Homework Assignment #2

Due: Monday Jan. 31, 2022 (11:55pm)

(Total 40 marks)

CMPUT 204
Department of Computing Science
University of Alberta

Problem 1. (10 marks) True or False? Justify your answer briefly.

a. $n^3 + 2n^2 \in \omega(0.1n^{3.1} + n)$

b. $n \log^5 n \in \Theta(n^{\frac{5}{4}})$

c. $2^{n \log n} \in \Theta(4^n)$

d. For any constant $a > 1$: $a^{n+1} \in o(n!)$.

e. $\sqrt{\log n} \in o(\log(\sqrt{n}))$

Problem 2. (10 marks) Order the following list of functions by increasing big-Oh growth rate. Group together those functions that are big-Theta of one another.

$3^{\sqrt{n}}$	$2^{2+1/n}$	$\log n/\log\log n$	$\log n$	$e^{\ln n}$
2^{2^n}	4^n	n^{100}	$1/\log n$	$4n^{3/2}$
$3^{\sqrt{\log n}}$	$5(n+1/n)$	$n \log^3 n$	$2^{n^2 \log n}$	$\log(n!)$
$n!$	n^3	$n^2 \log n$	$4^{\log n}$	$\sqrt{\log\log n}$

Problem 3. (10 marks) Prove or disprove each of the following statements:

a. For any functions $f, g : \mathbb{N} \longrightarrow R^+$, either $f \in O(g)$ or $g \in O(f)$.

b. For any functions $f, g : \mathbb{N} \longrightarrow R^+$, if $f(n) > g(n)$ for all $n > 0$ then $f(n) + g(n) \in \Theta(f(n))$.

c. For any functions $f, g : \mathbb{N} \longrightarrow R^+$, if $f(n) \in \Theta(n)$ and $g(n) \in \Theta(n)$ then $2^{f(n)} \in \Theta(2^{g(n)})$.

d. For any function $f : \mathbb{N} \longrightarrow R^+$, $f(n) \in \Theta((f(n))^2)$.

Problem 4. (10 marks) Suppose you are given a set of small boxes, numbered 1 to n, identical in every aspect except that each of the first i contains a pearl whereas the remaining $n - i$ are empty. In the first part of this question, you are given two magic wands that can each test if a box is empty or not in a single touch, except that a wand disappears if you test it on a box that is empty. Show that, without knowing the value of i, you can use the two wands to determine all the boxes containing pearls using no more than $O(\sqrt{n})$ wand touches.

a. **We here sketch a solution for this part of the question.**

We will fix number k later. The idea is first use one wand on boxes $1, k, 2k, 3k, \ldots$. The smallest i for which the wand burns on box ik indicates that the first empty box is among $(i-1)k+1, \ldots, ik$. Now we use the second wand sequentially from $(i-1)k+1$ to $ik-1$ to find it. The total number of touches will be at most: $n/k + k$, where n/k is the number of boxes for the first wand and k for the second.

Now, you are asked to complete this solution by describing how you choose the value of k and explaining why your choice leads to an algorithm that has the desired efficiency (i.e., to determine all the boxes containing pearls using no more than $O(\sqrt{n})$ wand touches).

b. Now suppose you are given three magic wands. How would you extend the above algorithm to obtain an even more efficient one? Describe your algorithm, give an upper bound in running time (in terms of the number of wand touches) using the big-O notation, and provide a brief analysis.

c. We should all be familiar with binary search: Given a sorted list of items, it works by repeatedly dividing in half the portion of the list that could contain the item, until you've narrowed down the possible locations to just one. In this question, the given list of boxes can be considered sorted: non-empty boxes proceed empty boxes. Give an algorithm to find the first empty box in $O(\log n)$ time independent of how many magic wards may be used. Then, answer the following questions: by your algorithm, in the worst-case how many magic wards are required (express your answer using the big-O notation)? What about the best-case? Justify your answers briefly.