Algorithms and Data Structures Coursework Questions 4-6

jsbl33

January 20, 2020

1 Question 4

a) $f(x) + g(x)$ is $o(f(x) \times g(x))$

For a function to be little-o of another function,

$$\lim_{x\to\infty} \frac{a(x)}{b(x)} = 0$$

In our case,

$$\begin{aligned}
&\lim_{x\to\infty} \frac{f(x)+g(x)}{f(x)g(x)} \\
&= \lim_{x\to\infty} (\frac{f(x)}{f(x)g(x)} + \frac{g(x)}{f(x)g(x)}) \\
&= \lim_{x\to\infty} (\frac{1}{g(x)} + \frac{1}{f(x)})
\end{aligned}$$

This is not always equal to zero, for instance when $f(x) = 1$ it is equal to 1.

$\therefore$ The statement is false.

b) $2^x \times x^2$ is $o(2.1^x)$

Considering the limit as x tends to infinity

$$\begin{aligned}
&\lim_{x\to\infty} \frac{2^x \times x^2}{2.1^x} \\
&= \lim_{x\to\infty} (\frac{2}{2.1})^x \times x^2 \\
&= 0
\end{aligned}$$

$\therefore$ The statement is true.

c) $x^2 \times \log x$ is $O(x^2)$

Assume that there are constants k and C such that $x^2 \times \log x \leq C \times x^2$ when $x \geq k$.

Then $\log x \leq C$ when $x \geq k$

But $\log x$ increases monotonically, a contradiction.

$\therefore$ the statement is false.

d) $x^2 \times \log x$ is $O(x^3)$

$$f(x) = x^2 \times \log x$$

For $x > 0$, $\log x < x$

$\therefore f(x) = x^2 \times \log x \leq x^2 \times x = x^3$

For the statement to be true, $x^3 \leq C \times x^3$

$C = 1, k = 0$ is a valid witness pair that makes this true.

$\therefore$ The statement is true.

e) $7x^5$ is $O(12x^4 + 5x^3 + 8)$

If $f(x) = O(12x^4 + 5x^3 + 8)$, then there must be a valid witness pair C and k such that

$7x^5 \leq C(12x^4 + 5x^3 + 8)$ for $x \geq k$

$\frac{7x^5}{12x^4+5x^3+8} \leq C$

C is not a constant here so the statement is false.

2 Question 5

a) $T(n) = 64T(n/8) - n^2 \times \log n$

$\log_b a = 2, f(n) = n^2 \times \log n$

The master theorem cannot be applied in this instance, since $f(n)$ is not positive.

b) $T(n) = 4T(n/2) + \frac{n}{\log n}$

$\log_b a = 2, f(n) = \frac{n}{\log n} = n \times \log^{-1} n$

This is case 1 of the master theorem, so $T(n) = \theta(n^2)$

c) $T(n) = 2^n T(n/2) + n^n$

The master theorem cannot be applied here since coefficient of $T(n/2)$ is not a constant.

d) $T(n) = 3T(n/4) + n \times \log n$

$\log_b a = 0.792, f(n) = n \times \log n$

This is looking like case 3 of the master theorem, though we need to check the regularity condition:

$\frac{3n}{4} \times \log n/4 \leq cn \log n$

This is true for $c = 1$.

$\therefore T(n) = \theta(n \times logn)$

e) $T(n) = 3T(n/3) + n^{\frac{1}{2}}$

$\log_b a = 1, f(n) = n^{\frac{1}{2}}$

This is case 1 of the master theorem, $\therefore T(n) = \theta(n)$

3 Question 6

b) With a regular single-pivot quicksort, the worst-case occurs when the pivot is the maximum or minimum value in the array. When this occurs, partitioning only removes one item from consideration, minimising the advantages of the 'divide-and-conquer' approach. For a multi-pivot quicksort, the worst-case would be when all selected pivots are maxima or minima compared to the remaining non-pivot elements. This would result in only k elements being removed from consideration on the partition, which is the minimum possible when there are k pivots. The selection of the pivot determines which inputs result in this worst-case. In my algorithm, the first k elements of the array are selected as pivots, so I will describe the input based on this. Given an input array of size n, the worst-case occurs when for every "block" of k elements, it contains the x smallest and y largest elements of the array from the start of the "block" until the end. x and y can take any positive value as long as x + y = k. This input ensures that any partition only removes k elements from consideration, minimising the advantage of divide-and-conquer.