



Algorithms and Complexity
Assignment 0

Name: Evangelos Argyropoulos

AM: 1115202200010

Exercise 1

A

To order the functions I am using the complexity classes of $\log_2(a_i)$, so only in the case where I end up with the same complexity for some $\log_2(a_i)$, I need compare the corresponding a_i to order them.

1.

$$\log(a_1) = \log(\sqrt{n!}) = \frac{1}{2} \log(n!) = \Theta(n \log n)$$

because:

$$\left. \begin{aligned} \bullet n! &= 1 \cdot 2 \cdot \dots \cdot n \leq n \cdot n \cdot \dots \cdot n = n^n \Rightarrow \log(n!) \leq \log(n^n) = n \log(n) \\ &\Rightarrow \log(a_1) = O(n \log(n)) \\ \bullet n! &= 1 \cdot 2 \cdot \dots \cdot (n-1) \cdot n \geq \frac{n}{2} \cdot \left(\frac{n}{2} + 1\right) \cdot \dots \cdot n \geq \frac{n}{2} \cdot \frac{n}{2} \cdot \dots \cdot \frac{n}{2} = \left(\frac{n}{2}\right)^{\frac{n}{2}} \\ &\Rightarrow \log(n!) \geq \log\left(\left(\frac{n}{2}\right)^{\frac{n}{2}}\right) = \frac{n}{2} \log\left(\frac{n}{2}\right) = \frac{n}{2} \log n - \frac{n}{2} \\ &\Rightarrow \log(a_1) = \Omega(n \log n) \end{aligned} \right\} \Rightarrow \log(a_1) = \Theta(n \log n)$$

2.

$$\begin{aligned} a_2 &= \binom{n}{6} = \frac{n!}{(n-6)!6!} = \frac{n(n-1)(n-2)(n-3)(n-4)(n-5)}{6!} \\ \log(a_2) &= \log\left(\frac{n!}{(n-6)!6!}\right) = \log\left(\frac{n(n-1)(n-2)(n-3)(n-4)(n-5)}{6!}\right) \\ &\approx \log(n^6) = 6 \log n = \Theta(\log n) \end{aligned}$$

3.

$$\log(a_3) = \log(\log^{10}(n)) = \log((\log n)^{10}) = \log(10 \cdot \log n) = \log(10) + \log(\log n) = \Theta(\log(\log n))$$

4.

$$\log(a_4) = \log(n \cdot 2^n) = \log(n) + \log(2^n) = \log n + n = \Theta(n)$$

5.

$$\log(a_5) = \log(\log(5 \cdot n!)) = \log(\log 5 + \log(n!)) \stackrel{\text{ignoring } \log 5 \text{ because } \log(n!) \text{ is increasing more}}{\approx} \log(\log(n!))$$

Using the basic arrangement:

$c < \log^* n < \log(\log(n)) < \log(n) < n^{0.5} < n < n \log(n) < n^p < a^n < n! < n^n, c \in \mathbb{R}, p \geq 2, a > 1$
there is:

$$\begin{aligned} 2^n < n! < n^n \xrightarrow{\log n} \log 2^n < \log n! < \log n^n \Rightarrow n < \log n! < n \log n \xrightarrow{\log n} \log n < \log(\log n!) < \\ \log(n \log n) \Rightarrow \Theta(\log n) = \log n < \log(\log n!) < \log n + \log(\log n) = \Theta(\log n) \Rightarrow \\ \Rightarrow \log(\log n!) = \Theta(\log n) \end{aligned}$$

So, current order is: $a_3 < a_2, a_5 < a_4 < a_1$. Now we have to define the order between a_2 and a_5 .

Comparing a_2 and a_5 :

$$a_2 = \binom{n}{6} = \frac{n!}{(n-6)!6!} = \frac{n(n-1)(n-2)(n-3)(n-4)(n-5)}{6!} = \Theta(n^6)$$

$$a_5 = \log(5 \cdot n!) = \log(5) + \log(n!) = \Theta(n \log n)$$

So, $a_5 < a_2$ (because $n^5 > \log n \Rightarrow n^6 > n \log n$).

Final order is $a_3 < a_5 < a_2 < a_4 < a_1$

B

To order the functions I am using the basic-known order $c < \log^* n < \log(\log(n)) < \log(n) < n^{0.5} < n < n \log(n) < n^p < a^n < n! < n^n, c \in \mathbb{R}, p \geq 2, a > 1$

So, there is:

$$b_3 = e^n = \Theta(e^n)$$

$$b_2 = 5n^7 = \Theta(n^7)$$

$$b_4 = \log(n^5) = 5\log(n) = \Theta(\log n)$$

So, $b_4 < b_2 < b_3$

Now I have to order $b_1 = \frac{n}{\log n} = \Theta(\frac{n}{\log n})$ and $b_5 = \frac{\log n}{3n} = \Theta(\frac{\log n}{n})$

Firstly, I will order b_1 :

$$\lim_{n \rightarrow +\infty} \frac{b_2}{b_1} = \lim_{n \rightarrow +\infty} \frac{5n^7}{\frac{n}{\log n}} = \lim_{n \rightarrow +\infty} 5n^6 \log n = +\infty \Rightarrow b_1 = O(b_2)$$

$$\lim_{n \rightarrow +\infty} \frac{b_4}{b_1} = \lim_{n \rightarrow +\infty} \frac{\log(n^5)}{\frac{n}{\log n}} = \lim_{n \rightarrow +\infty} \frac{5 \log^2 n}{n} \stackrel{\infty/\infty}{=} \lim_{n \rightarrow +\infty} \frac{10 \log n}{n} \stackrel{\infty/\infty}{=} \lim_{n \rightarrow +\infty} \frac{10}{n} = 0 \Rightarrow b_1 = \Omega(b_4)$$

So, $b_4 < b_1 < b_2 < b_3$. Now, I will order b_5 .

$$\lim_{n \rightarrow +\infty} \frac{b_4}{b_5} = \lim_{n \rightarrow +\infty} \frac{\log(n^5)}{\frac{\log n}{3n}} = \lim_{n \rightarrow +\infty} \frac{15n \cancel{\log n}}{\cancel{\log n}} = +\infty \Rightarrow b_5 = O(b_4)$$

Final order is: $b_5 < b_4 < b_1 < b_2 < b_3$

C

To order the functions I am using the complexity classes of $\log_2(c_i)$, so only in the case where I end up with the same complexity for some $\log_2(c_i)$, I need compare the corresponding c_i to order them.

1.

$$\log(c_1) = \log(n!) = \Theta(n \log n)$$

2.

$$\log(c_2) = \log(2^{2^{2^n}}) = 2^{2^n} \log(2) = 2^{2^n} = \Theta(2^{2^n})$$

3.

$$\log(c_3) = \log(8n) = \log(2^3) + \log n = 3 + \log n = \Theta(\log n)$$

4.

$$c_4 = n^{\log(\log n)} = (\log n)^{\log n} \Rightarrow \log(c_4) = \log((\log n)^{\log n}) = \log n \cdot \log(\log n) = \Theta(\log n \cdot \log(\log n))$$

5.

$$c_5 = 2n^8 + n^{\log n} \xRightarrow{(n^{\log n} \text{ is increasing more})} \log(c_5) \approx \log(n^{\log n}) = \log^2 n = \Theta(\log^2 n)$$

Comparing $\log n, \log(\log n)$

For significant large n ($n > 2^2$)

For $n > 2^2 \Rightarrow \log n > \log 2^2$

$$\Rightarrow \log(\log n) > \log(2 \log 2)$$

$$\Rightarrow \log(\log n) > 1$$

$$\Rightarrow \log n \cdot \log(\log n) > \log n \text{ [For } n > 2^2 \Rightarrow \log n > 2 > 0]$$

$$\Rightarrow \log(c_4) > \log(c_3)$$

Comparing $\log n \cdot \log(\log n), \log^2 n$

$$n > \log n \Rightarrow \log n > \log(\log n)$$

$$(\text{for } n > 2 \Rightarrow \log n > 1) \Rightarrow \log n \cdot \log n > \log n \cdot \log(\log n) \Rightarrow \log^2 n > \log n \cdot \log(\log n)$$

$$\Rightarrow \log(c_5) > \log(c_4)$$

So, we have the following arrangement:

$$\log n < \log n \cdot \log(\log n) < \log^2 n < n \log n < 2^{2^n}$$

And so,

Final order is: $c_3 < c_4 < c_5 < c_1 < c_2$

Exercise 2

(a.) **True**

$$6n^2 + 3n - 6 = \Omega(7n^2) \iff \exists n_0 \in \mathbb{N}, \exists c \in \mathbb{R}^+ : (\forall n) 6n^2 + 3n - 6 \geq c7n^2$$

So, we will show that $\lim_{n \rightarrow \infty} \frac{6n^2 + 3n - 6}{7n^2} \geq c$ for some constant $c > 0$

$$\lim_{n \rightarrow \infty} \frac{6n^2 + 3n - 6}{7n^2} = \lim_{n \rightarrow \infty} \frac{6 + \frac{3}{n} - \frac{6}{n^2}}{7} = \frac{6}{7}$$

Since $\frac{6}{7}$ is a positive constant, we can choose $c = \frac{6}{7}$

Therefore, we have shown that $6n^2 + 3n - 6 = \Omega(7n^2)$.

(b.) **False**

For instance, if $f(n) = n$ and $g(n) = n^2$

Then, $(f + g)(n) = n + n^2$

but $f(n) = \Theta(n), g(n) = \Theta(n^2)$ and $(f + g)(n) = \Theta(n^2) \neq \Theta(n) = \Theta(\min(f(n), g(n)))$

(c.) **True**

$$\omega(f(n)) \cap o(f(n)) =$$

$$= \{g(n) \mid \forall c > 0 \exists n_0 \in \mathbb{N} \forall n \geq n_0 : g(n) < cf(n)\} \cap \{g(n) \mid \forall c > 0 \exists m_0 \in \mathbb{N} \forall m \geq m_0 : g(n) > cf(n)\}$$

$$= \{g(n) \mid \forall c > 0 \exists k_0 \in \mathbb{N} \forall k \geq k_0 : g(n) < cf(n) \wedge g(n) > cf(n)\} = \emptyset$$

Otherwise, I can prove it using the definitions taught in the course:

$$f(n) = \omega(g(n)) \Rightarrow \nexists (C, k) : f(n) = O(g(n)) \Rightarrow \nexists (C, k) : f(n) = o(g(n)). \text{ So, } \omega(f(n)) \cap o(f(n)) = \emptyset$$

(d). **False**

For instance, if $g(n) = \left(\frac{1}{2}\right)^n$, then:

$$2024 \cdot g(n) + 2^{2024} = 2024 \cdot \left(\frac{1}{2}\right)^n + 2^{2024} = 2024 \cdot 2^{-n} + 2^{2024} = O(1) \neq O\left(\left(\frac{1}{2}\right)^n\right)$$

$$\begin{aligned}\mathcal{L} &= \lim_{n \rightarrow \infty} \frac{2024g(n) + 2^{2024}}{g(n)} = \lim_{n \rightarrow \infty} \left(2024 + \frac{2^{2024}}{\left(\frac{1}{2}\right)^n}\right) = \lim_{n \rightarrow \infty} (2024 + 2^{2024-(-n)}) = \\ &= \lim_{n \rightarrow \infty} (2024 + 2^{2024+n}) = +\infty\end{aligned}$$

So, I can infer that:

$$\mathbb{L} = \infty \Rightarrow 2024g(n) + 2^{2024} = \omega(g(n)) \Rightarrow \nexists (C, k) : 2024g(n) + 2^{2024} = O(g(n))$$

(e.) **False**

[if I get $\mathcal{L} = \lim_{n \rightarrow \infty} \frac{n+2024\sqrt{n}}{n\sqrt{n}}$, then
 $n + 2024\sqrt{n} = \Omega(n\sqrt{n}) \iff 0 < \mathcal{L} \leq \infty$]

$$\mathcal{L} = \lim_{n \rightarrow \infty} \frac{n + 2024\sqrt{n}}{n\sqrt{n}} = \lim_{n \rightarrow \infty} \left(\frac{1}{\sqrt{n}} + \frac{2024}{n}\right) = 0 \Rightarrow n + 2024\sqrt{n} \notin \Omega(n\sqrt{n})$$

Exercise 3

A	B	O	o	Ω	ω	Θ
$\left(\frac{4}{5}\right)^n$	$\log n$	Yes	Yes	No	No	No
$n^4 - 3n^3$	$16^{\log n}$	Yes	No	Yes	No	Yes
$\sum_{k=0}^n \binom{n}{k}$	$2^{\frac{n}{3}}$	No	No	Yes	Yes	No
$\sum_{k=1}^n (k + \log k)$	$n^2 \log n$	Yes	Yes	No	No	No
$\binom{n}{n-3}$	$2024n^3$	Yes	No	Yes	No	Yes

Proof:

- $\left(\frac{4}{5}\right)^n \text{ — } \log n$

$$0 < \frac{4}{5} < 1 \Rightarrow \left(\frac{4}{5}\right)^n \searrow \mathbb{R} \text{ while } \log n \nearrow (0, +\infty)$$

$\{\log n\}$ is an increasing function in contrast with $\left\{\left(\frac{4}{5}\right)^n\right\}$ which is decreasing,

so it is obvious that $\left(\frac{4}{5}\right)^n = o(\log n)$ (and of course $\left(\frac{4}{5}\right)^n = O(\log n)$)

because $\lim_{n \rightarrow \infty} \left(\frac{4}{5}\right)^n = 0$ and $\lim_{n \rightarrow \infty} (\log n) = +\infty \Rightarrow \{\exists n_0 : \forall n \geq n_0 \log n > \left(\frac{4}{5}\right)^n\}$

I will also prove it with the mathematical definitions using the corresponding limit:

$$\lim_{n \rightarrow \infty} \frac{\left(\frac{4}{5}\right)^n}{\log n} = \lim_{n \rightarrow \infty} \left(\frac{4}{5}\right)^n \cdot \frac{1}{\log n} = 0 \cdot 0 = 0 \Rightarrow \left(\frac{4}{5}\right)^n = o(\log n) \wedge \left(\frac{4}{5}\right)^n = O(\log n).$$

- $n^4 - 3n^3 \sim 16^{\log n}$

There is:

$$n^4 - 3n^3 = \Theta(n^4)$$

$$16^{\log n} = n^{\log 16} = n^{\log 2^4} = n^{4 \log 2} = n^4 = \Theta(n^4)$$

Obviously $n^4 - 3n^3 = \Theta(16^{\log n} (= n^4))$

$$\lim_{n \rightarrow \infty} \frac{n^4 - 3n^3}{16^{\log n}} = \lim_{n \rightarrow \infty} \frac{n^4 - 3n^3}{n^{\log 2^4}} = \lim_{n \rightarrow \infty} \frac{n^4 - 3n^3}{n^4} = 1$$

From definitions also is implied that $n^4 - 3n^3 = \Theta(16^{\log n}) = O(16^{\log n}) \cap \Omega(16^{\log n})$

- $\sum_{k=0}^n \binom{n}{k} \sim 2^{\frac{n}{3}}$

In accordance with the binomial theorem, there is: $(a + b)^n = \sum_{k=0}^n \binom{n}{k} a^{n-k} b^k$

So, for $a = b = 1$, there is $\sum_{k=0}^n \binom{n}{k} 1^{n-k} 1^k = (1 + 1)^n \Rightarrow \sum_{k=0}^n \binom{n}{k} = 2^n$.

$$\lim_{n \rightarrow \infty} \frac{\sum_{k=0}^{\infty} \binom{n}{k}}{2^{\frac{n}{3}}} = \lim_{n \rightarrow \infty} \frac{2^n}{2^{\frac{n}{3}}} = \lim_{n \rightarrow \infty} 2^{n - \frac{n}{3}} = \lim_{n \rightarrow \infty} 2^{\frac{2n}{3}} = +\infty$$

So, $\sum_{k=0}^n \binom{n}{k} = \Omega(2^{\frac{n}{3}})$ and $\sum_{k=0}^n \binom{n}{k} = \omega(2^{\frac{n}{3}})$

- $\sum_{k=1}^n (k + \log k) \sim n^2 \log n$

$$\begin{aligned}\sum_{k=1}^n (k + \log k) &= \sum_{k=1}^n k + \sum_{k=1}^n (\log k) = \frac{n(n+1)}{2} + \log 1 + \log 2 + \dots + \log(n-1) + \log n \\ &= \frac{n(n+1)}{2} + \log(1 \cdot 2 \cdot \dots \cdot (n-1) \cdot n) = \frac{n(n+1)}{2} + \log(n!)\end{aligned}$$

Also, I can infer the following two:

$$\rightarrow \frac{n(n+1)}{2} = \frac{1}{2}n^2 + \frac{1}{2}n = \Theta(n^2)$$

$$\rightarrow \log n! = \Theta(n \log n)$$

(the first $\left(\frac{n(n+1)}{2}\right)$ is obvious and the second $(\log n!)$ is proven in previous exercises.)

Also, $n > \log n \Rightarrow n^2 > n \log n$. So, $\sum_{k=1}^n (k + \log k) = \Theta(n^2)$.

Now we have that, for $n > 2 \Rightarrow \log n > 1 \Rightarrow n^2 \log n > n^2 \Rightarrow$

$$\Rightarrow \sum_{k=1}^n (k + \log k) = o(n^2 \log n) \wedge \sum_{k=1}^n (k + \log k) = O(n^2 \log n)$$

Also, because $\sum_{k=1}^n (k + \log k) = o(n^2 \log n)$, it cannot be neither $\Omega(n^2 \log n)$,

nor $\omega(n^2 \log n)$, nor $\Theta(n^2 \log n)$

- $\binom{n}{n-3} \sim 2024n^3$

$$\binom{n}{n-3} = \frac{n!}{(n-3)!3!} = \frac{n(n-1)(n-2)}{3!} = \Theta(n^3)$$

$$2024n^3 = \Theta(n^3)$$

$$\text{So, } \binom{n}{n-3} = \Theta(2024n^3) \Rightarrow \binom{n}{n-3} = O(2024n^3) \wedge \binom{n}{n-3} = \Omega(2024n^3)$$

I will also prove it using a limit:

$$\begin{aligned}\lim_{n \rightarrow \infty} \frac{\binom{n}{n-3}}{2024n^3} &= \lim_{n \rightarrow \infty} \frac{\frac{n!}{(n-3)!3!}}{2024n^3} = \lim_{n \rightarrow \infty} \frac{\frac{n(n-1)(n-2)}{3!}}{2024n^3} = \lim_{n \rightarrow \infty} \frac{n^3 - 3n^2 + 2n}{2024 \cdot 3! \cdot n^3} = \lim_{n \rightarrow \infty} \frac{n^3(1 - \frac{3}{n} + \frac{2}{n^2})}{2024 \cdot 3! \cdot n^3} = \\ &= \lim_{n \rightarrow \infty} \frac{(1 - \frac{3}{n} + \frac{2}{n^2})}{2024 \cdot 3!} = \frac{1}{2024 \cdot 3!} \in \mathbb{R}^* \Rightarrow \\ &\Rightarrow \binom{n}{n-3} = \Theta(2024n^3) \wedge \binom{n}{n-3} = O(2024n^3) \wedge \binom{n}{n-3} = \Omega(2024n^3)\end{aligned}$$

Exercise 4

Algorithm 1

1st for: n steps $\Rightarrow O(n)$

2nd for: i steps with $\max_i = n \Rightarrow O(n)$

Calculation of arg-CALC function:

I observe that s is increasing with a rate $1 + 2 + 3 + \dots + i$ until $s > m$

$$\Rightarrow \sum_{i=1}^{s_i} (i) > m \Rightarrow \frac{s_i(s_i+1)}{2} > m \Rightarrow s_i^2 + s_i > 2m \Rightarrow s_i^2 + s_i + \frac{1}{4} > 2m + \frac{1}{4} \Rightarrow$$

$$(s_i + \frac{1}{2})^2 > 2m + \frac{1}{4} (> 0) \Rightarrow s_i + \frac{1}{2} > \sqrt{2m + \frac{1}{4}} \Rightarrow s_i > \sqrt{2m + \frac{1}{4}} - \frac{1}{2} \Rightarrow$$

The arg-CALC function does $O(\sqrt{n})$ steps, because

$$\text{there is: } s_i > \sqrt{2m + \frac{1}{4}} - \frac{1}{2} \wedge \max_m = \max_j = i \wedge \max_i = n \Rightarrow O(\sqrt{n})$$

So, overall complexity is $O(n) \cdot O(n) \cdot O(\sqrt{n}) = \mathbf{O(n^2\sqrt{n})}$

Algorithm 2

1st for: n steps $\Rightarrow O(n)$

2nd for: In this loop, j is increasing with a rate $j_k \leftarrow j_{k-1} + 2j_{k-1} = 3j_{k-1}$, (1) and so we can easily infer that the step of the loop increases at a geometric rate 3^j (It is easy also to observe that j gets the values $1(=3^0)$, $3(=3^1)$, $9(=3^2)$, $27(=3^3)$, $81(=3^4)$, ..., 3^k).

So, the loop ends when:

$$3^j > n \Rightarrow j > \log_3 n = \frac{1}{\log_2 3} \cdot \log_2 n \Rightarrow \text{2nd loop complexity is: } O(\log n)$$

Calculating complexity of arg-CALC function:

It is easy to observe that the loop inside the function is executed $2m - m + 1 = m + 1$ times. This is because it starts from $s = m$ and ends at $s = 2m$ (including the iteration where $s = 2m$) with step = 1. Also, $\max_m = \max_j = n \Rightarrow \max \text{ steps} = n + 1$, so the complexity of the procedure is: $O(n)$

Finally, the overall complexity of the algorithm is: $\mathbf{O(n) \cdot O(\log n) \cdot O(n) = O(n^2 \log n)}$

Exercise 5

The probability (p_{n+1}) to not find the element 'x' we are looking for is complementary to the sum of the probabilities of finding it in any of the n positions of the array.

$$\begin{aligned} p_{n+1} &= 1 - \sum_{i=1}^n p_i = 1 - \sum_{i=1}^{\frac{n}{2}} \left(\frac{2}{3n} \right) - \sum_{i=\frac{n}{2}+1}^{n-2} \left(\frac{1}{3(n-4)} \right) - \sum_{i=n-1}^n \left(\frac{1}{4} \right) = \\ &= 1 - \left(\frac{n}{2} \right) \left(\frac{2}{3n} \right) - \left(n - 2 - \frac{n}{2} - 1 + 1 \right) \left(\frac{1}{3(n-4)} \right) - 2 \cdot \frac{1}{4} = \\ &= 1 - \frac{1}{3} - \left(\frac{n}{2} - 2 \right) \left(\frac{1}{3(n-4)} \right) - \frac{1}{2} = \frac{1}{2} - \frac{1}{3} - \frac{n-4}{6(n-4)} = 0 \end{aligned}$$

So, the expected value of the linear search in the array is:

$$\begin{aligned}
\sum_{i=1}^n (i \cdot p_i) &= \sum_{i=1}^{\frac{n}{2}} i \cdot \left(\frac{2}{3n}\right) + \sum_{i=\frac{n}{2}+1}^{n-2} i \cdot \left(\frac{1}{3(n-4)}\right) + \sum_{i=n-1}^n i \cdot \left(\frac{1}{4}\right) = \\
&= \left(\frac{2}{3n}\right) \sum_{i=1}^{\frac{n}{2}} i + \left(\frac{1}{3(n-4)}\right) \sum_{i=\frac{n}{2}+1}^{n-2} i + \left(\frac{1}{4}\right) (n + (n-1)) = \\
&= \left(\frac{2}{3n}\right) \frac{(\frac{n}{2})(\frac{n}{2}+1)}{2} + \left(\frac{1}{3(n-4)}\right) \left(\sum_{i=1}^{n-2} i - \sum_{i=1}^{\frac{n}{2}} i\right) + \frac{2n-1}{4} = \\
&= \frac{n+2}{12} + \left(\frac{1}{3(n-4)}\right) \left(\frac{(n-2)(n-1)}{2} - \frac{\frac{n}{2}(\frac{n}{2}+1)}{2}\right) + \frac{2n-1}{4} = \\
&= \frac{n+2}{12} + \left(\frac{1}{3(n-4)}\right) \left(\frac{n^2-3n+2}{2} - \frac{n^2+2n}{8}\right) + \frac{2n-1}{4} = \\
&= \frac{n+2}{12} + \left(\frac{1}{3(n-4)}\right) \left(\frac{3n^2-14n+8}{8}\right) + \frac{2n-1}{4} = \\
&= \frac{n+2}{12} + \frac{(3n-2)(\cancel{n-4})}{24(\cancel{n-4})} + \frac{2n-1}{4} = \\
&= \frac{17n-4}{24} = \Theta(n)
\end{aligned}$$