CS 660, HOMEWORK 3 (DUE: SUNDAY OCTOBER 13, 11:59 PM)

INSTRUCTOR: HOA VU

Each question is worth 27.5 points. There is no extra credit problem for this homework, but each problem is worth 2.5 more points. If you perfect this homework, then it is equivalent to getting the full 10 extra credits in other homework.

When you are asked to design an algorithm, do the following: a) describe the algorithm, b) explain (or more rigorously prove) why it is correct, and c) provide the running time.

Unless specified otherwise, the problems are from the textbook by Jeff Erickson that we use.

For dynamic programming questions. Make sure a) you define the dynamic programming table, b) what is the recursive structure (how to fill an entry of the table), c) boundary cases & the filling order, d) how to get the answer from the table, and e) state the running time.

(1) **Question 1:**
   - a) Show that $1^5 + 2^5 + 3^5 + \ldots + n^5 = \Theta(n^6)$. Hint:
     First, show that $1^5 + 2^5 + 3^5 + \ldots + n^5 = O(n^6)$ by showing that $1^5 + 2^5 + 3^5 + \ldots + n^5 \leq n^6$.
     Next, show that $n^6 = O(1^5 + 2^5 + 3^5 + \ldots + n^5)$. Use the following fact, if $f(x)$ is an increasing function. Then,
     $$\sum_{k=1}^{n} f(k) \geq \int_0^n f(k)dk \ .$$
   - Prove via induction that for every natural number n, we have that $n^3 + 2n$ is divisible by 3.

(2) **Question 2:** Given a value K, if we want to make change for K cents, and we have infinite supply of each of $\{S_1, S_2, .., S_m\}$ denominations, how many ways can we make the change? The order of coins doesn't matter. Hint: Use dynamic programming. Define $T[i, j]$ be the number of ways to make change for i cents using coins of values $\{S_1, \ldots, S_j\}$.

(3) **Question 3:** Problem 33 page 141.

(4) **Question 4:** Problem 14a and 14b page 180.