

# Videregående Algoritmik

DIKU, 4 timers skriftlig eksamen, 17. april 2008

David Pisinger, Pawel Winter, Christian Wulff-Nilsen

**Alle hjælpemidler må benyttes dog ikke lommeregner eller computer. Besvarelsen kan udarbejdes med blyant eller kuglepen.**

Opgavesættet består af 19 opgaver, navngivet Q1-Q19. Opgaverne Q1-Q7, Q9 og Q11-Q17 er *multiple-choice opgaver*, som har netop ét korrekt svar. For at besvare en sådan opgave skal man, uden yderligere forklaring, skrive opgavens nummer samt den korrekte svarmulighed. For eksempel kan opgave Q1 besvares med "1A". Opgaverne Q8, Q10, Q18 og Q19 er sædvanlige *tekstopgaver*, som skal besvares tilstrækkeligt detaljeret til at løsningsmetoden kan følges. Hvert korrekt svar til en *multiple-choice opgave* giver 4 point. Hvert korrekt svar til en *tekstopgave* giver 10 point. Man kan samlet opnå 100 point.

## LP på standard form

Lad  $a_{11}, a_{12}, a_{13}, a_{21}, a_{22}, a_{23}, c_1, c_2, c_3, b_1, b_2$  være ikke-negative reelle tal. Hvilket af nedenstående LP problemer (hvis nogen) er på standard form?

Q 1:

- |     |   |     |   |
|-----|---|-----|---|
| 1A) | minimer $c_1x_1 + c_2x_2 + c_3x_3$<br>hvor $a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \leq b_1$<br>$a_{21}x_1 + a_{22}x_2 - a_{23}x_3 \geq -b_2$<br>$x_1 \geq -5, x_2 \geq 6, x_3 \geq 0$ | 1D) | maximer $c_1x_1 + c_2x_2 + c_3x_3$<br>hvor $a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \leq b_1$<br>$a_{21}x_1 + a_{22}x_2 - a_{23}x_3 \leq b_2$   |
| 1B) | maximer $c_1x_1 + c_2x_2 + c_3x_3$<br>hvor $a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \leq b_1$<br>$a_{21}x_1 - a_{22}x_2 - a_{23}x_3 \leq b_2$<br>$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$   | 1E) | minimer $c_1x_1 + c_2x_2 + c_3x_3$<br>hvor $a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \geq -b_1$<br>$a_{21}x_1 - a_{22}x_2 - a_{23}x_3 \geq -b_2$<br>$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$ |
| 1C) | maximer $c_1x_1 + c_2x_2 + c_3x_3$<br>hvor $a_{11}x_1 - a_{12}x_2 + a_{13}x_3 = b_1$<br>$a_{21}x_1 + a_{22}x_2 - a_{23}x_3 \leq b_2$<br>$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$      | 1F) | Ingen af ovenstående  |

■

## Dual til et LP problem

Betragt følgende LP-problem

$$\begin{array}{ll}
\text{maximer} & x_1 + 3x_2 - x_3 \\
\text{hvor} & 2x_1 + 2x_2 - x_3 = 10 \\
& 3x_1 - 2x_2 + x_3 \leq 10 \\
& x_1 - 3x_2 + x_3 \geq 10 \\
& x_1 \geq 0, x_2 \geq 0
\end{array}$$

**Q 2:** Antag at metoden beskrevet i afsnit 29.1 i Cormen benyttes til at omskrive P til et problem på standard form. Hvor mange begrænsninger og variable (bortset fra begrænsninger der tvinger variable til at være ikke-negative) har det duale problem?

- |                                   |                                   |
|-----------------------------------|-----------------------------------|
| 2A) 3 begrænsninger og 3 variable | 2D) 5 begrænsninger og 4 variable |
| 2B) 4 begrænsninger og 3 variable | 2E) 4 begrænsninger og 5 variable |
| 2C) 4 begrænsninger og 4 variable | 2F) Ingen af ovenstående.         |

■

## Lovlige løsninger

Betragt følgende problem P

$$\begin{array}{ll}
\text{maximer} & x_1 + x_2 \\
\text{hvor} & sx_1 + tx_2 \geq -1 \\
& x_1 \geq 0, x_2 \geq 0
\end{array}$$

**Q 3:** For hvilke værdier af  $s$  og  $t$  har P ingen lovlige løsninger?

- |                              |                          |
|------------------------------|--------------------------|
| 3A) $s > 0, t > 0$           | 3D) $s \leq 0, t \leq 0$ |
| 3B) $s \geq 0, t \leq 0$     | 3E) $s \geq 0, t \geq 0$ |
| 3C) $s \geq 0, t$ vilkårligt | 3F) Ingen af ovenstående |

■

## Basisløsninger

Betragt følgende LP problem

$$\begin{array}{ll}
\text{maximer} & 8x_1 + x_2 \\
\text{hvor} & 18x_1 + 10x_2 \leq -4 \\
& 9x_1 + 3x_2 \leq 21 \\
& x_1 \geq 0, x_2 \geq 0
\end{array}$$

**Q 4:** Hvilken af nedenstående løsninger er den allerførste lovlige basisløsning til  $L_{AUX}$  fundet af INITIALIZE\_SIMPLEX?

- |   |   |
|---|---|
| 4A) $x_0 = x_1 = x_2 = x_3 = x_4 = 0$         | 4D) $x_0 = x_1 = x_2 = x_3 = 0, x_4 = 25$ |
| 4B) $x_0 = 4, x_1 = x_2 = x_3 = 0, x_4 = 25$  | 4E) $x_0 = -4, x_1 = x_2 = x_3 = x_4 = 0$ |
| 4C) $x_0 = 4, x_1 = x_2 = x_3 = 0, x_4 = -25$ | 4F) Ingen af ovenstående.                 |

■

## SIMPLEX

Betragt følgende LP problem

$$\begin{aligned} \text{maximer} \quad & -3x_1 + 3x_2 - 6x_3 \\ \text{hvor} \quad & x_1 - 3x_2 + 2x_3 \leq 3 \\ & -8x_1 - 4x_2 + 4x_3 \leq 5 \\ & x_1 + x_2 + 2x_3 \leq 8 \\ & x_1 \geq 0, x_2 \geq 0, x_3 \geq 0 \end{aligned}$$

**Q 5:** Hvilken af nedenstående er en basisløsning efter INITIALIZE\_SIMPLEX og efter den første pivoteringsiteration af SIMPLEX?

5A)  $x_1 = 0, x_2 = 0, x_3 = 0, x_4 = 3, x_5 = 5, x_6 = 8$

5B)  $x_1 = 0, x_2 = 0, x_3 = 0, x_4 = 0, x_5 = 0, x_6 = 0$

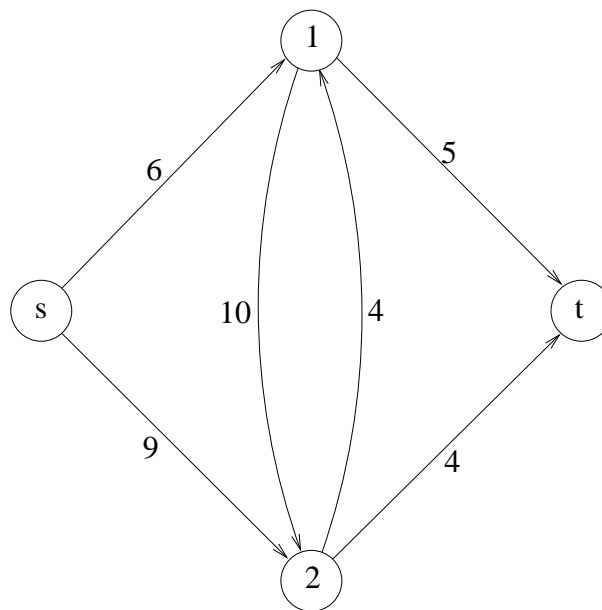
5C)  $x_1 = 0, x_2 = 8, x_3 = 0, x_4 = 25, x_5 = 42, x_6 = 0$

5D)  $x_1 = 0, x_2 = 1, x_3 = 0, x_4 = 6, x_5 = 9, x_6 = 7$

5E)  $x_1 = 0, x_2 = 8, x_3 = 0, x_4 = 27, x_5 = 37, x_6 = 0$

5F) Ingen af ovenstående.

## Edmonds-Karp Algoritme



Figur 1: Netværk  $G$

Figur 1 viser et netværk  $G$  med kilde  $s$  og dræn  $t$ . Ved hver kant er angivet dens kapacitet.

**Q 6:** Hvor mange augmenting paths bestemmes af Edmonds Karp algoritme på netværket i figur 1, før en maksimal strømning er fundet?

6A) 1

6B) 2

6C) 3

6D) 4

6E) 5

6F) 6

## Minimalt snit

**Q 7:** Hvilket snit i  $G$  vist i figur 1 har mindste kapacitet?

7A)  $(\emptyset, \{s,1,2,t\})$

7D)  $(\{s,2\}, \{1,t\})$

7B)  $(\{s\}, \{1,2,t\})$

7E)  $(\{s,1,2\}, \{t\})$

7C)  $(\{s,1\}, \{2,t\})$

7F)  $(\{s,1,2,t\}, \emptyset)$

■

## Push-relabel algoritme

**Q 8: (tekstopgave).** Find en maksimal strømning i grafen  $G$  fra figur 1 ved at håndkøre den generiske push-relabel-algoritme. I besvarelsen er det vigtigt at medtage alle trin i håndkørslen. ■

## Minimum-cost flow problemet

I det generelle minimum-cost flow-problem (GMCFP) er der givet en orienteret graf  $G = (V, E)$ . For en knude  $v \in V$  lader vi  $\delta^+(v)$  betegne alle udgående kanter fra  $v$ , og  $\delta^-(v)$  betegne alle indgående kanter til  $v$ .

- Hver knude  $v \in V$  har et *demand*  $b_v \in \mathbb{R}$ .
- Hver kant  $e \in E$  har en *nedre kapacitet*  $l_e \in \mathbb{R} \cup \{-\infty\}$  og en *øvre kapacitet*  $u_e \in \mathbb{R} \cup \{\infty\}$ .
- Hver kant  $e \in E$  har en *omkostning*  $c_e \in \mathbb{R}$ .

En *b-strømning* (eller *b-flow*) er en tildeling af reelle tal  $x_e$  til hver kant  $e \in E$  således, at

$$\begin{aligned}x_e &\geq l_e, \forall e \in E, \\x_e &\leq u_e, \forall e \in E, \\ \sum_{e \in \delta^-(v)} x_e - \sum_{e \in \delta^+(v)} x_e &= b_v, \forall v \in V.\end{aligned}$$

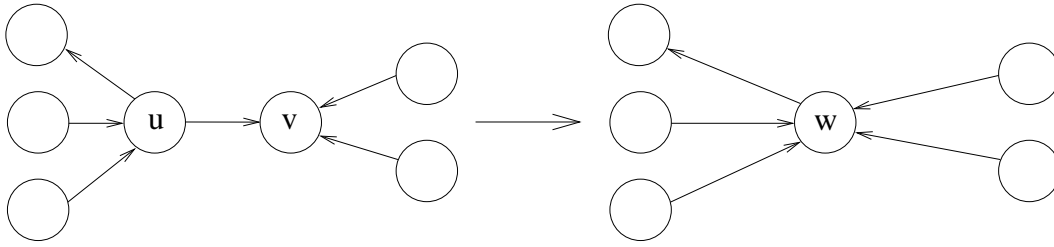
I GMCFP ønskes en *b-strømning* i  $G$  med *minimal* total omkostning over alle kanter, hvor omkostningen af en kant  $e \in E$  er  $c_e x_e$ . Vi vil opfatte en instans af GMCFP som en tupel  $(V, E, b, l, u, c)$  bestående af en graf med demands for knuder og kapaciteter og omkostninger for kanterne.

Betragt en instans  $I = (V, E, b, l, u, c)$  af GMCFP. I denne opgave betragter vi det første reduktionstrin i Opgave 4 fra P1-opgaven. Vi ønsker altså at reducere  $I$  til en instans  $I' = (V', E', b', l', u', c')$  af GMCFP, hvor det for alle kanter  $e \in E'$  gælder, at  $l_e \in \mathbb{R}$  eller  $u_e \in \mathbb{R}$ . Jævnfør P1-opgaven kræves det, at man ud fra en optimal løsning til  $I'$  kan danne en optimal løsning til  $I$ .

Der er flere måder at foretage en sådan reduktion på. En af dem benytter såkaldte *kantsammentrækninger*. Ved en kantsammentrækning fjernes en kant  $(u, v)$ , og endeknuderne  $u$  og  $v$  erstattes af en ny knude  $w$ . Knude  $w$  har som indgående (udgående) kanter alle indgående (udgående) kanter for  $u$  og  $v$ . Et eksempel er vist i figur 2.

Lad  $E_\infty$  være mængden af kanter  $e \in E$ , hvor  $l_e = -\infty$  og  $u_e = \infty$ . Instansen  $I'$  fås ved at foretage kantsammentrækninger af alle kanter  $e \in E_\infty$  samt ved passende at justere demands og omkostninger (kapaciteter justeres ikke). For nemheds skyld antages det, at kantsammentrækninger ikke giver anledning til løkker (kanter med samme start- og slutknude) eller til multiple kanter mellem samme ordnede knudepar.

For  $e \in E_\infty$  lader vi  $v(e) \in V'$  betegne knuden, der opstår ved kantsammentrækning af  $e$ .



Figur 2: Sammentrækning af en kant  $(u, v)$  til en knude  $w$ .

**Q 9:** For en given kant  $e = (u, v) \in E_\infty$ , hvilken af følgende svarmuligheder giver en korrekt specification af demands for  $v(e)$  og omkostninger for ind- og udgående kanter for  $v(e)$ ? (svarene angiver kun de omkostninger, der ændres ift. den oprindelige instans  $I$ )

- |  |  |
|--|--|
| 9A) $b'_{v(e)} = (b_u + b_v)/2,$<br>$\forall e^+ \in \delta^+(u) : c'_{e^+} = c_{e^+} + c_e,$<br>$\forall e^- \in \delta^-(u) : c'_{e^-} = c_{e^-} - c_e.$ | 9D) $b'_{v(e)} = b_u + b_v,$<br>$\forall e^+ \in \delta^+(u) : c'_{e^+} = c_{e^+} + c_e,$<br>$\forall e^- \in \delta^-(u) : c'_{e^-} = c_{e^-} - c_e.$ |
| 9B) $b'_{v(e)} = (b_u + b_v)/2,$<br>$\forall e^+ \in \delta^+(u) : c'_{e^+} = c_{e^+} - c_e,$<br>$\forall e^- \in \delta^-(u) : c'_{e^-} = c_{e^-} + c_e.$ | 9E) $b'_{v(e)} = b_u + b_v,$<br>$\forall e^+ \in \delta^+(u) : c'_{e^+} = c_{e^+} - c_e,$<br>$\forall e^- \in \delta^-(u) : c'_{e^-} = c_{e^-} + c_e.$ |
| 9C) $b'_{v(e)} = (b_u + b_v)/2,$<br>$\forall e^+ \in \delta^+(u) : c'_{e^+} = c_{e^+} + c_e,$<br>$\forall e^- \in \delta^-(u) : c'_{e^-} = c_{e^-} + c_e.$ | 9F) $b'_{v(e)} = b_u + b_v,$<br>$\forall e^+ \in \delta^+(u) : c'_{e^+} = c_{e^+} + c_e,$<br>$\forall e^- \in \delta^-(u) : c'_{e^-} = c_{e^-} + c_e.$ |

■

**Q 10: (tekstopgave).** I besvarelsen af opgaven ovenfor har vi kun defineret demands for visse knuder i  $V'$  og omkostninger for visse kanter i  $E'$ . Hvorledes skal øvrige demands og omkostninger defineres?

Argumenter for hvorfor ovenstående er en reduktion, herunder hvorledes man kommer fra en optimal løsning til  $I'$  til en optimal løsning til  $I$ . ■

## Kvadratisk 0-1 optimering

Det kvadratiske 0-1 optimeringsproblem QP er givet ved

$$\begin{aligned} & \text{maximize} && \sum_{i \in N} \sum_{j \in N} d_{ij} x_i x_j \\ & \text{subject to} && x_j \in \{0, 1\}, \quad j \in N \end{aligned} \tag{1}$$

Hvis alle koefficienter udenfor diagonalen er positive kan problemet løses effektivt ved at transformere det til en instans af MAXIMUM-FLOW med  $V = N \cup \{s, t\}$  og  $E = \{s\} \times N \cup N \times N \cup N \times \{t\}$ . Kapaciteten af kanterne sættes til:

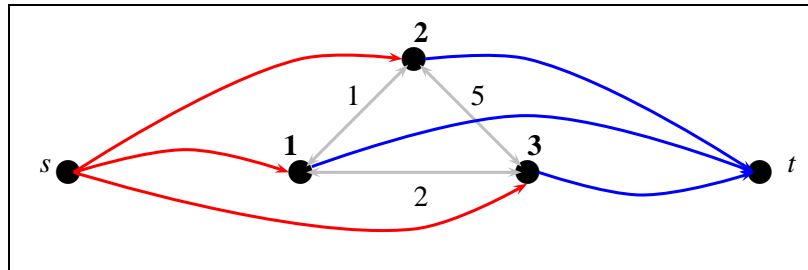
$$\begin{aligned} c_{si} &= \max\{0, \sum_{j \in N} d_{ij}\}, & i \in N & & c_{ij} &= d_{ij}, & i, j \in N, i \neq j \\ c_{it} &= \max\{0, -\sum_{j \in N} d_{ij}\}, & i \in N & & c_{ii} &= 0, & i \in N \end{aligned}$$

Betragt følgende instans med  $N = \{1, 2, 3\}$  hvor  $k \in \mathbb{R}$  er et givet tal:

$i^J$	1	2	3
1	$-k$	1	2
2	1	$4-k$	5
3	2	5	$2-k$

(2)

Hvis vi transformerer den til en instans af MAXIMUM-FLOW fremkommer følgende netværk, som vi betegner  $\mathcal{N}$  (dobbel-rettede kanter kan opfattes som to kanter i hver sin retning med samme kapacitet):



**Q 11:** Bestem værdierne af  $c_{si}$  og  $c_{it}$  idet det antages at negative kantværdier *efterfølgende* bliver sat til nul:

- |  |  |
|--|--|
| 11A) $c_{s1} = 3 - k, c_{s2} = 10 - k, c_{s3} = 9 - k,$<br>$c_{1t} = k - 3, c_{2t} = k - 10, c_{3t} = k - 9$ | 11D) $c_{s1} = k - 3, c_{s2} = k - 10, c_{s3} = k - 9,$<br>$c_{1t} = 3 - k, c_{2t} = 10 - k, c_{3t} = 9 - k$ |
| 11B) $c_{s1} = k, c_{s2} = k, c_{s3} = k,$<br>$c_{1t} = 0, c_{2t} = 0, c_{3t} = 0$                           | 11E) $c_{s1} = 0, c_{s2} = 0, c_{s3} = 0,$<br>$c_{1t} = k, c_{2t} = k, c_{3t} = k$                           |
| 11C) $c_{s1} = -k, c_{s2} = 4 - k, c_{s3} = 2 - k,$<br>$c_{1t} = k, c_{2t} = k - 4, c_{3t} = k - 2$          | 11F) $c_{s1} = k, c_{s2} = k - 4, c_{s3} = k - 2,$<br>$c_{1t} = -k, c_{2t} = 4 - k, c_{3t} = 2 - k$          |

■

**Q 12:** For  $k = 7$ , find et minimalt snit  $(S, T)$  i netværket  $\mathcal{N}$ . Hvad er kapaciteten af dette snit?

- |                    |                    |
|--------------------|--------------------|
| 12A) $c(S, T) = 0$ | 12D) $c(S, T) = 3$ |
| 12B) $c(S, T) = 1$ | 12E) $c(S, T) = 4$ |
| 12C) $c(S, T) = 2$ | 12F) $c(S, T) = 5$ |

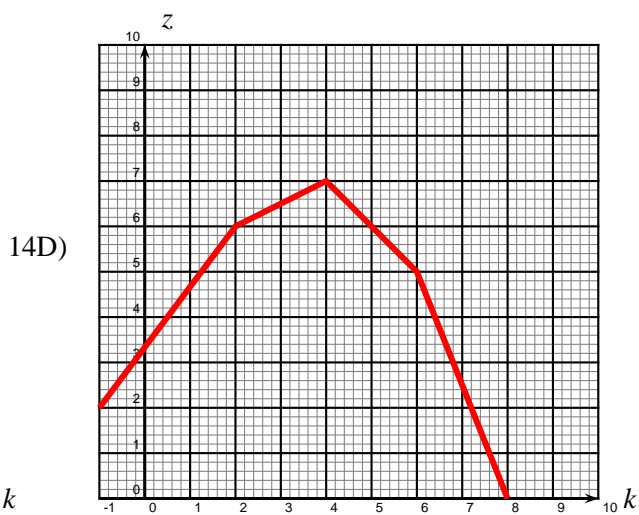
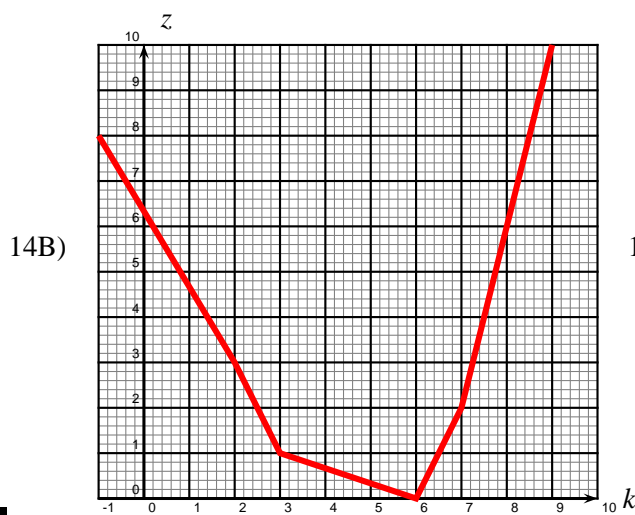
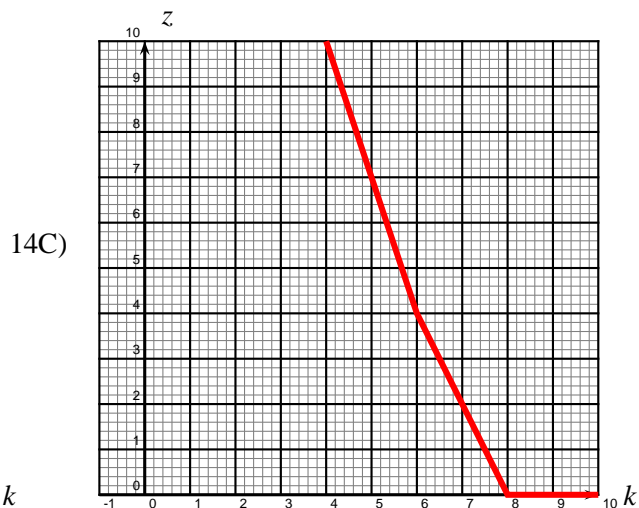
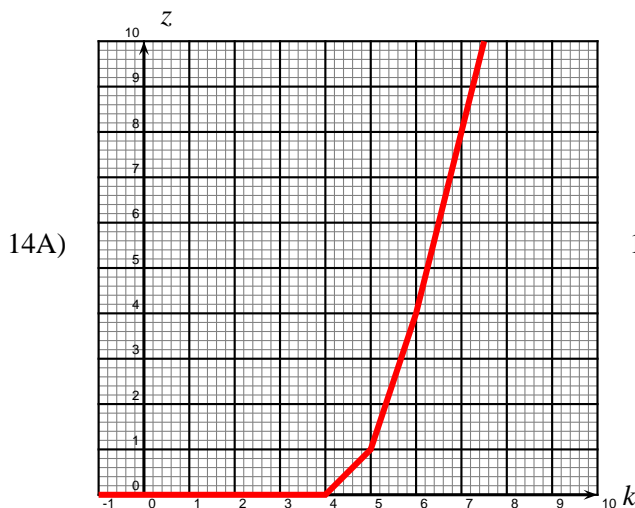
■

**Q 13:** Hvad er den optimale løsning til den tilhørende QP-instans?

- |                                  |                                  |
|----------------------------------|----------------------------------|
| 13A) $x_1 = 0, x_2 = 0, x_3 = 0$ | 13E) $x_1 = 1, x_2 = 0, x_3 = 0$ |
| 13B) $x_1 = 0, x_2 = 0, x_3 = 1$ | 13F) $x_1 = 1, x_2 = 0, x_3 = 1$ |
| 13C) $x_1 = 0, x_2 = 1, x_3 = 0$ | 13G) $x_1 = 1, x_2 = 1, x_3 = 0$ |
| 13D) $x_1 = 0, x_2 = 1, x_3 = 1$ | 13H) $x_1 = 1, x_2 = 1, x_3 = 1$ |

■

**Q 14:** Vi løser nu QP som funktion af  $k$ . Hvilken af følgende figurer svarer til objektfunktionen  $z$



Q 15: Betragt nu følgende kvadratiske knapsack problem med to kapacitets-begrænsninger:

$$\begin{aligned}
 & \text{maximize} && \sum_{i \in N} \sum_{j \in N} p_{ij} x_i x_j \\
 & \text{subject to} && \sum_{j \in N} w_{1j} x_j \leq c_1, \\
 & && \sum_{j \in N} w_{2j} x_j \leq c_2, \\
 & && x_j \in \{0, 1\}, \quad j \in N
 \end{aligned} \tag{3}$$

Hvis vi Lagrange relaxerer kapacitets-begrænsningerne ved brug af multiplikatorer  $\lambda_1, \lambda_2 \geq 0$  fremkommer et problem på formen

$$\begin{aligned}
 & \text{maximize} && \sum_{i \in N} \sum_{j \in N} d_{ij} x_i x_j + \lambda_1 c + \lambda_2 c \\
 & \text{subject to} && x_j \in \{0, 1\}, \quad j \in N
 \end{aligned} \tag{4}$$

Hvad er den korrekte definition af  $d_{ij}$

$$15A) \quad d_{ij} = \begin{cases} p_{ij} & \text{hvis } i = j \\ p_{ij} - \lambda_1 - \lambda_2 & \text{hvis } i \neq j \end{cases}$$

$$15D) \quad d_{ij} = \begin{cases} p_{ij} & \text{hvis } i = j \\ p_{ij} + \lambda_1 + \lambda_2 & \text{hvis } i \neq j \end{cases}$$

$$15B) \quad d_{ij} = \begin{cases} p_{ij} - \lambda_1 - \lambda_2 & \text{hvis } i = j \\ p_{ij} & \text{hvis } i \neq j \end{cases}$$

$$15E) \quad d_{ij} = \begin{cases} p_{ij} & \text{hvis } i = j \\ p_{ij} - \lambda_1 w_{1j} - \lambda_2 w_{2j} & \text{hvis } i \neq j \end{cases}$$

$$15C) \quad d_{ij} = \begin{cases} p_{ij} + \lambda_1 + \lambda_2 & \text{hvis } i = j \\ p_{ij} & \text{hvis } i \neq j \end{cases}$$

$$15F) \quad d_{ij} = \begin{cases} p_{ij} - \lambda_1 w_{1j} - \lambda_2 w_{2j} & \text{hvis } i = j \\ p_{ij} & \text{hvis } i \neq j \end{cases}$$

■

## Kryptering

I det følgende anvendes RSA krypteringssystemet. Din offentlige nøgle er  $e = 7$  og  $n = 15$ . Du modtager en meddelelse et tegn ad gangen. Bogstaver modsvares af tallene ( $A = 1, B = 2$ , etc). Du modtager følgende meddelelse bestående af tre tal:

2      1      3

**Q 16:** Afkod beskeden. Hvad er den afsendte tekst?

16A) FRA

16D) TAK

16B) NEJ

16E) FAR

16C) HAL

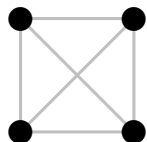
16F) HAN

■

## Approximationsalgoritmer

Givet en ikke-orienteret graf  $G = (V, E)$ . MAX-CUT problemet søger en opdeling af knuderne  $V$  i to dele  $S, T$ , så antal kanter mellem  $S$  og  $T$  er størst mulig. Betragt følgende algoritme MOVE:

- 1 Sæt  $S = \emptyset, T = V$ .
- 2 Hvis en knude  $i$  kan flyttes fra  $S$  til  $T$  eller fra  $T$  til  $S$  så antal kanter over snittet øges, flyttes knuden  $i$ .
- 3 Gentag skridt 2 så længe der findes en knude  $i$  som kan øge antal kanter over snittet.



**Q 17:** Håndkør algoritmen for ovenstående graf  $G = (V, E)$ . Hvad er den fundne løsningsværdi (dvs. antal kanter over snittet)?

17A) 0

17D) 3

17B) 1

17E) 4

17C) 2

17F) 5

■

**Q 18: (tekstopgave)** Bevis at MOVE er en 2-approximations algoritme (for enhver instans). ■

## NP-fuldstændighed

Et vigtigt optimeringsproblem er pakning af rektangler PACKING. Formelt set består inddata af  $n$  rektangler af bredde  $b_i$  og højde  $h_i$  for  $i = 1, \dots, n$ , samt en plade af bredde  $B$  og højde  $H$ . Problemet er nu at afgøre om alle  $n$  rektangler kan anbringes indenfor pladens dimensioner uden at nogen af dem overlapper.

SET-PARTITION problemet tager som input en mængde  $S = \{s_1, \dots, s_n\}$  af tal. Spørgsmålet er om tallene kan opdeles i to mængder  $A$  og  $S \setminus A$  således at

$$\sum_{s_i \in A} s_i = \sum_{s_i \in S \setminus A} s_i$$

I Cormen, opgave opgave 34.5-5 vises det at SET-PARTITION er NP-fuldstændigt.

**Q 19: (tekstopgave)**

- Vis at PACKING ligger i klassen NP.
- Vis at PACKING er NP-hårdt ved reduktion fra SET-PARTITION.

■

## Vejledende svar

**S 1** 1B ■

**S 2** 2C ■

**S 3** 3F ■

**S 4** 4B ■

**S 5** 5E ■

**S 6** 6B ■

**S 7** 7E ■

**S 8** ■

**S 9** Svarmulighed E: den  $b$ -strømning, der samlet strømmer ind i (ud af)  $u$  og  $v$  i  $I$ , skal strømme ind i (ud af)  $v(e)$  i  $I'$ . Derfor sættes  $b'_{v(e)} = b_u + b_v$ . Lad os i første omgang antage, at  $u$  har demand 0. For en  $b'$ -strømning i  $I'$  må forskellen mellem det, der strømmer ind i  $v(e)$  gennem kanter til  $u$  og det, der strømmer ud fra  $v(e)$  gennem kanter fra  $u$  være det, der skal strømme gennem  $e$  i  $I$ . Ændringerne i omkostningerne i svarmulighed E betaler netop for denne forskel. Hvis  $u$  ikke har demand 0, ses det, at forskellen i omkostninger mellem en optimal løsning til  $I$  og en optimal løsning til  $I'$  kun afviger med en konstant (kun afhængig af demands). ■

**S 10** Øvrige demands og omkostninger justeres ikke.

Argumentet for at der er tale om en reduktion er delvist indeholdt i løsningen til opgaven ovenfor. Bemærk desuden at man altid kan komme fra en strømning til  $I'$  til en tilsvarende strømning til  $I$  (idet strømmingen kan vælges frit i de kanter i  $E$ , der sammentrækkes). Omkostningen af disse to strømninger afviger med en konstant.

Givet en optimal løsning til  $I'$ , da fås derfor en optimal strømning til  $I$  ved at bestemme den entydige strømning gennem hver kant i  $E_\infty$ , der tillades af strømmingen i  $I'$  (dette er altid muligt jvf. ovenstående). ■

**S 11** 11A) ■

**S 12** 12D) ■

**S 13** 13D) ■

**S 14** 14C) ■

**S 15** 15F) ■

**S 16**  $n = 3 \cdot 5$ , dvs  $\phi(n) = 2 \cdot 4 = 8$ . Den multiplikativt inverse af  $e = 7$  er  $d = 7$  da  $7 \cdot 7 = 49 = 1 \pmod{8}$ . Vi finder nu at

$$2^7 \pmod{15} = 128 \pmod{15} = 8 \pmod{15} \quad (H)$$

$$1^7 \pmod{15} = 1 \pmod{15} \quad (A)$$

$$3^7 \pmod{15} = 2187 \pmod{15} = 12 \pmod{15} \quad (L)$$

16C) ■

**S 17** Da grafen er symmetrisk er det lige meget hvilken knude man starter med. Den fundne løsning er 4. 17E) ■

**S 18** Bemærk først at MAX-CUT er et maximeringsproblem, dvs. skal vise  $C/C^* \geq 2$ .

Når MOVE standser, vil der for enhver knude  $i \in V$  gælde at mindst halvdelen af dens kanter ligger over snittet (ellers kunne vi blot flytte knuden til den modsatte mængde og opnå en bedre objektfunktion, hvilket strider mod stopkriteriet i MOVE). Den fundne approximationsløsning  $C$  må derfor opfylde at  $C \geq |E|/2$ .

Den optimale løsning  $C^*$  kan højst indeholde alle kanter, dvs  $C^* \leq |E|$ . Dermed har vi at  $C/C^* \geq 2$ . ■

**S 19** a) Lad  $(x_i, y_i)$  angive koordinater for det nedre venstre hjørne af hvert rektangel. Et muligt certifikat er således  $(x_i, y_i)$  for  $i = 1, \dots, n$ .

Man kan i polynomiel tid kontrollere at en foreslået løsning er korrekt. At hvert rektangel er indenfor pladens dimensioner kan kontrolleres med:

$$0 \leq x_i \leq B - b_i \quad 0 \leq y_i \leq H - h_i$$

At to rektangler  $i, j$  ikke overlapper kan kontrolleres med:

$$x_i + b_i \leq x_j \vee x_j + b_j \leq x_i \vee y_i + h_i \leq y_j \vee y_j + h_j \leq y_i$$

Det tager  $O(n^2)$  at kontrollere ovenstående.

b) Givet en instans  $S$  af SET-PARTITION. Konstruer en instans af PACKING ved for hvert  $s_i$  at konstruere et rektangel med bredde  $b_i := s_i$ , højde  $h_i := 1$ . Pladens størrelse sættes til  $B = \sum_{s_i \in S} s_i/2$  og  $H = 2$ .

Hvis SET-PARTITION har en lovlig løsning  $A$ , så konstrueres en lovlig løsning til PACKING ved at tildele rektanglerne i  $A$   $y$ -koordinat 0 mens rektanglerne i  $S \setminus A$  får  $y$ -koordinat 1.

Omvendt, antag at PACKING har en lovlig løsning. For at konstruere en lovlig løsning til SET-PARTITION sætter vi  $A$  til de rektangler, der har  $y$ -koordinat 0.

Da de to instanser således er ækvivalente, og transformationen kan foretages i polynomiel tid har vi at PACKING er NP-fuldstændig. ■