Ve281 Data Structures and Algorithms
Written Assignment One

This assignment is announced on September 16th, 2018. It is due by 5:40 pm on September 26th, 2018. The assignment consists of six problems.

1. Suppose that you and your friend are playing a game of guessing a number. At the beginning, your friend writes down an integer x in the range $[1, N]$. Your goal is to guess that number using the minimum number of guesses. Each time when you guess a number, your friend will tell you whether the number you guess is equal to, less than, or greater than the number x he initially sets. Once your guess equals x, the game ends.

   Describe a good strategy that you could use to guess the initial number using as few number of guesses as possible. Suppose that $N = 2^m - 1$. What is the number of guesses required in the worst case? What is the number of guesses required on average? Give the exact value, not the big-oh, big-omega, or theta notation.

2. Consider the following code

   ```
   while(n > 1)
   {
     if(n % 2 == 1)
       n = 3 * n + 1;
     else
       n /= 2;
   }
   ```

   What is the *best lower* bound $f(n)$ so that the complexity of the above code is in the set $\Omega(f(n))$?

3. Is the statement $n^{100} = O(1.001^n)$ true or not? Prove your answer in a strict way.

4. Prove the following result on the **theta notation** based on its definition:

   Suppose that $g_1(n)$ and $g_2(n)$ are non-negative functions when $n \geq 0$. If $f_1(n) = \Theta(g_1(n))$ and $f_2(n) = \Theta(g_2(n))$, then $f_1(n)+f_2(n) = \Theta(h(n))$, where $h(n) = \max\{g_1(n), g_2(n)\}$.

5. Consider the following code

```
void func(int n, int a, int b) {
  for(i = 1; i <= n; i *= a)
    for(j = 1; j <= i * b; j++)
      Statement;
}
```

Assume that $n \geq 1$, $a > 1$, and $b \geq 1$ are integers. How many times will "Statement" be called? Write the answer in terms of n, a, and b.

6. A new house is being built down the street. The builders are currently working on the roof. From observation, there seems to be three different methods for moving the shingles from the shingle truck to the roof.

Method 1: A builder can carry two packages (known as squares) of shingles on his shoulder as he climbs up the ladder. He then climbs down and carries two more squares up the ladder. So on and so forth. Each round trip (up the ladder with two squares and down the ladder with none) costs \$2.

Method 2: A builder rents a lift for \$10. The lift can move 20 squares up the roof at one time at a cost of \$1 per round trip.

Method 3: A builder rents a super-lift for \$40. Unfortunately, the lift has a slow leak in its hydraulic system. The lift is able to lift roughly half of the necessary squares to the roof on the first round trip. However, on the second trip the lift is only able to lift half of the remaining squares, then half of the remaining, and so on. To be strict, if the number of the remaining squares is n, the lift lifts $\lfloor \frac{n}{2} \rfloor$ squares. Even when the super-lift has no hydraulic fluid left in its system, it can still lift one square of shingles. Each round trip using the super lift costs \$2.

Note that in all three methods, it costs \$4/square to nail the shingles into the roof.

Question:

In the following questions, "cost" refers to total cost, including nailing shingles to the roof.

(a) For each method, write a formula $T(n)$ that expresses the number of round trips required to move n squares of shingles to the roof.

(b) For each method, write a formula $C(n)$ that expresses the entire cost to install n squares of shingles on the roof.

(c) Which is cheapest method for a doghouse (8 squares)? How much does it cost?

(d) Which is cheapest method for a shed (128 squares)? How much does it cost?

(e) Which is cheapest method for a house (2048 squares)? How much does it cost?