CS 170 Algorithms
Fall 2012 Satish Rao
HW 8

Due November 1, 6:00pm

1. **(20 pts.) Problem 6.4 (Corrupted text document)**

2. **(20 pts.) Problem 6.14 (Cutting cloth)**

3. **(20 pts.) Problem 6.20 (Optimal binary search tree)**

4. **(20 pts.) Problem 6.29 (Exon chaining)**

5. **(20 pts.) Timesheets Part 2**

   Recall problem 4 from homework 7.

   Suppose we have N jobs labelled $1,\ldots,N$. For each job, you have determined the bonus of completing the job, $V_i \geq 0$, a penalty per day that you accumulate for not doing the job, $P_i \geq 0$, and the days required for you to successfully complete the job $R_i > 0$.

   Every day, we choose one unfinished job to work on. A job i has been finished if we have spent R_i days working on it. This doesn't necessarily mean you have to spend R_i contiguous sequence of days working on job i. We start on day 1, and we want to complete all our jobs and finish with maximum reward. If we finish job i at the end of day t, we will get reward $V_i - t \cdot P_i$. Note, this value can be negative if you choose to delay a job for too long.

   Now, what we did not tell you last time is that we have a time limit of T days, in which we can choose to work on some of these jobs in only those T days. Given this information, what is the optimal job scheduling policy with a time limit of T days? Notice that 0 is a lower bound since we can choose to do no jobs at all if all of them happen to have negative value, or all of them take more than time T.