CSCI 533 Analysis of Algorithms
Homework #5
Due Tuesday, October 31st at the beginning of class. No late assignments accepted.

The first two problems are worth 20 points each, and the third is worth 10 points.

1. Coin changing [CLRS Problem 61-1, p. 402]

   Consider the problem of making change for \( n \) cents using the fewest number of quarters, dimes, nickels, and pennies. Assume that each coin’s value is an integer.

   (a) Describe a greedy algorithm to make change consisting of quarters, dimes, nickels, and pennies. Prove that your algorithm yields an optimal solution.

   (b) Suppose that the available coins are in the denominations that are powers of \( c \), i.e., the denominations are \( c^0, c^1, \ldots, c^k \) for some integers \( c > 1 \) and \( k \geq 1 \). Show that the greedy algorithm always yields an optimal solution.

   (c) Give a set of coin denominations for which the greedy algorithm does not yield an optimal solution. Your set should include a penny so that there is a solution for every value of \( n \).

   (d) Give an \( O(nk) \)-time algorithm that makes change for any set of \( k \) different coin denominations, assuming that one of the coins is a penny.

2. A sequence of \( n \) operations is performed on a data structure. The \( i \)th operation costs \( i \) if \( i \) is a power of 2, and 1 otherwise. Use each of the following methods of analysis to determine the amortized cost per operation.

   (a) Aggregate method [CLRS Exercise 17.1-3, p. 410]

   (b) Accounting method [CLRS Exercise 17.2-2, p. 412]

   (c) Potential method [CLRS Exercise 17.3-2, p. 416]

3. Show that if all \( \text{Unions} \) precede the \( \text{Finds} \), then the disjoint set algorithm with path compression requires linear time (in the number of operations). Give a charging scheme that will show this.