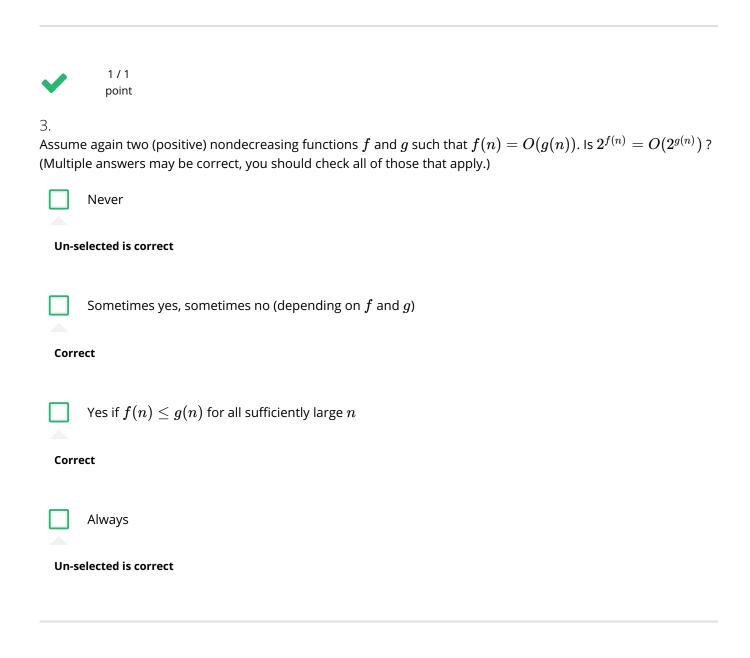
Problem Set #1 Quiz, 5 questions

5/5 points (100.00%)

✓	Congratulations! You passed! Next Item
~	1/1 point
each th	Merge Sort : Suppose that instead of dividing in half at each step of Merge Sort, you divide into thirds, sort aird, and finally combine all of them using a three-way merge subroutine. What is the overall asymptotic g time of this algorithm? (Hint: Note that the merge step can still be implemented in $O(n)$ time.)
	$n(\log(n))^2$
0	$n\log(n)$
Correct That's correct! There is still a logarithmic number of levels, and the overall amount of work at each level is still linear.	
	$n^2\log(n)$
	n
~	1 / 1 point
	e given functions f and g such that $f(n)=O(g(n)).$ Is $f(n)*log_2(f(n)^c)=O(g(n)*log_2(g(n)))$? is some positive constant.) You should assume that f and g are nondecreasing and always bigger than 1.
	False
	Sometimes yes, sometimes no, depending on the functions $oldsymbol{f}$ and $oldsymbol{g}$
	Sometimes yes, sometimes no, depending on the constant $oldsymbol{c}$

True

Problem Set #1
That's correct! Roughly, because the constant c in the exponent is inside a logarithm, it becom to be points (100.00%)
Quiz, 5 questions
the leading constant and gets suppressed by the big-Oh notation.





1/1 point

4

k-way-Merge Sort. Suppose you are given k sorted arrays, each with n elements, and you want to combine them into a single array of kn elements. Consider the following approach. Using the merge subroutine taught in lecture, you merge the first 2 arrays, then merge the 3^{rd} given array with this merged version of the first two arrays, then merge the 4^{th} given array with the merged version of the first three arrays, and so on until you merge in the final (k^{th}) input array. What is the running time taken by this successive merging algorithm, as a function of k and k? (Optional: can you think of a faster way to do the k-way merge procedure?)



Problems Set #1

Quiz, 5 dust is some content of the upper bound, the merged list size is always O(kn), merging is linear in the size of the larger array, and there are k iterations. For the lower bound, each of the last k/2 merges takes $\Omega(kn)$ time.

- $\theta(nk)$
- $\theta(n\log(k))$
- $\theta(n^2k)$



1/1 point

5

Arrange the following functions in increasing order of growth rate (with g(n) following f(n) in your list if and only if f(n) = O(g(n))).

- a) \sqrt{n}
- b) 10^{n}
- c) $n^{1.5}$
- d) $2^{\sqrt{\log(n)}}$
- e) $n^{5/3}$

Write your 5-letter answer, i.e., the sequence in lower case letters in the space provided. For example, if you feel that the answer is a->b->c->d->e (from smallest to largest), then type abcde in the space provided without any spaces before / after / in between the string.

You can assume that all logarithms are base 2 (though it actually doesn't matter).

WARNING: this question has multiple versions, you might see different ones on different attempts!

Preview

d, a, c, e, b daceb

Please note: Each of the following will be interpreted as a single variable, not as a product of variables: daceb. To multiply variables, please use * (e.g. enter x*y to multiply variables x and y).

daceb

Problems Ser #1

Quiz, 5 Question 3).

Quiz, 5 Question 3).

Quiz, 5 Question 3.

Your answer, daceb, is equivalent to the instructor's answer daceb.





