COMP 285 (NC A&T, Spr ‘22) Weekly Quiz 1

1

Recall the definitions of big-Oh, big-Omega, and big-Theta. Based on these definitions, select all statements below which are true.

Solution

The below are all trust statements.

1. Big-Oh is an upper bound. That is, intuitively, $T(n) = O(f(n))$ means $T(n) \leq cf(n)$ for some c.
2. Big-Theta provides both upper and lower bounds.
3. Big-Omega is a lower bound.

The above are by definition.

2

What is the running time of the following function: $T(n) = 13 + 12nlog(n) + 3n^2$

Solution

$\Theta(n^2)$ because n^2 is both upper and bower bounds. The other terms grow much more slowly, which means for large n, they do not impact the result much.

3

Which of the following is the correct English description of $f(n) = O(g(n))$?

Solution

There is some $c > 0$ and some n_0, such that for all $n \geq n0$ we have $f(n) \leq c \cdot g(n)$.

4

Suppose that $f(n) = O(g(n))$. Which of the following is implied by this fact?

Solution

$g(n) = \Omega(f(n))$. Intuitively, $f(n) = O(g(n))$ means that $f(n) \leq cg(n)$ for some c and large enough n. This means that $g(n) \geq \frac{1}{c}f(n)$ for large enough n (divide both sides by c), which fits the definition of $g(n) = \Omega(f(n))$.

5

What is the smallest exponent x such that $n^2 + n^3 - n = O(n^x)$

Solution

3. Because the asymptotic upper bound is n^3

6

Is Merge Sort's worst case runtime asymptotically faster than Insertion Sort's worst case runtime?

Solution

Yes. Merge Sort's worst case runtime is $O(n \log n)$, and Insertion Sort's worst case runtime is $O(n^2)$.

7

Which of the following describes $\frac{n(n+1)(n+2)}{6}$

Solution

- $O(n^4)$: $\frac{n(n+1)(n+2)}{6} = \frac{n^3+3n^2+2n}{6}$ so the asymptotic upper bound should be $O(n^3)$. Since $O(n^4)$ grows faster than $O(n^3)$, it can be the upper bound.
- $O(n^3)$: It's the asymptotic upper bound.
- $\Theta(n^3)$: It's both the upper bound and the lower bound.
- $\Omega(n^2))$: Since $O(n^2)$ grows slower than $O(n^3)$, it can be the lower bound.

8

Is Merge Sort faster than Insertion Sort on some arrays?

Solution

Yes, Merge Sort's worst case runtime is $O(n \log n)$, and Insertion Sort's worst case runtime is $O(n^2)$.

In particular, Insertion Sort will take $\Theta(n^2)$ steps on an array thats sorted in reverse order, while merge sort will that $\Theta(n \log n)$ steps.

9

Is Merge Sort faster than Insertion Sort on all input arrays?

Solution

No. Merge Sort's best case runtime is$O(n \log n)$, and Insertion Sort's best case runtime is $O(n)$.
In particular, Insertion Sort will take $\Theta(n)$ steps on an array that's already sorted, while merge sort will take $\Theta(n \log n)$ steps.

10

The Merge operation takes two arrays A and B of size n which are already sorted and outputs the union of the two in sorted order. What is the smallest bound on the runtime of the Merge algorithm?

Solution

$O(n)$. The Merge operation needs to walk through every element in A and B.