CS 577: Introduction to Algorithms	Homework 0
Out: 01/18/17	**Due: Not to be turned in**

Ground Rules

- Review problems will be discussed at the following week's review.
- Self-graded problems will be discussed at the following week's review, and are to be self-graded by you during the review. Only students present at the review session and turning in their graded solutions in person will earn points.
- Graded problems are to be turned in on the due date at the beginning of the lecture.
- Self-graded problems should be done individually.
- Graded problems should be done in pairs.
- Please follow page limits for all self-graded and graded problems.
- Write your name(s) clearly on your submissions, and turn in each problem on **a separate sheet of paper**.

Ungraded Problems

1. Find a partner for doing homework.
2. Familiarize yourself with the course webpage[1], calendar, and Piazza. Make sure you know all course policies.
3. Revise the following topics from CS 240 curriculum[2]: induction, solving recurrences, asymptotic notation, and graphs.
4. Prove by induction that the number of leaves in a full binary tree is one more than the number of internal nodes.
5. Prove that every integer (positive, negative, or zero) can be written in the form $\sum_i \pm 3^i$, where the exponents i are distinct non-negative integers. For example,

$$\begin{aligned} 44 &= 3^4 - 3^3 - 3^2 - 3^0 \\ 23 &= 3^3 - 3^1 - 3^0 \\ 19 &= 3^3 - 3^2 + 3^0 \end{aligned}$$

6. You are given a $2^n \times 2^n$ chessboard with one square missing (that we will call a hole). Prove using induction on n that regardless of the position of the hole, you can tile the chessboard with L-shaped pieces containing three squares each. That is, you can find an arrangement of the L-shaped tiles such that every square of the chessboard is covered by exactly one tile and the hole is left uncovered.

Review Problems

7. Order the following functions according to the asymptotically smallest to the asymptotically largest.

$$\begin{array}{ccc} \log n & \sqrt{n} & 5^n \\ n^{\log n} & 5^{\sqrt{\log n}} & (\log n)^n \\ 3^{n+10} & \log(5^n) & \sqrt{5^{\log n}} \end{array}$$

[1] http://pages.cs.wisc.edu/∼shuchi/cs577.html

[2] You can find lecture notes at http://pages.cs.wisc.edu/∼cs240-1/.

8. For this problem you are given a tree, with each node labeled by a number. A node is called a local maximum if the labels of all of its neighbors are smaller than or equal to its own label. Design a divide and conquer algorithm to find a local maximum in the tree. If there are multiple local maxima, your algorithm needs only to return one. What is the asymptotic running time of your algorithm?

9. Consider the following sorting algorithm. The initial call is SuperSort$(A, 0, \text{length}(A) - 1)$ where A is an array of integers.

```
void SuperSort(int A[],int i,int j){  \\sorts the subarray A[i..j]
    if (j == i+1)                     \\when there are only 2 elements
      if (A[i] > A[j]) swap(A,i,j)    \\swaps A[i] and A[j]
    else {
      int k = (j-i+1)/3;
      SuperSort(A,i,j-k);             \\sort first two thirds
      SuperSort(A,i+k,j);             \\sort second two thirds
      SuperSort(A,i,j-k);             \\sort first two thirds again
      }
}
```

   (a) Prove using induction that this algorithm is correct, that is, it always produces a sorted array.

   (b) Determine the asymptotic number of comparisons this algorithm makes.