

TDT4120 Algoritmer og datastrukturer

Eksamen, 3. august 2026, 09:00–13:00

Faglig kontakt Magnus Lie Hetland

Hjelpemiddelkode E

Løsningsforslag

Løsningsforslagene i rødt nedenfor er *eksempler* på svar som vil gi uttelling. Det vil ofte være helt akseptabelt med mange andre, beslektede svar, spesielt der det bes om en forklaring eller lignende. Om du svarte noe litt annet, betyr ikke det nødvendigvis at du svarte feil!

Merk: Det å forstå hva det spørres etter er *en del av oppgaven*. Enkelte oppgaver kan være konstruert slik at det er nettopp dette som er utfordringen, så om du ikke får dem til, vil de kunne virke uklare eller tvetydige.

- 5 % **1** Hva er kjøretiden til BFS? Oppgi svaret i O -notasjon.

Anta at grafen er representert med nabolister.

$O(V + E)$

- 5 % **2** Hva er kjøretiden til INSERTION-SORT?

Oppgi både øvre og nedre grense, om mulig kombinert vha. Θ -notasjon.

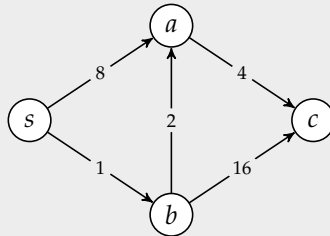
$O(n^2), \Omega(n)$

- 5 % **3** Hva kreves for at FLOYD-WARSHALL skal gi riktig svar, hvis noen av kantvektene er negative?

Ingen negative sykler.

- 5 % **4** Om du utfører QUICKSORT (som beskrevet i læreboka) på en tabell der alle n elementer er like, hva blir kjøretiden, i Θ -notasjon? Hjelper det å bruke den randomiserte utgaven, RANDOMIZED-QUICKSORT? Forklar kort.

Figur 1



$\Theta(n^2)$. Randomisering hjelper ikke.

- 5 % 5 Stemmer følgende påstand?

$$2^{n+1} = O(2^n)$$

Forklar kort.

Ja: $2^{n+1} = 2 \cdot 2^n = O(2^n)$.

- 5 % 6 Stemmer følgende påstand?

$$4^n = O(2^n)$$

Forklar kort.

Stikkord: Nei: $4^n > c \cdot 2^n$ for store nok n , uansett konstant c .

- 5 % 7 Du skal utføre RADIX-SORT på tabellen $A = \langle 011, 110, 001, 100 \rangle$.

Hva er A etter *andre* runde med sortering?

Her består altså hvert tall av tre siffer. For eksempel har 011 sifrene 0, 1 og 1.

$\langle 100, 001, 110, 011 \rangle$.

Her får man 2 poeng for $\langle 001, 100, 011, 110 \rangle$, 1 poeng for andre rekkefølger der midterste siffer er sortert og 0 poeng ellers.

- 5 % 8 Om du utfører DIJKSTRA på grafen i figur 1, med startnode s , hvilke ulike verdier får $c.d$ under utførelsen?

Her skal du altså liste opp alle verdiene feltet $c.d$ får, i rekkefølge, etter hvert som algoritmen utføres. Oppgi hvilken verdi feltet initialiseres til, og verdien det får hver gang det endres etter det.

$\infty, 17, 7$.

Nodene hentes ut i rekkefølge s, b, a, c med nøkler $0, 1, 3, 7$.

- 5% 9 I en instans av det kontinuerlige ryggsekkproblemet (*fractional knapsack*) har du tre gjenstander med følgende verdier $v[1:3]$ og vekter $w[1:3]$:

$$v = \langle 32, 30, 45 \rangle$$

$$w = \langle 8, 12, 15 \rangle$$

Kapasiteten W er 20. Hva blir den optimale totalverdien? Forklar kort.

Totalverdi: 68.

Her velges hele gjenstand 1, så 12 of gjenstand 3 sine 15 kg. De velges grådig etter kilopris 4, 3 og 2.5.

- 5% 10 Du har løst stavkappingsproblemet (*rod cutting*), og ender med følgende:

$$r[0:8] = \langle 0, 3, 7, 12, 15, 19, 24, 27, 31 \rangle$$

$$s[1:8] = \langle 1, 2, 3, 1, 2, 3, 1, 2 \rangle$$

Her er $r[j]$ den optimale prisen for en stav av lengde j og $s[j]$ er lengden til den første delen av den optimale løsningen (når $j > 0$).

Hva blir lengdene til de ulike bitene i en optimal løsning for en stav med total lengde 8 og hva blir den totale verdien?

Lengder: 2, 3, 3

Total verdi: 31

Dette finnes via $s[8] = 2$, $s[6] = 3$ og $s[3] = 3$.

- 5% 11 Gløgsund, Klokland og Flinckenhagen diskuterer flyt. De er enige om at i et flytnett (*flow network*) med heltallskapasiteter så vil den største mulige flytverdien være et heltall. Men:

- Gløgsund mener også at flyten i alle kantene da også må være heltall.
- Klokland mener at det stemmer for løsninger funnet med Ford–Fulkerson-metoden, men at det kan finnes andre optimale løsninger der noen av kantene ikke har heltallsflyt.
- Flinckenhagen mener også at det kan finnes optimale løsninger der noen kanter ikke har heltallsflyt, men hun mener også at Ford–Fulkerson-metoden også kan ende opp med noen av disse.

Hvem av dem har rett? Gi en kort, intuitiv forklaring.

Forklaringen trenger ikke være et bevis eller fullstendig argument.

Her har Klokland rett. Dette følger av heltallsteoremet og et moteksempel dersom man ikke følger Ford–Fulkerson.

Man kan også forklare den første delen med alle endringer er størst mulig, basert på en heltallig flaskehals, som også bare endres med heltallsverdier underveis.

5 % 12 Hva er forskjellen mellom NP-hardhet og NP-kompletthet? Forklar kort.

Du kan anta at det her kun er snakk om beslutningsproblemer.

Forskjellen er at NP-komplette problemer også er i NP, noe som ikke kreves av NP-harde problemer generelt.

5 % 13 Løs rekurrensen $T(n) = 3T(n/2) + n$. Oppgi svaret i Θ -notasjon.

Hint: $\log_2 3 = 1.5496 \dots$

$\Theta(n^{1.5496\dots})$

Følger av masterteoremet. $\Theta(n^{\log_2 3})$ er et fullgodt svar.

5 % 14 Løs rekurrensen $T(n) = T(n/5) + T(4n/5) + n$.

Bruk et rekursjonstre og gi en kort, overordnet forklaring på hvordan treet gir både en øvre og nedre grense. Oppgi svaret i Θ -notasjon.

Poenget er å bruke *tankegangen* bak rekursjonstrær. Du trenger ikke faktisk tegne opp eller oppgi selve treet.

$\Theta(n \lg n)$

Hvert nivå summerer til $O(n)$, og treet har dybde $O(\lg n)$, som gir $O(n \lg n)$. Vi har også $\Omega(\lg n)$ fulle nivåer, som gir $\Omega(n \lg n)$.

Tilsvarende problem (med $n/3$ og $2n/3$) diskuteres i 4.4 i pensumboka.

5 % 15 Din venn Gløgsund påstår at hun har konstruert en metode som tar inn en min-haug (*min-heap*) av lengde n og gjør den om til en maks-haug (*max-heap*) med de samme elementene, med kjøretid $O(n)$. Kan hun ha rett? Forklar kort.

Ja. Bruk BUILD-MAX-HEAP.

- 5% 16 Visdal vil finne spenntrær i en sammenhengende, urettet graf med positive kantvekter. Han prøver å implementere Prim's algoritme (MST-PRIM), men der algoritmen sammenligner eller tilordner $w(u, v)$, skriver han $u.key + w(u, v)$ i stedet. Beskriv hva denne modifiserte algoritmen produserer. Utgjør mengden av kanter på formen $(v, v.\pi)$ fortsatt et spenntré? Er det i så fall alltid et minimalt spenntré? Forklar kort.

Dette er DIJKSTRA. Det gir et spenntré, men ikke nødvendigvis et MST.

- 5% 17 Lurvik har tegnet en rettet, asyklisk graf (*directed acyclic graph*). Han påstår (a) at den har minst to ulike topologiske sorteringer og (b) at den inneholder en rettet hamiltonsti, altså en sti som besøker hver node nøyaktig én gang. Tror du han har rett? Forklar kort.

Nei. Hamiltonstien gir en fullstendig ordning.

- 5% 18 Flinckenhagen vil leie et sett med utsalgsboder på Martnan.* Bodene hun kan velge mellom ligger i en ring rundt torget, noen med bedre plassering enn andre; bod i gir en forventet profitt $p_i \geq 0$, for $i = 1 \dots n$. Reglene forbyr henne å leie to boder ved siden av hverandre. Hun vil velge et lovlig sett med boder som maksimerer den totale forventede profitten. Beskriv en effektiv algoritme for problemet. Hva blir kjøretiden? Forklar.

Merk: Bod 1 og n ligger også ved siden av hverandre.

Bruk dynamisk programmering. Ta med første bod (og eliminer naboer) eller ikke; løs videre fra andre bod og velg maks av de to alternativene.

Første bod har to naboer (andre og siste bod). For andre delinstanser har første bod kun én nabo, nemlig andre bod.

- 5% 19 I en m meter lang tunnel finnes det n monteringspunkter, $x_1, x_2, \dots, x_n$ meter fra inngangen. Du skal velge ut k av disse, der det skal monteres antenner. For å minimere interferens, skal den *minste* avstanden mellom to antenner være så stor som mulig. Beskriv en algoritme for problemet med kjøretid $\Theta(n \lg m)$.

Hint: Betrakt problemet der «så stor som mulig» erstattes med «minst d meter».

Du kan anta at $x_1, \dots, x_n$ er oppgitt i sortert rekkefølge og at x_i og m er positive heltall. Det er mulig å bruke dynamisk programmering for å få en kjøretid på $\Theta(nk)$, men det er ikke dette vi er ute etter.

* Martnan er et årlig marked på torget i Trondheim.

For å se om avstanden kan være minst d meter, velg punkter grådig. (Du taper aldri på å velge første monteringspunkt.) For å finne største d som fungerer, bruk binærsøk, f.eks. i området $1 \dots m$.

For interesserte: En løsning basert på dynamisk programmering kan f.eks. bruke en tabell $A[i, j]$ for svaret der vi plasserer første av j antenner på monteringspunkt i . En rett frem løsning vurderer alle påfølgende posisjoner for neste antenne, som gir $\Theta(n^2k)$. Vi kan optimere dette ved å iterere over i fra høyre til venstre når vi fyller ut for j . Når vi plasserer førstse antenne lenger til venstre, vil optimum for neste posisjon aldri bevege seg til høyre, så vi trenger ikke begynne på nytt for hver posisjon; vi starter fra forrige optimum, og itererer mot venstre til nåværende optimum. Det gir $\Theta(nk)$.

- 5% **20** Du får oppgitt en sammenhengende urettet graf G og vil finne et spennetre for G med så få løvnoder som mulig. Forklar enten hvordan dette kan gjøres effektivt eller hvorfor det trolig *ikke* kan gjøres effektivt.

Et slikt tre vil være en sti (to løvnoder) som besøker alle nodene én gang (spennetre), om det eksisterer, og å finne en slik sti er NP-hardt. Vi kan f.eks. redusere fra HAM-CYCLE ved å duplisere én av nodene i grafen. Grafen har da en slik sti hvis og bare hvis originalen har en hamiltonsykel.

Å avgjøre om en graf har en hamiltonsti er et velkjent NP-komplett problem, men det er sykkelvarianten som er i pensum. Om man svarer basert på hamiltonstiproblemet, HAM-PATH, er det også et fullgodt svar. Det er også akseptabelt med mer generelle reduksjoner, med flere kall til orakelet, etc. Det spørres her bare etter et argument for at problemet trolig ikke kan løses effektivt, ikke spesifikt om at det er NP-hardt under Karp-reduksjoner.