NTNU

Institutt for datateknikk
og informasjonsvitenskap

Eksamensoppgave i TDT4120 Algoritmer og datastrukturer

Faglig kontakt under eksamen	Magnus Lie Hetland
Tlf.	918 51 949
Eksamensdato	12. desember, 2014
Eksamenstid (fra–til)	0900–1300
Hjelpemiddelkode	D. Ingen trykte eller håndskrevne hjelpemidler tillatt. Bestemt, enkel kalkulator tillatt.
Målform/språk	Bokmål
Antall sider	9
Antall sider vedlegg	0
Kontrollert av	Pål Sætrom

Vennligst les hele eksamenssettet nøye før du begynner. Disponér tiden og forbered evt. spørsmål før faglærer kommer til eksamenslokalet.

* * *

Gjør antagelser der det er nødvendig. Skriv svarene dine på oppgavearket, som angitt, og hold dem korte og konsise, om mulig. Lange forklaringer som ikke direkte besvarer spørsmålene tillegges liten eller ingen vekt.

Med mindre annet er oppgitt kan du beskrive algoritmene dine i prosa, pseudokode eller programkode, etter eget ønske, så lenge det kommer klart frem hvordan algoritmen virker. Korte, abstrakte forklaringer kan være like gode som utførlig pseudokode, så lenge de er klare og presise.

Algoritmene dine bør være så effektive som mulige, med mindre annet er oppgitt. Kjøretider gis i asymptotisk notasjon, så presist som mulig.

Den maksimalt oppnåelige poengsummen er oppgitt for hver oppgave. Man kan maksimalt oppnå 100 poeng totalt.

Merk: I oppgave 2b brukes notasjonen $v.d$, i tråd med den nyeste utgaven av læreboka. Dette tilsvarer notasjonen $d[v]$ som brukes i eldre utgaver.

Oppgave 1

a) Hvilket problem løser Dijkstras algoritme?

Svar (5 p):

b) Hvilken traverseringsalgoritme bruker vi til topologisk sortering?

Svar (5 p):

c) Hvilken designmetode ville du bruke for å lage en algoritme for å parentessette en matrisekjedemultiplikasjon?

Svar (5 p):

d) Hva har Prims algoritme i min-prioritets-køen sin?

Svar (5 p):

e) Hva er den typiske gjennomsnittlige (*average-case*) kjøretiden for sorteringsalgoritmer basert på design-metoden *divide and conquer*?

Svar (5 p):

Oppgave 2

a) Hvilken designmetode ville du bruke for å lage en algoritme for det fraksjonelle ryggsekkproblemet (*the fractional knapsack problem*)?

Svar (5 p):

b) Under utførelsen av en eller annen korteste-vei-algoritme har vi $u.d = 5$, $v.d = 7$, $w(u, v) = 1$ og $w(v, u) = -2$. Vi utfører ett kall til $\textsc{Relax}(u, v, w)$ og deretter ett kall til $\textsc{Relax}(v, u, w)$. Hvilken verdi har $u.d$ nå?

Svar (5 p):

c) Hva kan vi si om det asymptotiske forholdet mellom φ og ψ hvis $a \cdot \varphi(i) \leq b \cdot \psi(i)$ for enhver $i \geq k$? (Her er φ og ψ ikke-negative reelle funksjoner, definert på positive heltall, a og b er positive reelle konstanter og k er en positiv heltallskonstant.)

Svar (5 p):

d) Svært kort, hva karakteriserer en god hashfunksjon?

Svar (5 p):

e) Du undersøker et ukjent problem A, og du ønsker å relatere det til problem B, som du vet har en kjøretid på $\Omega(n^2)$ i verste tilfelle. Du ønsker å bruke en reduksjon til å vise at den samme grensen også holder for A. Svært kort, hva kan du si om denne reduksjonen?

Svar (5 p):

Oppgave 3

Du ønsker å sortere sekvensen $A = (a_1, a_2, \dots, a_n)$. Det er velkjent at for sammenligningsbasert sortering er $\Omega(n \log n)$ den bese kjøretiden vi kan få, i forventet (*average-case*) og verste tilfelle.

a) Anta at elementene er reelle tall, distribuert etter en gitt sannsynlighetsfordeling, som kan beregnes i konstant tid for ethvert element. Hva er den beste forventede kjøretiden du kan få, og hvordan kan du oppnå den? Forklar kort.

Svar (5 p):

b) Anta nå at elementene er positive heltall, og at $a_i \leq p(n)$ for $i = 1 \dots n$, der p er et polynom. Hva er den beste forventede kjøretiden du kan få, og hvordan kan du oppnå den? Forklar kort.

Svar (5 p):

Oppgave 4

a) Hvilket problem løser Floyd-Warshall? Hvilke antagelser må eller kan du gjøre, og hvordan påvirker de kjøretiden? Forklar kort.

Svar (5 p):

b) Hvilket problem løser Ford-Fulkerson? Hvilke antagelser må eller kan du gjøre, og hvordan påvirker de kjøretiden? Forklar kort.

Svar (5 p):

Oppgave 5

a) Løs rekurrensen $T(n) = 2T(n/2) + T(n)/2 + n$, der $T(1) = 1$. Oppgi svaret med asymptotisk notasjon.

Svar (5 p):

b) En algoritme prosesserer et bilde i form av en $n \times n$-matrise av piksler ved å dele det opp i ikke-overlappende kvadrater av størrelse $(n/2) \times (n/2)$, prosessere disse rekursivt og så kombinere resultatene i lineær tid, som en funksjon av antall piksler involvert. Hva er den totale kjøretiden, som funksjon av n?

Svar (5 p):

Oppgave 6

Du planlegger å plante hager og grave vanningskanaler fra en gitt vannkilde. Du kan bestemme plasseringen selv, og har bestemt deg for det følgende. Kanalene utgjør et binærtre med vannkilden som rot og hagene som løvnoder. Avstandene er helt regulære. Avstanden langs en kanal fra kilden til en forgrening, eller fra en forgrening til en hage eller en annen forgrening er den samme, og er satt til $\ell(n)$ for et gitt antall hager, n.

a) Hva er den totale lengden med vanningskanaler som må graves, som funksjon av n?

Svar (5 p):

Hver hage i krever en mengde vann per dag gitt av w_i. For å oppnå dette må noen kanalsegmenter graves så de er bredere enn andre, og det fører til mer graving. Hvis hage i er plassert i en avstand på d_i segmenter fra vannkilden så bidrar den til den nødvendige gravingen med en verdi proporsjonal med $d_i w_i$. Den totale mengden graving er dermed proporsjonal med $\sum_{i=1}^{n} d_i w_i$.

b) Beskriv en algoritme for å bestemme hvordan et slikt tre av kanaler skal konstrueres for å minimere mengden graving som kreves.

Svar (5 p):

Oppgave 7

I det såkalte balanserte partisjonerings-problemet (*balanced partition problem*) har vi oppgitt n heltall $a_1, \ldots, a_n$ i verdiområdet $0 \ldots k$, og ønsker å fordele dem i to mengder S_1 og S_2 slik at $\left| \sum_{x \in S_1} x - \sum_{y \in S_2} y \right|$ minimeres.

a) Vis at problemet er NP-hardt (*NP-hard*).

Svar (5 p):

b) Skissér en algoritme for å løse problemet i pseudopolynomisk tid.

Svar (5 p):