Due: Sep 29, 2016, in class.

Note: for simplicity, the base of “log” is always 2.

1. (10 × 2 = 20 points) Prove (1) \( \sum_{i=1}^n \log i = \theta(n \log n) \); (2) \( \sum_{i=1}^n i = \theta(n^2) \).

2. (10 × 2 = 20 points) Compare the growth rates of \( T_1(n) \) and \( T_2(n) \):
   1. \( T_1(n) = n^2 \) and \( T_2(n) = 6n \log n - \frac{1}{2}n \);
   2. \( T_1(n) = n^3 \) and \( T_2(n) = 1.5n^3 + (-1)^{\lceil n \rceil}n^3 \).

3. (20 points) Given an array \( A = [6, 3, 5, 7, 0, 2] \), use insertion sort algorithm to sort \( A \) in an increasing order. The algorithm scans \( A \) from left to right by \( \text{size}(A) = 6 \) steps, so please list the changes of \( A \) during these 6 steps.

4. (20 points) Array \( A = [a_1, a_2, \cdots, a_n] \) is unsorted, design an \( O(n \log n) \)-time algorithm to report the number of inversions in \( A \). An inversion is a pair of numbers \( a_i \) and \( a_j \) such that \( i < j \) but \( a_i \geq a_j \) (hint: you may use divide-and-conquer idea).

5. (20 points) Given an array \( A = [3, 10, 2, 7, 12, 4, 20] \), build the MaxHeap for \( A \). Please list the updated \( A \) after calling \texttt{MaxHeapify} each time.