Jovan Shandro

Homework 1 A.D.S.

1 a) $f(n) = 3n$ and $g(n) = n^3$

we first notice that

$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{3}{n^2} = 0 \Rightarrow f(n) = o(g(n)) \Rightarrow g(n) = \omega(f(n))$

from what we said above $f(n) \neq \omega(g(n))$ and $g(n) \neq o(f(n))$

Now we check if $f(n) = O(g(n))$?

since as $n\to\infty$ $\frac{f(n)}{g(n)} \to 0$ we can find $c \in \mathbb{R}$ and $N \in \mathbb{N}$

such that $f(n) \leq g(n)\cdot c \quad \forall n \geq N$

so $f(n) = O(g(n))$

and since $\frac{f(n)}{g(n)} \to 0$ we can't find a constant

$c \neq 0$ such that there $\exists N \in \mathbb{N}$ so that

$\frac{f(n)}{g(n)} \geq c \quad \forall n \geq N$ so $f(n) \notin \Omega(g(n))$

thus $f(n) \neq \Theta(g(n))$

Now we check if $g(n) = O(f(n))$?

Since $\frac{g(n)}{f(n)} \to \infty$ as $n\to\infty$ we can't find a

constant c s.t $\frac{g(n)}{f(n)} \leq c$ as $n\to\infty$

so $g(n) \neq O(f(n))$.

but since $\frac{g(n)}{f(n)} \to \infty$ we can find a constant c

such that $\frac{g(n)}{f(n)} \geq c$ as $n\to\infty$ so

$g(n) = \Omega(f(n))$ this and the last inequality

prove $g(n) \neq \Theta(f(n))$

so in total we have

$\boxed{f(n) = O(g(n))}$ $\boxed{f(n) = o(g(n))}$

$\boxed{g(n) = \Omega(f(n))}$ $\boxed{g(n) = \omega(f(n))}$

1.b) $f(n) = 7n^{0.7} + 2n^{0.2} + 13 \log_2 n$

$g(n) = \sqrt{n}$

we first find that doing basic limit calculations and then applying L'Hospital's rule we get

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} 7 \cdot n^{0.2} + 2\lim_{n\to\infty} n^{-0.3} + 13 \lim_{n\to\infty} \frac{\log_2 n}{\sqrt{n}}$$

$$\lim_{n\to\infty} 7n^{0.2} = \infty \quad ; \quad \lim_{n\to\infty} n^{-0.3} = 0$$

and using L'Hospital

$$\lim_{n\to\infty} \frac{\log_2 n}{\sqrt{n}} = \lim_{n\to\infty} \frac{\frac{1}{\ln(2)} \cdot \frac{1}{n}}{\frac{1}{2\sqrt{2}}} = \frac{1}{\ln 2} \lim_{n\to\infty} \frac{2}{\sqrt{n}} = 0$$

So $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty + 0 + 0 = \infty$

So $f(n) = \omega(g(n))$ and $g(n) = o(f(n))$

and from that $f(n) \neq o(g(n))$ and $g(n) \neq \omega(f(n))$

Now, since $\frac{f(n)}{g(n)} \to \infty$ as $n \to \infty$ we can find a constant c s.t $\frac{f(n)}{g(n)} \geq c$ as $n\to\infty$ but we cannot find a constant c such that

$\frac{f(n)}{g(n)} \leq c$ as $n \to \infty$ so

$$f(n) = \Omega(g(n)) \text{ but } f(n) \neq O(g(n))$$

$$\Rightarrow f(n) \neq \Theta(g(n))$$

And since $\frac{g(n)}{f(n)} \to 0$ as $n\to\infty$ we can find a constant c so that $\frac{g(n)}{f(n)} \leq c$ as $n \to \infty$ but we can't find $c \neq 0$ such that $\frac{g(n)}{f(n)} \geq c$ as $n \to \infty$

So

$$g(n) = O(f(n)) \text{ but } g(n) \neq \Omega(f(n))$$

$$\Rightarrow g(n) \neq \Theta(f(n)).$$

So only the underlined in red are true.

1.c) $f(n) = \frac{n^2}{\log_2(n)}$ and $g(n) = n \cdot \log_2(n)$

Using L'Hospital rule 2 times we get

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{n^2}{\log_2(n) \cdot n \cdot \log_2(n)} = \lim_{n\to\infty} \frac{n}{(\log_2 n)^2}$$

$$\overset{L'H}{=} \lim_{n\to\infty} \qquad = \frac{\ln(2)}{2} \cdot \lim_{n\to\infty} \left(\frac{n}{\log_2(n)} \right)$$

$$\overset{L'H}{=} \frac{\ln(2)}{2} \cdot \lim_{n\to\infty} \frac{1}{\frac{1}{n \cdot \ln(2)}} = \infty$$

So $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty \Rightarrow$ $f(n) = \omega(g(n))$ and $g(n) = o(f(n))$

Since $\frac{f(n)}{g(n)} \to \infty$ we can not bound it from above but we can bound it from below so we can't find $c \in \mathbb{R}$ s.t $\frac{f(n)}{g(n)} \leq c$ as $n \to \infty$ but $\exists c$ s.t $\frac{f(n)}{g(n)} \geq c$ as $n\to\infty$ so

$$\left.\begin{array}{l} f(n) = \Omega(g(n)) \\ \text{but } f(n) \neq O(g(n)) \end{array}\right\} \Rightarrow f(n) \neq \Theta(g(n))$$

Since $\frac{g(n)}{f(n)} \to 0$ we can not bound it from below but we can from above so $\exists c$ s.t $\frac{g(n)}{f(n)} \leq c$ as $n\to\infty$ but $\nexists c$ s.t $\frac{g(n)}{f(n)} \geq c$ as $n\to\infty$ so we can say

$$\left.\begin{array}{l} g(n) = O(f(n)) \\ \text{but } g(n) \neq \Omega(f(n)) \end{array}\right\} \Rightarrow g(n) \neq \Theta(f(n))$$

So only those underlined in red are equalities

1. d) $f(n) = (\log_2(3n))^3$ and $g(n) = 9 \cdot \log n$

we see using again L'Hospital's rule that

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \frac{1}{9} \lim_{n\to\infty} \frac{(\log_2(3n))^3}{\log n} \overset{L'H}{=} \frac{1}{9} \lim_{n\to\infty} \frac{3(\log_2(3n))^2 \cdot \frac{1}{\ln(2)\cdot n} \cdot 3}{\frac{1}{\ln(2)\cdot n}}$$

$$= \lim_{n\to\infty} \frac{(\log(3n))^2}{3} = \infty$$

so $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty \Rightarrow$ $f(n) = \omega(g(n))$ and $g(n) = o(f(n))$

Since $\frac{f(n)}{g(n)} \to \infty$ as $n\to\infty$ $\nexists\, c \in \mathbb{R}$ such that $\frac{f(n)}{g(n)} \leq c$ as $n\to\infty$ but $\exists c$ such that $\frac{f(n)}{g(n)} \geq c$ as $n\to\infty$

So we can say that

$f(n) = \Omega(g(n))$ and $f(n) \neq O(g(n))$ $\Big\} \Rightarrow f(n) \neq \Theta(g(n))$

Since $\frac{g(n)}{f(n)} \to 0$ as $n\to\infty$ $\nexists c$ such that $\frac{g(n)}{f(n)} \geq c$ as $n\to\infty$ but $\exists c$ such that $\frac{g(n)}{f(n)} \leq c$ as $n\to\infty$

So $g(n) = O(f(n))$ but $g(n) \neq \Omega(f(n))$ $\Rightarrow g(n) \neq \Theta(f(n))$

So only the underlined in red are equalities

Jovan Shandro

Exercise2.

a) The source code in java is in the zip file.

b). When the 1st loop starts none of the elements are considered sorted. Starting from the left, on the first iteration of the loop the minimum element is found and swapped with the 1st one. So now the first element is sorted.

On the j^{th} iteration of the loop, the $(j-1)$ first elements will be sorted and the other elements on the right are bigger than these $(j-1)$ numbers. The loop finds again the smallest element on the right and swapps it with the j^{th} number and since this number is bigger than all the numbers in the left and smaller than the ones on the right now j elements are sorted.

The loop ends when the $(n-1)$ first elements are sorted but since the n^{th} is left bigger than all the others it is also sorted. So the array is fully sorted and the algorithm is correct.

c) d) Explained in code which is both in .java and .txt files.

e) Since the algorithm contains a nested loop and because of the fact that even if the numbers are already sorted it still needs to check whether every element is smaller than all elements on the right on every iteration. That's why we expect the algorithm to perform $O(n^2)$ in all cases, which is also explained and verified by the graphs (plots)