

## 2. Homework

Due **2/3/09** before class

### 1. Climb a Ladder (5 points)

Suppose there is a ladder with  $n$  rungs,  $n \geq 1$ . The way the rungs are spaced, a person can either climb one rung, two rungs, or three rungs at a time. How many different ways are there to climb a ladder with  $n$  rungs? For example, if  $n = 4$  then there are 7 different ways to climb the ladder: 1111, 112, 121, 211, 22, 13, 31.

- (2 points) Give pseudocode for a recursive algorithm that computes, for any  $n \geq 1$ , the number of ways to climb a ladder with  $n$  rungs.
- (1 point) Set up a runtime recurrence for your algorithm (base case and recursive case). You do not need to solve it.
- (1 point) Give the number of ways to climb a ladder with  $n$  rungs for  $n = 1, 2, 3, 4, 5, 6$ .
- (1 point) Modify your pseudo code to actually print out all different ways to climb a ladder with  $n$  rungs.

### 2. Min in a Max-Heap (2 points)

Where is the minimum element located in a max-heap? How can you compute it, and what is the runtime of your algorithm?

### 3. Number of Leaves in a Heap (4 points)

Let  $n$  be the number of nodes in a heap. Show by (weak) induction on  $n$  that the number of leaves is  $\lceil n/2 \rceil$ . (*Hint: It may be easier to deal with the ceiling by considering two cases  $n = 2a$  and  $n = 2a + 1$ . Note that a left child always has an even index, and a right child always has an odd index.*)

### 4. Recursive Mystery (4 points)

Consider the recursive method below:

```
mystery(n){
    if(n==1)
        return 1;

    int j=0;
    for(int i=0; i<n; i++){
        j+=5;
    }

    return mystery(n-1);
}
```

- (2 points) Set up a runtime recurrence for this method, including base case and recursive case. Shortly argue why your recurrence correctly specifies the runtime.

- (b) (2 points) Using the recursion tree method, come up with a guess what this runtime recursion will solve to. What is the runtime in  $\Theta$ -notation, using simple a simple runtime function (such as  $\Theta(n)$ ,  $\Theta(n^2)$ ,  $\Theta(n^3)$ )?

5. **Runtime Recurrence (3 points)**

Let  $T(0) = 1$  and  $T(n) = 2 * T(n - 1) + 1$  be a runtime recurrence. Prove using induction on  $n$  that  $T(n) = 2^{n+1} - 1$ .