

## Avsluttende eksamen i TDT4120 Algoritmer og datastrukturer

<b>Eksamensdato</b>	18. august 2010
<b>Eksamenstid</b>	0900–1300
<b>Sensurdato</b>	8. september
<b>Språk/målform</b>	Bokmål
<b>Kontakt under eksamen</b>	Magnus Lie Hetland (tlf. 91851949)
<b>Tillatte hjelpemidler</b>	Ingen trykte/håndskrevne; bestemt, enkel kalkulator

- ! **Les alle oppgavene før du begynner, disponer tiden og forbered spørsmål til faglærer ankommer lokalet.**  
Gjør antagelser der det er nødvendig. Skriv kort og konsist på angitt sted. Lange forklaringer og utledninger som ikke direkte besvarer oppgaven tillegges liten eller ingen vekt.

Algoritmer kan beskrives med tekst, pseudokode eller programkode, etter eget ønske, så lenge det klart fremgår hvordan den beskrevne algoritmen fungerer. Korte, abstrakte forklaringer kan være vel så gode som utførlig pseudokode, så lenge de er presise nok. Kjøretider oppgis med asymptotisk notasjon, så presist som mulig.

a) Hvis du kjører DFS i en urettet graf vil du ikke få noen *forward* eller *cross edges*. Hvorfor?

Svar (11%):

b) Det kan være ønskelig å implementere QUICKSORT slik at den velger pivotobjektet tilfeldig (såkalt RANDOMIZED-QUICKSORT). Hva oppnår man med dette? Begrunn svaret kort.

Svar (11%):

- c) Anta at du implementerer Dijkstras algoritme, og at du bruker en usortert tabell eller liste som prioritetskø i stedet for en binær heap. Med andre ord kan DECREASE-KEY nå gjøres i konstant tid, mens EXTRACT-MIN tar lineær tid. For en graf med  $m$  kanter og  $n$  noder, hva blir den totale kjøretiden for algoritmen? Begrunn svaret kort.

Svar (11%):

- d) Du kommer over et problem FOO som ligner på SUBSET-SUM-problemet. Hva slags relasjon vil du prøve å etablere mellom FOO og SUBSET-SUM for å vise at FOO er NP-komplett?

Svar (11%):

- e) Du har oppgitt en bipartitt vektet graf med  $n$  noder og  $m$  kanter. Du skal finne en bipartitt matching med maksimal vekt. Du vet at alle kantvektene er unike heltallspotenser av 2. Hvordan vil du løse problemet, og hva blir kjøretiden? Begrunn svaret kort. (Nodene til en bipartitt graf kan deles i to mengder slik at ingen av mengdene har noen kanter internt. En matching er et subsett av kantene i grafen, der ingen av kantene har en felles node.)

Svar (11%):

Du skal flette sammen  $m = 2^k$  sorterte lister med lengder  $L_1, \dots, L_m$  (der  $k$  er et positivt heltall). Å flette sammen to lister med lengder  $L_p$  og  $L_q$  tar lineær tid og gir en ny liste med lengde  $L_p + L_q$ . Du vurderer to ulike algoritmer:

**Algoritme 1** (Splitt og hersk): Hvis du har kun to lister, flett dem sammen (i lineær tid). Ellers: Gruppér de  $m/2$  første listene i én mengde A og resten i en mengde B. Løs problemet for A og B rekursivt og flett de resulterende listene

**Algoritme 2** (Huffman): Gjenta det følgende til du står igjen med én liste: Velg ut de to korteste listene og flett dem sammen (og legg den resulterende listen tilbake i mengden).

f) Gi et eksempel på et sett med listelengder som vil gi asymptotisk forskjellig kjøretid med Algoritme 1 og Algoritme 2. Begrunn svaret kort.

Svar (11%):

g) Anta at du har  $n$  punkter i planet og du ønsker å finne et minimalt euklidsk spennetre over punktene. Det betyr at du skal finne et spennetre for den komplette grafen over punktene, der alle kanter har vekt lik avstanden mellom punktene. Gi en tett nedre asymptotisk grense for kjøretiden. Forklar svaret kort.

Svar (11%):

Det såkalte *ubegrensede* ryggsekkproblemet er nært beslektet med 0-1-ryggsekkproblemet. I stedet for at hvert objekt enten kan være med eller ikke, så kan det være med null eller flere ganger (men du kan fortsatt bare ha med hele objekter, i motsetning til *fractional knapsack*). Det er altså snakk om *objekt-typer* mer enn individuelle objekter.

h) Beskriv kort hvordan du vil løse det ubegrensede ryggsekkproblemet. Hva blir kjøretiden, som funksjon av  $n$  (antall objekt-typer) og  $W$  (ryggsekkkapasitet)? Begrunn svaret kort.

Svar (11%):

Du skal invitere venner til fest. Du vurderer et sett med  $n$  kandidater, men du vet at hver av dem bare vil ha det hyggelig dersom han eller hun kjenner minst  $k$  andre på festen. (Du kan anta at dersom A kjenner B så kjenner B automatisk A.)

i) Beskriv en algoritme som finner en størst mulig delmengde av de  $n$  vennene dine der alle kjenner minst  $k$  av de andre, dersom en slik delmengde eksisterer. Forklar kort hvorfor algoritmen er korrekt og optimal.

Svar (12%):