringen er korrekt, og trenger ikke angi den i svaret ditt.)

| Linje | Korrigert versjon |
|-------|-------------------|
|       |                   |
|       |                   |

**Oppgave 4 (25%)**

Et handelsskip følger en fast rute som går fra havn  $A$  til havn  $Z$ , og som går innom et antall (potensielt 0) havner på veien (den besøker  $n$  havner i alt). I hver havn kan kapteinen kjøpe og selge antikviteter. Prisen for hver antikvitet varierer fra havn til havn, og antikviteten kan naturligvis kun selges etter at den er kjøpt. Skipet har begrenset kapasitet (i kilo).

- a.** Vis at å finne en plan for kjøp og salg som maksimerer profitten er NP-HARD. (Vis tydelig at du benytter den generelle metoden fra pensum for denne typen bevis.)



. . .

Anta at du har oppgitt et sett  $S$  med linjestykker i planet, angitt ved koordinatene til deres start- og sluttpunkter. Du har også oppgitt to andre punkter i planet,  $p_1$  og  $p_2$ , og ønsker å finne den korteste veien fra  $p_1$  til  $p_2$  uten å krysse noen av linjestykkene i  $S$ .

- b.** Skisser kort en effektiv algoritme som løser problemet. Hva blir kjøretiden? Begrunn svaret.

