$22\mathbf{C} : 231$ ($CS : 5350 : 0001$) **Design and Analysis of Algorithms**
Homework 2

The homework has two types of problems – reinforcement problems and regular problems. For the reinforcement problems, we are not concerned with originality in coming up with the solution, but rather with how well you write up the solution. You can get help in coming up with the solution – from friends, online, etc. – but understand the solution and explain it in your own words. For the regular problems, the only type of help you can get is collaboration with classmates, and discussion with the instructor or TA. No record or notes, electronic or written, should be taken from such collaborations. For these problems we do care about originality in coming up with the solution.

The homework is worth 10 points, with each question being worth 2 points. The theme is dynamic programming. The homework is due in class on Tuesday, Feb 14.

Reinforcement Problems

- Exercise 2 of Chapter 6.
- Exercise 6 of Chapter 6. (One way to do this is to reduce to the segmentation problem.)
- Exercise 16 of Chapter 6.

Regular Problems

- We are given a set $\{a_1, a_2, \ldots, a_n\}$ of n positive numbers, and we have to determine if the set can be *partitioned* into three subsets, so that the *sum* of the elements in each of the subsets is the same. Give an algorithm for this whose running time is polynomial in n and W, where $W = \max_{1 \le i \le n} a_i$. (You may need to understand our algorithm for the Knapsack problem before attempting this.)
- We are given a set $\{R_1, R_2, \ldots, R_n\}$ of rectangles. A rectangle is specified by four numbers a, b, c, d: they determine the rectangle $\{(x, y) \in \Re^2 \mid a \le x \le b,\ c \le y \le d\}$. Given rectangles R and R', we can determine in constant time if $R \subseteq R'$, that is, if R is contained in R'. The goal in the problem is to find a maximum *cardinality* subset $\{i_1, i_2, \ldots, i_m\}$ of $\{1, 2, \ldots, n\}$ such that $R_{i_1} \subseteq R_{i_2} \subseteq \cdots R_{i_{m-1}} \subseteq R_{i_m}$. (You can assume that the rectangles are ordered so that $R_i \subseteq R_j \implies i \le j$.)