Algorithms and Data Structures 2023/24
Notes for Week 3

1. Work out (don't bother proving) for each pair of expressions below, whether A is $O(B)$, $\Omega(B)$, $\Theta(B)$ (it could be none of these). Assume $k \geq 1, \epsilon > 0, c > 1$ are all constants.

A	B	O	Ω	Θ
$\lg^k n$	n^ϵ	X		
n^k	c^n	X		
$\sqrt{n}$	$n^{\sin n}$			
2^n	$2^{n/2}$		X	
$n^{\ln m}$	$m^{\ln n}$	X	X	X
$\lg(n!)$	$\lg(n^n)$	X	X	X

The more tricky ones are the 3rd and the last.

The 3rd:

The answer is that $\sqrt{n}$ is neither $O(n^{\sin(n)})$ nor $\Omega(n^{\sin(n)})$. The reason for this is the sin curve and the erratic behaviour of $\sin(n)$.

No matter how big n_0 is, there are always infinitely many $n > n_0$ so that $\sin(n)$ approaches 1; also there are infinitely many $n > n_0$ so that $\sin(n)$ approaches -1.

The last:

It is actually the case that this pair of expressions are Theta of each other. This might be surprising to the students because it is absolutely *not* true if we don't take the 'lg' of each side.

Use the formula $n^{n/2} \leq n! \leq n^n$. Tell them you are using this - they will be using it later in the course, but haven't seen it yet.

2. For (c) you only need to show (a) and (b), then you get (c) automatically. So the main thing to do is prove (a) and (b).

I'll do (b) (the lower bound Ω), the more tricky one (tutors/students probably able to do (a) themselves – though tutors might want to use it as a warm-up).

Proof of (b): Assume at least one coefficient a_i, for $i < d$, is non-zero (otherwise the proof is easy), and define $C = (|a_0| + |a_1| + \ldots + |a_{d-1}|)$. Take $c = a_d/2$, $n_0 = 2\lceil C/a_d \rceil$.

Then for all $n \geq n_0$, we have

$$\begin{aligned}
\sum_{i=0}^{d} a_i n^i &\geq a_d n^d - \sum_{i=0}^{d-1} |a_i| n^i \\
&\geq (a_d/2) n^d + C n^{d-1} - \sum_{i=0}^{d-1} |a_i| n^i \\
&\geq (a_d/2) n^d + C n^{d-1} - C n^{d-1} \quad = \quad (a_d/2) n^d \\
&\geq (a_d/2) n^k \qquad \text{for any } k \leq d.
\end{aligned}$$

Hence by definition of Ω, we have $p(n) = \Omega(n^k)$ for all $k \leq d$.

A good way to "arrive at" the proof (rather than present it as a 'fait accompli') is to consider what we need to show $\Omega(n^k)$ *(for* $k \leq d$*). We need to find* $c > 0, n_0 \in \mathbb{N}$ *so that*

$$\begin{aligned}
& \textstyle\sum_{i=0}^{d} a_i n^i \geq c n^k \qquad \text{for } k \leq d. \\
\Leftrightarrow \quad & \textstyle\sum_{i=0}^{d} a_i n^i - c n^k \geq 0 \qquad \text{for } k \leq d.
\end{aligned}$$

and develop this downwards to see what setting of c, n_0 *are sufficient to make this possible.*

3. Let a and m be fixed and consider the complexity in n.

   POWER-REM2: Get the number of arithmetic operations by counting.

   $$T_A(n) = 1 + (n-1) * O(1)$$

   Thus POWER-REM2 has complexity $\Theta(n)$.

   POWER-REM3: Get a recurrence for $T_A(n)$. We have $T_A(1) = 1$ and $T_A(n) \leq T_A(n/2) + 4$, yielding $O(\log n)$. Similarly $T_A(n) \geq T_A(n/2) + 2$, yielding $\Omega(\log n)$. Thus POWER-REM3 has complexity $\Theta(\log n)$.