

22C : 231 (CS : 5350 : 0001) **Design and Analysis of Algorithms**  
**Homework 1**

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 2.5 points. The theme is the use of recursion as a problem solving tool, with divide-and-conquer being a special case of this. The homework is due in class on Thursday, Feb 7th.

**Reinforcement Problems**

1. Exercise 2 of Chapter 5.
2. Exercise 3 of Chapter 5.

**Regular Problems**

1. Consider a  $2^n \times 2^n$  chessboard (for any  $n \geq 1$ ) with one (arbitrarily chosen) square removed. Prove that any such chessboard can be tiled without gaps or overlaps by L-shaped pieces, each composed of three squares.
2. Suppose you are given a stack of  $n$  pancakes of different sizes. You want to sort the pancakes so that smaller pancakes are on top of larger pancakes. The only operation you can perform is a *flip* - insert a spatula under the top  $k$  pancakes, for some integer  $k$  between 1 and  $n$  of your choice, and flip them all over. See Figure on next page. Describe an algorithm to sort an arbitrary stack of  $n$  pancakes using  $O(n)$  flips.



Figure 1: Flipping the top 3 pancakes