TDT4120 Algoritmer og datastrukturer

Eksamen, 28. november 2019, 09:00–13:00

Faglig kontakt Magnus Lie Hetland
Hjelpemiddelkode D

Oppgaver

5 % **1** Vurder følgende utsagn om DIJKSTRA:

> Kantvektene kan være negative, i motsetning til i BELLMAN-FORD.

Stemmer dette? Svar ja eller nei og forklar kort.

5 % **2** Vurder følgende utsagn om INSERTION-SORT:

> Den har kjøretid $\Omega(n \lg n)$ i beste tilfelle (*best-case*).

Stemmer dette? Svar ja eller nei og forklar kort.

5 % **3** Vurder følgende utsagn om splitt-og-hersk (*divide-and-conquer*):

> Metoden bør unngås når vi har overlappende delproblemer.

Stemmer dette? Svar ja eller nei og forklar kort.

5 % **4** Vurder følgende utsagn om TOPOLOGICAL-SORT:

> Nodene sorteres etter synkende starttid (*discover-time*).

Stemmer dette? Svar ja eller nei og forklar kort.

5 % **5** Vurder følgende utsagn om hauger (*heaps*):

> BUILD-MAX-HEAP har kjøretid $\Theta(n \lg n)$.

Stemmer dette? Svar ja eller nei og forklar kort.

5 % **6** Hvilke algoritmer i pensum finner korteste veier fra én til alle (*single-source shortest paths*) i vektede, rettede grafer?

Ikke inkluder algoritmer som finner korteste veier fra alle til alle.

5 % **7** Hvilke algoritmer i pensum finner minimale spenntrær (*minimum spanning trees*)? (Vi antar her vektede, urettede, sammenhengende grafer.)

5 % **8** Følgende delvis sensurerte lemma er tatt fra et delkapittel i læreboka som viser at en pensumalgoritme er korrekt.

> **Lemma** [redacted]
>
> Let C be an [redacted] in which each [redacted] $c \in C$ has frequency *c.freq*. Let x and y be two [redacted] in C having the lowest frequencies. Then there exists an optimal [redacted] for C in which the [redacted] for x and y have the same length and differ only in the last [redacted].

Hvilken algoritme er det snakk om, og hvilket problem løser den?

5 % **9** Din venn Smartnes har gitt deg følgende tabell, som hun påstår er en maks-haug (*max-heap*):

$A = \langle 7, 3, 8, 6, 5, 9, 4, 2 \rangle$

Du er ikke enig i at dette er en haug, men sier deg likevel villig til å utføre HEAP-EXTRACT-MAX på tabellen, *selv om resultatet blir feil*. (Du skal altså bare utføre trinnene i HEAP-EXTRACT-MAX, uten å korrigere A på noe vis.)

Hvordan ser A ut etterpå?

(Oppgi kun de 7 første elementene av A etter operasjonen.)

5 % **10** Hva er $O(n) + \Omega(n) + \Theta(n) + o(n) + \omega(n)$?

5 % **11** Løs følgende rekurrens eksakt, der n er et positivt heltall:

$$T(1) = 1$$
$$T(n) = T(n-1) + 2^{n-1} \qquad \text{(hvis } n > 1\text{)}$$

Oppgi svaret *uten* bruk av asymptotisk notasjon.

5 % **12** Din venn Klokland har laget en versjon av MERGE med kjøretid $\Theta(n^2)$. Om du bruker Kloklands versjon i stedet for den vanlige, hva blir kjøretiden til MERGE-SORT? Oppgi svaret i Θ-notasjon. (Evt. forklar kort.)

5 % **13** I et flytnett, hvis du har restkapasiteter (*residual capacities*) $c_f(u, v) = 9$ og $c_f(v, u) = 3$, og $f(u, v) > 0$, hva er flyten $f(u, v)$ og kapasiteten $c(u, v)$? Oppgi verdiene som normalt, som to tall med skråstrek imellom, f.eks. 1/2 hvis du mener $f(u, v) = 1$ og $c(u, v) = 2$.

5 % **14** Vurder følgende utsagn om Floyd-Warshall:

> Hvis $d_{ij}^{(k)}$ settes til $d_{ik}^{(k-1)} + d_{kj}^{(k-1)}$ så settes $\pi_{ij}^{(k)}$ til $\pi_{ik}^{(k-1)}$.

Stemmer dette? Svar ja eller nei og forklar kort.

5 % **15** Vurder følgende utsagn om minimale spenntrær (*minimum spanning trees*) i en vektet, urettet graf $G = (V, E)$ der alle kantene har forskjellig vekt:

> Hvis nodene V kan deles i mengdene X og Y (uten overlapp) så vil det minimale spenntreet nødvendigvis inneholde kanten mellom X og Y med lavest vekt.

Stemmer dette? Svar ja eller nei og forklar kort.

(Her holder det ikke nødvendigvis å beskrive hva en pensumalgoritme gjør.)

5 % **16** Din venn Lurvik påstår han har laget en algoritme basert på dynamisk programmering, som setter sammen det kuleste antrekket basert på klærne i garderoben hans. Når du ser på algoritmen, ser du at den kun beregner kulhetsgraden til det kuleste antrekket, *uten å faktisk si noe om hvilke klær som inngår*! Han mener det er en triviell forskjell, og at det vil være enkelt å legge til den funksjonaliteten. Hva tror du? Forklar kort.

5 % **17** Din venn Gløgsund vil vite om en graf har et minimalt spenntre med vekt mindre eller lik k, og mener hun har klart å redusere problemet i polynomisk tid til handelsreiseproblemet (*the traveling-salesman problem*). Din andre venn Klokland klarer ikke helt å følge beviset hennes, men mener det umulig kan stemme. Tror du det er mulig? Forklar kort.

5 % **18** Anta at du har en rettet graf $G = (V, E)$ med positive heltalls-kantvekter, der $s, t \in V$. Denne grafen kan ha flere enkle stier fra s til t med minimal lengde, det vil si flere «korteste» veier. Hvordan vil du finne den av dem som består av *flest mulig* kanter?

5 % **19** Du har oppgitt tre sekvenser $A = \langle a_1, \ldots, a_m \rangle$, $B = \langle b_1, \ldots, b_n \rangle$ og $X = \langle x_1, \ldots, x_{m+n} \rangle$ og ønsker å avgjøre om X er en sammenfletting av A og B, dvs., at X består av elementene til A og B, i sin opprinnelige rekkefølge, men flettet i hverandre, så f.eks.

$$X = \langle a_1, a_2, b_1, a_3, b_2, b_3, a_4, \ldots, b_n, a_{m-1}, a_m \rangle \,.$$

Du kan tenke på en slik sammenfletting som å først sette sammen A og B til $\langle a_1, \ldots, a_m, b_1, \ldots, b_n \rangle$ og så (kanskje) endre rekkefølgen på elementene (uten å legge til eller fjerne noen elementer, og der a_i fortsatt kommer før a_{i+1} og b_i fortsatt kommer før b_{i+1}). Det er godt mulig at A og B inneholder noen like verdier, og at disse forekommer flere ganger.

Beskriv en algoritme som løser problemet.

5 % **20** Du har et spillbrett, representert ved en rettet graf $G = (V, E)$, med et sett med k startposisjoner $A \subset V$ og et sett med k sluttposisjoner $Z \subset V$ oppgitt. Du har også k brikker som til å begynne med står i startposisjonene A (én brikke i hver posisjon). Du skal flytte brikkene langs kantene i G slik at det til slutt står én brikke i hver sluttposisjon i Z. (Det spiller ingen rolle hvilken brikke som havner i hvilken sluttposisjon.)

Du skal utføre flyttingen som en serie med maks n trekk. I hvert trekk kan du flytte så mange brikker du vil (samtidig), men de kan maks flytte seg ett hakk hver (altså langs én kant), og du kan ikke flytte to brikker til samme posisjon (node) samtidig. Du kan anta at $k \geqslant 1$ og at det er mulig å finne en løsning.

Beskriv en algoritme som løser problemet.