Technische Universität Berlin
Institut für Mathematik

Dr. Rüdiger Stephan

Mathematical Tools for Engineering and Management
Winter Term 2011/2012

January 04, 2012

Exercise sheet 9

The due date of the graded homework (Exercise 3) is January 18, 2012. Please send your AIMMS model file (suffix `.amb`) by email to `stephan@math.tu-berlin.de`.

Exercise 1

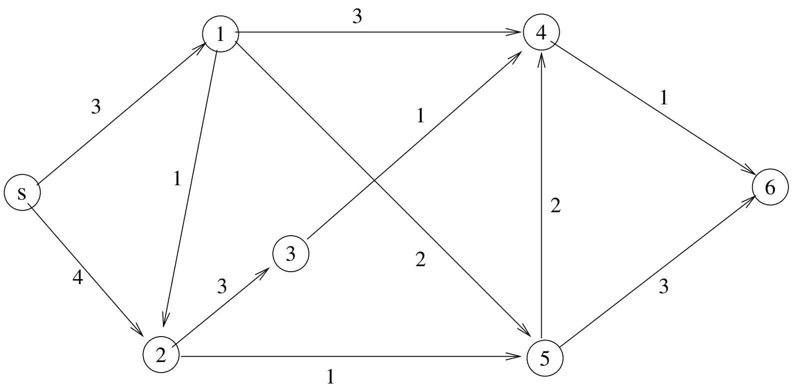

Figure 1: Digraph with nonnegative arc weights.

a) Compute shortest paths from s to any other node using Dijkstra's algorithm, and draw a shortest-path tree with root s.

b) Is the shortest-path tree in a) unique?

c) Change the weight of the arc $(3, 4)$ to -2. Does Dijkstra's algorithm working correctly under this modification of the graph?

Exercise 2

Let be given a simplified street network of a city where all streets run parallel to x- or y-axis. A nonegative weight is associated with each street of the network to represent the cost for driving on this street. Unfortunately this city has no traffic lights, and hence, taking a left turn is really a disaster. The additional costs for turning left at crossroads are assumed to be constant α.

a) Describe an algorithm that solves the shortest-path problem with turning left costs as precised in the introductory paragraph.
Hint: Create a new graph G^* that has a node for each arc (i, j) of the network, and join nodes of the form (i, j) and (j, k).

b) Use this algorithm to solve the instance given by Figure 2 where $\alpha = 5$ and starting node 1.

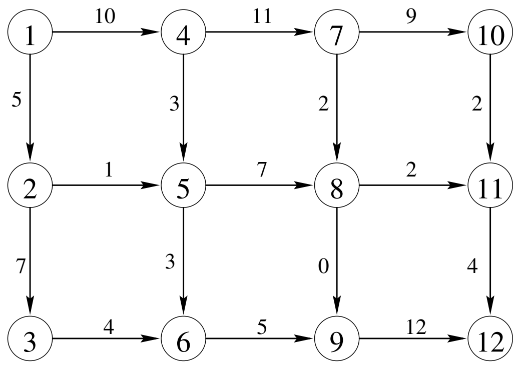

Figure 2: City map of Little-Manhattan.

Graded Homework

Exercise 3

Consider the *Binary Knapsack problem.* Imagine you are a burglar and want to maximize the money you get for your stolen goods. You have a number of items to choose from and each item has a certain weight and value. Your knapsack can handle only so much weight and you want to maximize the value you get for the stolen items.

- Suppose you have a knapsack which can handle 30kg and you have the following items available:

Item	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Weight	5	8	10	2	4	1	8	10	2	8	7	5	4	5	10
Value	3	3	10	5	2	4	2	9	2	5	3	9	10	3	10

Figure 3: Items for Knapsack

Determine an optimal burglaring policy (state which elements you should take and the value you get). [3 points]

- Formulate this problem as an integer program. [3 points]
- Consider another variant of the problem, which is called the *Bin-Packing Problem.* In this problem, you have a number of items to choose from and bins which all have the same capacity. The items have a certain weight. You have to pack all the items into the bins and minimize the number of bins that you use while doing that. Items cannot be splitted, that is, every item can only be packed into one bin. Formulate this problem as an Integer Program. In the formulation, you may assume that the weight of any single item doesn't exceed the capacity of an empty bin. [3 points]
- Solve the following BIN PACKING instance where each bin has a capacity of 50:

Item	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Weight	5	10	15	20	30	35	40	30	20	15	11	30	35	40	43

Item	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Weight	2	31	22	28	41	20	48	21	29	23	14	24	12	17	32

Item	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
Weight	30	4	46	23	2	45	33	48	25	19	21	35	44	7	27

Figure 4: Items for Bin Packing

How many bins do you need? Which element goes to which bin? [5 points]