

# TDT4125 Algoritmekonstruksjon

Eksamen, 27. mai 2021, 09:00–13:00

Faglig kontakt                      Magnus Lie Hetland  
Hjelpemiddelkode                  A

## Oppgaver

- 1 Ford-fulkerson-metoden for bipartitt matching går ut på å finne én og én forøkende sti. Din venn Smartnes mener dette går for langsomt, og mener det vil være bedre å finne flere forøkende stier på én gang. Hva tenker du om planen hennes? Forklar og diskuter.
- 2 Gi en overordnet forklaring, med egne ord, på hvordan den ungarske metoden (KUHN-MUNKRES) fungerer. Hvorfor gir det mening å kalle den en primal–dual-algoritme? Forklar og diskuter.
- 3 Gi en overordnet beskrivelse av LP-relaksering og randomisert avrunding. Hvilke egenskaper har løsningen og hvilke utfordringer støter man gjerne på? Forklar og diskuter.
- 4 Du har gravd ned en skatt i strandkanten, men er usikker på hvor. Du starter på et vilkårlig punkt og begynner å gå frem og tilbake langs stranden med en metalledetektor. For å minimere hvor langt du må gå totalt, så dobler du hvor langt du går fra startpunktet for hver gang du snur og går tilbake.  
  
Tenk på dette som en form for onlinealgoritme, der offlinevarianten innebærer at du vet om skatten ligger mot nord eller sør, og du går i riktig retning fra starten av, til du finner skatten. Hva blir ytelsesgarantien (*competitive ratio*)? Forklar og diskuter.
- 5 Du har oppgitt følgende lineærprogram:

$$\min cx : Ax \geq 1$$

I hver av radene i  $A$  er én av verdiene 1, én av verdiene  $-1$ , og resten 0. Du er bare interessert i å avgjøre om programmet er *gyldig* (*feasible*), og kan dermed anta at  $c = 0$ .

Forklar og diskuter hva dualen til programmet blir, hvordan du kan bruke dualen til å finne ut om primalen er gyldig, og hvordan du kan gjøre det med en velkjent kombinatorisk algoritme.

- 6 Du har oppgitt en vektet rettet asyklisk graf, og skal finne den tyngste delgraf som er en rettet skog, med kanter fra foreldre til barn.
- Hvor bra vil en grådig algoritme fungere?
  - Hvor bra vil en grådig algoritme fungere dersom du skal finne den *letteste* løsningen i stedet?
  - Hvor bra vil en grådig algoritme (for den tyngste løsningen) fungere, dersom du har en vilkårlig rettet (ikke nødvendigvis asyklisk) graf?

Forklar og diskuter.

- 7 Du skal fordele oppgaver på identiske maskiner  $i = 1 \dots m$ , der hver jobb  $j = 1 \dots n$  har prosesseringstid  $p_j$ . Du vil minimere sluttiden,  $C_{\max}$ . Din venn Lurvik har et forslag til løsningsstrategi:

Fordel jobbene utover maskinene, med uniform sannsynlighet. Hver maskin får da forventet sluttid  $t = t_j = \sum_j p_j / m$ , og den forventede sluttiden for hele problemet blir  $\max_j t_j = t$ . Du kan så derandomisere for å få en deterministisk løsning med  $C_{\max} = t$ .

Diskuter løsningen til Lurvik. Argumenter generelt for hvorfor du tror det er mulig eller umulig å lage en algoritme med den garantien Lurvik kommer med, og mer spesifikt for hvorfor du mener Lurviks metode er korrekt eller ikke.

- 8 Du har et sett med ansatte, som hver er ekspert på et sett med temaer. Ekspertisen deres er diversifisert, slik at ethvert par med ansatte har maksimalt ett tema til felles. Du skal konstruere en komité som består av maksimalt  $k$  eksperter som tilsammen dekker alle temaene, eller rapportere at det er umulig å konstruere en slik komité. Beskriv hvordan du effektivt kan redusere en instans til en problemkjerne (*kernel*). For enkelhets skyld kan du anta at du *ikke* er interessert i å begrense antall ansatte. Hvor mange temaer vil kjernen maksimalt bestå av? Forklar og diskuter.