Mock In-class Test
COMS10007 Algorithms 2018/2019

Throughout this paper log() denotes the binary logarithm, i.e, $\log(n) = \log_2(n)$, and ln() denotes the logarithm to base e, i.e., $\ln(n) = \log_e(n)$.

1 O-notation

1. Let $f : \mathbb{N} \to \mathbb{N}$ be a function. Define the set $\Theta(f(n))$.
2. Give a formal proof of the statement:
$$10\sqrt{n} \in O(n) \ .$$
3. Use the racetrack principle to prove the following statement:
$$n \in O(2^n) \ .$$
Hint: The following facts can be useful:
   - The derivative of 2^n is $\ln(2)2^n$.
   - $\frac{1}{2} \le \ln(2) \le 1$ holds.

2 Sorting

1. Why is Mergesort not an in-place sorting algorithm?
2. A divide-and-conquer algorithm consists of three parts: The divide, the conquer, and the combine phase. Compare Mergesort and Quicksort with regards to these three phases.
3. What is the runtime (in Big-O notation) of Insertionsort when executed on the following arrays of lengths n: (no justification needed)
   (a) $1, 2, 3, 4, \dots, n-1, n$
   (b) $n, n-1, n-2, \dots, 2, 1$

3 Loop-Invariant

Consider the following algorithm:

Algorithm 1

Require: integer $n \geq 1$

```
1: x ← 1
2: for i ← 1, ..., n − 1 do
3:     x ← 2 · x + 1
4: end for
5: return x
```

The goal of this exercise is to show that this algorithm computes the value $2^n - 1$ on input n. Let x_i be the value of x at the beginning of iteration i (i.e., after i is updated in Line 2 and before Line 3 is executed). Consider the following loop invariant:

$$x_i = 2^i - 1$$

1. *Initialization:* Argue that at the beginning of the first iteration, i.e. when $i = 1$, the loop-invariant holds.

2. *Maintenance:* Suppose that the loop invariant holds at the beginning of iteration i. Argue that the loop-invariant then also holds at the beginning of iteration $i + 1$.

3. *Termination:* Use the loop invariant to conclude that the algorithm indeed computes the value $2^n - 1$ on input n.

4. What are the worst-case and best-case runtimes of the algorithm?