#7: \( \log(n^2) = ? \log(n^3) \). Choose the tightest bound.

On #7 for the prerequisite practice exam, my study group is a little confused and we are arriving at 2 different answers.

\[
\log(n^2) = 2 \cdot \log(n) \quad \text{and} \quad \log(n^3) = 3 \cdot \log(n).
\]

Then the time complexity of \(2\log(n)\) is just \(\log(n)\) & the time complexity of \(3\log(n)\) is just \(\log(n)\) because we ignore the constants.

Some of us think the answer is \(\log(n^2) = o(\log(n^3))\) because \(\log(n^2) < \log(n^3)\) for \(n > 1\).

Can anyone clear this up for us? We find both answers to have a strong case as to why they are correct.

Thanks

---

The are big theta to each other as the reason you stated. All logarithms of polynomials are big theta. The proof is not that difficult either.

The reason it can't be little omicron follows from above.

---

#7 asks for math notation. Although that is true for time complexity, wouldn't the answer be little omicron for math notation purposes? Just as \(n^2 = o(n^3)\)?

---

What is true for math notation? That \(n2n2 \) is \(o(n^3)\)?
The answer doesn't have to be little omicron. Remember the rules of log, \( \log(n^k) = k\log(n) \) and the answer should jump out at you.

Subject: Re: Comparing logarithms time complexity
Posted by pcburns on Mon, 23 Jan 2017 19:36:59 GMT

I figured out the answer by reading about tight bounds in the book