Assignment 5

Nilanjan Datta

IAI, TCG CREST

1. Prove or disprove the following statements:
   (a) Given two non-negative functions $f(n)$ and $g(n)$, either $f(n) = O(g(n))$, or $g(n) = O(f(n))$.
   (b) Given non-negative functions $f_1(n)$, $f_2(n)$, $g_1(n)$, $g_2(n)$ with $f_1(n) = O(g_1(n))$, $f_2(n) = O(g_2(n))$, and for all integers $n \geq 0$, $g_1(n) < g_2(n)$, then $f_1(n) < f_2(n)$.

2. Show that there are exactly $(n + 1)$ way so that one can fill a bag with n fruits subject to the following constraints:
   - The number of apples must be even.
   - The number of bananas must be a multiple of 5.
   - There can be at most 4 oranges.
   - There can be at most 1 pear.

3. We call a finite set $\mathcal{S} \subset \mathbb{N}$ to be "*crazy*" if $|\mathcal{S}| \in \mathcal{S}$. How many subsets of $\{1, 2, \ldots, n\}$ are there that are minimal crazy sets (minimality is defined in the sense that subsets that are crazy and do not properly contain any other crazy set). E.g., for a set $\{1, 2, 3\}$, the minimal crazy sets are $\{1\}$ and $\{2, 3\}$.

4. A diagonal in a (convex) polygon is a straight line that connects two non-adjacent vertices of the polygon. Two diagonals are different if they have at least one different endpoint. A triangulation of a polygon is a division of the polygon into triangles by drawing non-intersecting diagonals. For example, the 6-sided polygon ABCDEF below is triangulated into 4 triangles by using the diagonals AD, AE, BD. Two triangulations are different if at least one of the diagonals in a triangulation is different from all diagonals in the other triangulation.

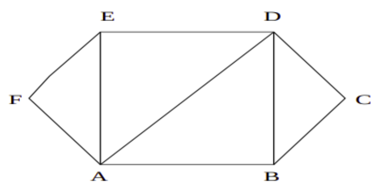

Figure 1: A Triangulation of a 6-sided Polygon

   (a) Prove that any triangulation of an n sided convex polygon has $(n - 2)$ triangles and $(n - 3)$ diagonals.
   (b) How many different triangulation is possible for a convex polygon of n vertices?

5. Consider the following algorithm for fast multiplication of two $2n$-bit integers. Suppose that a and b are integers with binary expansions of length $2n$ (add initial bits of zero in these expansions if necessary to make them the same length). Let $a = (a_{2n-1}a_{2n-2}\cdots a_1a_0)_2$ and $b = (b_{2n-1}b_{2n-2}\cdots b_1b_0)_2$. We can write

$$a = 2^nA_1 + A_0,\ b = 2^nB_1 + B_0,$$

where $A_1 = (a_{2n-1}\cdots a_{n+1}a_n)_2,\ A_0 = (a_{n-1}\cdots a_1a_0)_2,\ B_1 = (b_{2n-1}\cdots b_{n+1}b_n)_2,$ $B_0 = (b_{n-1}\cdots b_1b_0)_2$ are n-bit strings. The algorithm for fast multiplication of integers is based on the fact that ab can be rewritten as

$$ab = (2^2n + 2^n)A_1B_1 + 2^n(A_1 - A_0)(B_0 - B_1) + (2^n + 1)A_0B_0.$$

Find the total number of bit operations needed to multiply two $2n$-bit integers following the above algorithm.