Written exam
Introduction to linear and Integer programming (DM515)

Institut for Matematik & Datalogi
Syddansk Universitet

Monday, June 6, 2011, kl. 9–13

ENGLISH preamble:

All usual supporting material (books, notes, weekly notes etc) as well as use of pocket calculator is allowed. It is not allowed to use a computer. There are 6 exam problems whose relative weight is given i percentages. **Remember to justify all claims you make!**

You may refer to results from the book(s) or notes handed out during the course. In particular it is OK to justify a claim by giving a reference to a result in the teaching material from which it follows (provided this is true and easy to see!). You may apply methods or extensions of theorems which have been derived via exercises posed during the course. **Note, however that it is not allowed to answer a question by saying that it follows from one of the exercises unless that exercise concerns a general method (e.g. Gomory cuts). References to other books or material (not on the pensum list) will not be accepted as a correct answer!**

Dansk forord:

Alle sædvanlige hjælpemidler (lærebøger, notater etc.) samt brug af lommeregner er tilladt. Det er ikke tilladt at anvende en computer. Eksamenssættet består af 6 opgaver. De enkelte opgavers vægt ved bedømmelsen er angivet i procent. **Husk at begrunde alle dine påstande!**.

Der må gerne refereres til resultater fra lærebogen og de øvrige noter fra pensum. Specielt må man gerne begrunde en påstand med at henvise til, at den umiddelbart følger fra et resultat i lærebogen (hvis dette altså er sandt!). I må gerne bruge metoder eller udvidelser af sætninger som er udledt i opgaver, der er stillet i løbet af kurset **Bemærk dog, at det ikke er tilladt at besvare et delspørgsmål, udelukkende med en henvisning til, at det følger af en af opgaverne med mindre denne opgave omhandler**

en generel metode (som f.eks. Gomory Cuts). Henvisninger til andre bøger (ud over pensum) accepteres ikke som besvarelse af et spørgsmål!

PROBLEM 1 Relaxation, the simplex method and the dual problem (20 %)

Consider the following integer programming problem (IP):

$$\begin{array}{lll} \text{maximize} & 3x_1 - x_2 + 2x_3 & \\ \text{subject to} & x_1 - x_2 + x_3 & \leq 5 \\ & 2x_2 + x_3 & \leq 4 \\ & x_1 & \leq 3 \\ & x_1, x_2, x_3 & \geq 0,\ x_1, x_2, x_3 \in \mathbb{Z} \end{array}$$

Question a:

Write the LP relaxation (P1) of (IP) and explain why the objective value of an optimal solution to (P1) is an upper bound on the value of an optimal solution to (IP).

Question b:

Convert the problem (P1) to equational form by adding slack variables x_4, x_5, x_6 corresponding to the three in-equalities in the order from top to bottom. Next write the first simplex tableau with x_4, x_5, x_6 as the basic solution.

Question c:

Explain how x_1 may be brought into the basic solution and why this will increase the current objective value. Perform a pivot step that brings x_1 into the basis and explain how you select the variable to leave the basis in that step.

Question d:

After two more pivot steps (you do not have to perform these!) we obtain the following simplex tableau:

$$\begin{array}{rcl} x_1 & = & 3 - x_6 \\ x_2 & = & \frac{2}{3} + \frac{1}{3}x_4 - \frac{1}{3}x_5 - \frac{1}{3}x_6 \\ x_3 & = & \frac{8}{3} - \frac{2}{3}x_4 - \frac{1}{3}x_5 + \frac{2}{3}x_6 \\ \hline z & = & 13\frac{2}{3} - \frac{5}{3}x_4 - \frac{1}{3}x_5 - \frac{4}{3}x_6 \end{array}$$

Argue that we have found an optimal solution to (P1). State the solution and its objective value.

Question e:

Write up the dual problem (DP1) of (P1) where you use dual variables y_1, y_2, y_3 corresponding to the three in-equalities in (P1) from top to bottom.

Question f:

Show that $(y_1, y_2, y_3) = (\frac{5}{3}, \frac{1}{3}, \frac{4}{3})$ is an optimal solution to (DP1).

Question g:

As indicated in Question d, the optimal solution to (P1) is not an integer solution and hence not a solution to (IP). Use the last simplex tableau to derive the following Gomory cut (these cuts were introduced on weekly note 3):

$$\frac{2}{3}x_4 + \frac{1}{3}x_5 + \frac{1}{3}x_6 \geq \frac{2}{3} \tag{1}$$

Give a short explanation why this is a valid in-equality for (IP) (when we think of the slack variables being integer variables added to the original formulation), while adding (1) to (P1) will make the current optimal LP solution infeasible.

PROBLEM 2 Flows(20 %)

At the annual party at a well-known academic institution on Fyn not all went smoothly and some participants had to be taken to medical emergency treatment at Odense University Hospital. In total 150 had to get a transfusion of one bag of blood. The hospital had 155 bags in stock. The distribution of blood groups in the supply and amongst the participants in need on blood is shown in the table below (the last part is written as demand).

Blood type	A	B	0	AB
Bags in stock	44	31	42	38
Demand	37	33	40	40

- Type A patients can only receive blood of type A or type 0.
- Type B patients can receive only type B or type 0.
- Type 0 patients can receive only type 0.
- Type AB patients can receive any of the four types.

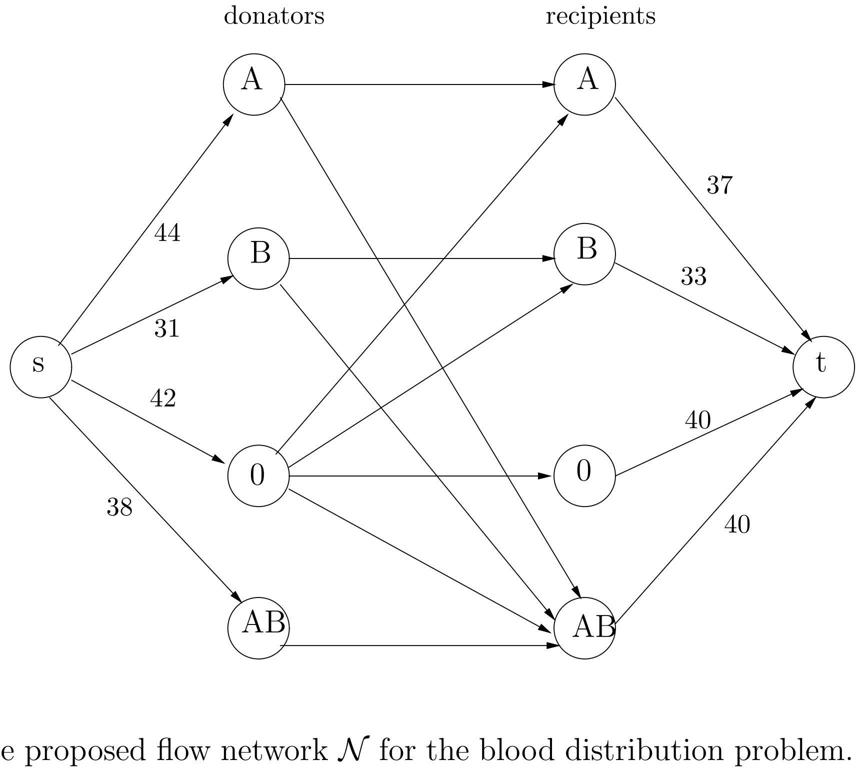

Figure 1: The proposed flow network $\mathcal{N}$ for the blood distribution problem. All capacities on the arcs in the middle are infinite.

Question a:

Consider the flow network $\mathcal{N}$ in Figure 1. Show that one can use this network to model the problem of checking whether it is possible to give all participants a blood transfusion that is compatible with the restrictions above. In particular, you should

- Explain the meaning of the arcs in the middle (what do they model?).
- Say what flow value ($b_x(s)$) you are looking for and how to interpret an (s,t)-flow of that value as a proper assignment of blood bags.
- You should also give an argument that we may replace all infinite capacities by finite numbers and say which (smallest) numbers will work.

Question b:

Show how to find a maximum (s,t)-flow in $\mathcal{N}$ by the augmenting path method (Ford-Fulkerson or shortest augmenting paths). You may start by listing four easy to find (s,t)-paths and send flow along these. After this you should show the resulting flow x and the residual network $\mathcal{N}(x)$ and show how to continue from there.

Question c:

Show a feasible assignment of blood bags that is found via your flow algorithm above and say how you obtained it from your maximum (s,t)-flow.

Question d:

Suppose now that five more participants show up also in need of a blood transfusion (so now all blood bags of the hospital would be needed if there is a solution). Use the network representation to analyse what their blood types must be in order for a feasible solution to exist.

PROBLEM 3 Project scheduling (15 %)

A small project has 6 sub-activities A, B, C, D, E, F whose individual dependency (shown by the immediate predecessors) is given in Figure 2. Here we also list the normal time (in weeks), the absolute minimum time and the cost of shortening the activity by one week.

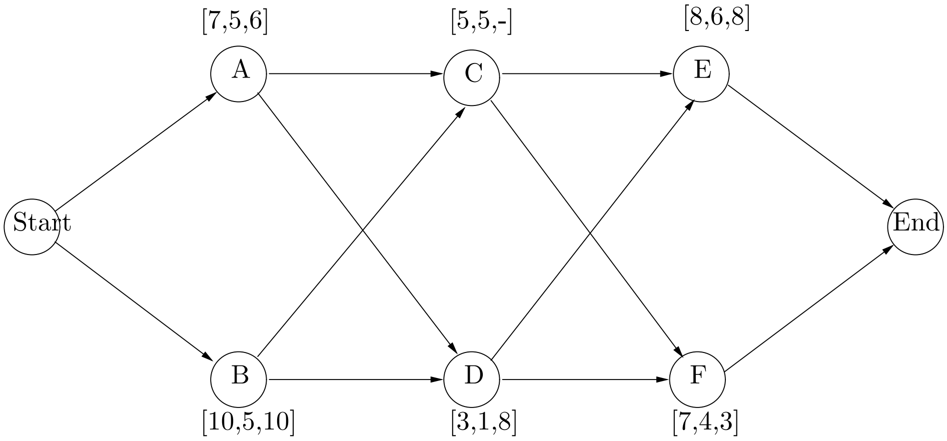

Figure 2: An AON network for a small project with 6 activities. For each activity the following data is given in that order from left to right: normal time, minimum time in weeks, and the cost of shortening the duration of the activity by one week.

Question a:

Describe (in words) an algorithm for finding the duration of a given project when it is modelled as an AON network and all activities are at the normal duration. What is the running time of the algorithm?

Question b:

Illustrate your algorithm on the project network in Figure 2 and state the duration of the project found by the algorithm. It suffices to show a few steps and then write the solution. You must also show the result of the same calculation when all activities are at the minimum duration. Here it is enough to show the final result (you may also show both calculations on one graph).

Question c:

The goal is now to shorten the duration of the project to 19 weeks. This means that the duration of one or more activities has to be shortened. Of course we want to select these so that the total cost of shortening the duration to 19 weeks is minimized. Formulate this problem a linear programming problem and argue that the optimal solution to this LP will provide the correct answer. Note that you must use the actual data in the LP formulation!

PROBLEM 4 Branch and Bound (15 %)

Question a:

Consider the following knapsack problem:

$$\begin{array}{rl} \max & 50x_1 + 60x_2 + 140x_3 + 40x_4 \\ \text{such that} & 5x_1 + 10x_2 + 20x_3 + 20x_4 \leq 30 \\ & x_1, x_2, x_3, x_4 \in \{0, 1\} \end{array}$$

State the LP-relaxation and show how to find an optimal solution to this without using the simplex method. Hint: a greedy approach works.

Question b:

Solve the knapsack problem above to optimality using branch and bound. Use the depth first search strategy when exploring the nodes of the B&B tree and when you branch, you should first explore the branch corresponding to including the item that you are branching on. Show the B&B tree in each step and explain briefly (with justification) what you conclude in each step of the algorithm. This includes which nodes in the B&B tree you must continue exploring and which nodes you can finish (and with what conclusion).

PROBLEM 5 Formulation of IP problems and the cutting plane method (20 %)

Let $G = (V, E)$ be a connected graph on n vertices $V = \{1, 2, \ldots, n\}$ and non-negative weights w_e on its edges. The fact that we can solve the minimum spanning tree problem in polynomial time is ignored in this problem.

Question a:

Explain why the following is a correct integer programming formulation of the minimum spanning tree problem and state the LP-relaxation:

$$\begin{aligned}
\text{Minimize} \quad & \textstyle\sum_{e\in E} w_e x_e && (2)\\
\text{such that} \quad & \textstyle\sum_{e\in E} x_e = n-1 && (3)\\
\textstyle\sum_{\{e:|e\cap S|=1\}} & x_e \geq 1 \ \forall \ \emptyset \neq S \subseteq V - \{1\} && (4)\\
& x_e \in \{0,1\} && (5)
\end{aligned}$$

For later reference we call this formulation MSTIP and its LP-relaxation MSTLP. The conditions (4) are not nice to work with, since there are exponentially many of those, so even MSTLP may be impossible to solve when n gets large.

Consider the following integer programming problem MST-NO-CUTS:

$$\begin{aligned}
\text{Minimize} \quad & \textstyle\sum_{e\in E} w_e x_e && (6)\\
\text{such that} \quad & \textstyle\sum_{e\in E} x_e = n-1 && (7)\\
& x_e \in \{0,1\} && (8)
\end{aligned}$$

Question b:

Explain how to solve the LP-relaxation of MST-NO-CUTS. Why does this linear programming problem always have an optimal integer solution x^*?

Question c:

What can you say about the objective value of the optimum solution x^* to MST-NO-CUTS compared to the optimal solution to MSTIP (the minimum spanning tree solution)?

Question d:

How do you check (algorithmically and in polynomial time!) whether x^* above is also a feasible solution to MSTIP? What can you conclude if it is a feasible solution to MSTIP?

Question e:

Suppose now that x^* is a not feasible solution to MSTLP, that is, the condition (4) is violated for some X. Explain how to find at least one cut $(X, V - X)$ with $\emptyset \neq X \subseteq V - \{1\}$ such that adding the condition $\sum_{\{e:|e \cap X|=1\}} x_e \geq 1$ to the LP relaxation of the MST-NO-CUTS formulation makes x^* infeasible for this extended problem.

Suppose that we have iterated this process a number of times and that y^* is an optimum solution to the current LP problem (obtained from the LP-relaxation of MST-NO-CUTS by adding the violated cuts found so far).

Question f:

Explain briefly how to use a maximum flow algorithm (as a subroutine) to check whether y^* satisfies (4).

PROBLEM 6 Maximum weight matchings in bipartite graphs (10 %)

Consider the maximum weight matching problem in a bipartite graph $G = (V, E)$ with a non-negative weight function w on the edges, as described on page 33 in MG:

$$\begin{array}{rl} \text{Maximize} & \sum_{e\in E} w_e x_e \\ \text{such that} & \sum_{\{e\in E: v\in e\}} x_e = 1 \;\; \forall v \in V \\ & x_e \in \{0,1\} \;\; \forall e \in E \end{array} \tag{9}$$

Suppose that we are given an optimal solution x^* to the LP-relaxation of (9) for a bipartite graph G on 6 vertices such that for some 6-cycle $abcdefa$ of G x^* has the values $x^*_{ab} = x^*_{cd} = x^*_{ef} = \frac{1}{3}$ and $x^*_{bc} = x^*_{de} = x^*_{af} = \frac{2}{3}$ (see Figure 3).

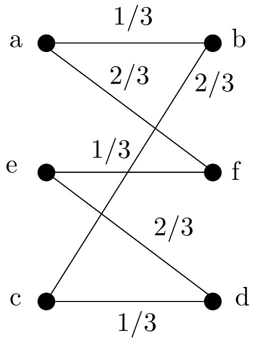

Figure 3:

Question a:

Prove (without knowing what the cost of the edges are!) that the edges af, de, bc form a maximum weight matching in G. Hint: Consider the proof of Theorem 3.2.1 in MG.

Question b:

Can you give a different maximum weight matching?