Optimization
Summer 2010

Homework 12

N. Megow, K. Mehlhorn
J. Mestre

This assignment is **due on July 15/16** in your tutorial session. You are allowed (even encouraged) to discuss these problems with your fellow classmates. All submitted work, however, must be *written individually* without consulting someone else's solutions or any other source like the web.

Problem 1: Solve the following Cutting Stock problem by using one of the algorithms presented in the lecture.
The width of the rod is 100 inches with the following order details:

1. 3 raws of width 60 in.
2. 5 raws of width 30 in.
3. 9 raws of width 35 in.

Problem 2: Consider the following version of the knapsack problem:
A set of n items, each of non-negative weight w_j are given and a nonnegative profit c_j is associated with them. We are allowed to pick **several copies** of the same item and the objective is to maximize the profit subject to the constraint that the total weight being upper bounded by a constant K. Modify the dynamic programming solution for the Knapsack problem shown in the lecture to solve the problem.

Problem 3: In the lecture, the dynamic programming solution to the knapsack problem requires space $O(n \cdot K)$ where n is the number of items and K is the capacity of the knapsack. As we observed in the lecture, the value of knapsack can be computed using $O(K)$ space without increasing the running time. Now we also want to compute the solution set. Propose a method with space complexity $O(\sqrt{n} \cdot K)$ and time complexity $O(n \cdot K)$. (**Hint**: Store a carefully chosen $\sqrt{n}$ rows intially and recompute the other rows when necessary while maintaining the space complexity.).