Complexity exercise set #1

for the tutorial on
April 14, 2022

Exercises marked with an asterisk ($*$) may be handed in for grading and can earn you a small bonus[1] on the exam, provided you submit your solutions via Brightspace in PDF before **15:15 on Monday April 18**.

Exercise 1$*$ (10 pts.) Choose an exercise class via Brightspace before April 15.

Exercise 2 Two algorithms, `A` and `B`, exist to perform a certain task on a large dataset. Based on the documented asymptotic running times, $T_{\mathtt{A}}(n) = \mathcal{O}(n^2)$ and $T_{\mathtt{B}}(n) = \mathcal{O}(n^2 \lg(n))$, which one would you recommend?

Suppose that after implementation of the algorithms, measurements indicate that $T_{\mathtt{A}}(n) \approx 3\,n^2\,\mathrm{ms}$, and $T_{\mathtt{B}}(n) \approx 0.1\,n^2 \lg(n)\,\mathrm{ms}$.

Would you change your recommendation?

Solution. Since n^2 grows much slower than $n^2 \lg(n)$ — in fact, $n^2 \in o(n^2 \lg(n))$ — we'd choose algorithm `A` over `B` based only on the asymptotic bounds.

Given the concrete values for $T_{\mathtt{A}}$ and $T_{\mathtt{B}}$ we see that $T_{\mathtt{B}}(n) \leq T_{\mathtt{A}}(n)$ for all $n \leq 2^{30}$. Whence we would recommend algorithm `B`, because in practise `A` never outperforms `B`, because on an input of size 2^{30}, running `A` takes

$$T_{\mathtt{A}}(2^{30}) \;=\; 3458764513820540928\,\mathrm{ms} \;\approx\; 109 \cdot 10^6\,\text{years!} \quad \blacksquare$$

Exercise 3 Give an algorithm to compute x^n using $\Theta(\lg(n))$ multiplications.

(Hint: $x^m = (x^2)^{\lfloor \frac{m}{2} \rfloor}\, x^\sigma$ where $\sigma = 0$ if $2 \mid m$ and $\sigma = 1$ otherwise.)

Solution. The idea of the algorithm, which is commonly known as "exponentiation by squaring", is that we split the work done in the multiplication by splitting the exponent into powers of 2. Using the hint, we have that for n even

$$x^n = (x^2)^{\frac{n}{2}}$$

For n odd, we simply decrease the exponent by 1 and add an extra multiplication by x. We then apply this recursively, which can be expressed as:

$$x^n = \begin{cases} 1 & \text{if } n = 0 \\ \left(x^2\right)^{\frac{n}{2}}, & \text{if } n > 0 \text{ and } n = 2k \\ x \cdot \left(x^2\right)^{\frac{n-1}{2}} & \text{if } n > 0 \text{ and } n = 2k+1 \end{cases}$$

[1] For more details, see `https://cs.ru.nl/~awesterb/teaching/2022/complexity.html`.

Let $T(n)$ denote the number of multiplications used by the algorithm described by the equation above with n as input. Then T satisfies the recurrence relation

$$T(n) \;=\; T(\left\lfloor \frac{n}{2} \right\rfloor) \;+\; f(n),$$

where

$$f(n) \;=\; \begin{cases} 1 & \text{when } n \text{ is even} \\ 2 & \text{when } n \text{ is odd} \end{cases},$$

and so $f \in \Theta(1)$.

Hence $T \in \Theta(\lg(n)) = \Theta(\log(n))$ by case II of the Master theorem.

■

Exercise 4 Only one of the following statements is true. Which one is it?

1. $2^{\mathcal{O}(n)} = \mathcal{O}(2^n)$
2. $\sum_{n=1}^{N} \sum_{k=1}^{n} k = \frac{N(N+1)(N+2)}{6}$
3. $f(n) = \mathcal{O}(f(n)^2)$ for all f
4. If $f = o(n^2)$, then there is $\varepsilon > 0$ with $f = \mathcal{O}(n^{2-\varepsilon})$.
5. $(n+1)! = \mathcal{O}(n!)$
6. $\lceil a \rceil \lceil b \rceil = \lceil ab \rceil$ for all $a, b \in [0, \infty)$

Solution. Only statement number 2 is true; here's why:

1. Remember that, by definition, $\mathcal{O}(f)$ is the set of functions g such that there exists a positive constant c and a natural number n_0 such that for all $n > n_0$, $g(n) \leq c \cdot f(n)$. Suppose the statement is true, that is, $2^{\mathcal{O}(n)} = \mathcal{O}(2^n)$. Then, we must have that $2^{2n} \in \mathcal{O}(2^n)$. Hence, we would have a $c > 0$ and an $n_0 \in \mathbb{N}$ such that for all $n > n_0$:

   $$2^{2n} \leq c2^n \implies 2^n \cdot 2^n \leq c2^n \implies 2^n \leq c.$$

   But this is absurd, since $2^n > c$ for large enough n (namely $n > \lg(c)$.)

2. This statement is true. To prove this, we proceed in two main steps. First we prove, by induction on n, that $\sum\limits_{k=1}^{n} k = \frac{n(n+1)}{2}$, then again by induction we prove that $\sum\limits_{n=1}^{N} \sum\limits_{k=1}^{n} k = \sum\limits_{n=1}^{n} \frac{n(n+1)}{2} = \frac{N(N+1)(N+2)}{6}$, as required.

   (a) To prove $\sum\limits_{k=1}^{n} k = \frac{n(n+1)}{2}$ we proceed by induction.

   Base Case. Trivial, really it is trivial.

   Induction Step. Suppose that the statement hold for an arbitrary $n > 1$. We prove it for $n+1$, as follows:

   $$\begin{aligned} 1 + \cdots + n + (n+1) &= \frac{n(n+1)}{2} + (n+1) \\ &= \frac{(n+2)(n+1)}{2} \end{aligned}$$

(b) Secondly, we proceed by induction on N as follows:

Base Case. You guessed it. It's trivial this time around too.

Inductive Step.

$$\begin{aligned} 1+3+\cdots+\frac{N(N+1)}{2}+\frac{(N+1)(N+2)}{2} &= \frac{N(N+1)(N+2)}{6} \\ &+ \frac{(N+1)(N+2)}{2} \\ &= \frac{(N+1)(N+2)(N+3)}{6} \end{aligned}$$

3. For $f(n) = \frac{1}{n}$, we don't have $f(n) = \mathcal{O}\left(f(n)^2\right)$, because $\frac{1}{n^2} = o(\frac{1}{n})$.

4. To prove such implication is false we need to find a function f such that the hypothesis hold, that is, $f \in o(n^2)$, and the conclusion fails, that is, there is no $\epsilon > 0$ with $f \in \mathcal{O}\left(n^{2-\epsilon}\right)$.

   Notice that the set $o(f)$ is the set of functions g such that $\lim_{n\to\infty} \frac{g(n)}{f(n)} = 0$. Take $f(n) = n^2/\log(n)$. Then, using this limit we show that $f \in o(n^2)$.

$$\lim_{n\to\infty} \frac{n^2/\log n}{n^2} = \lim_{n\to\infty} \frac{1}{\log n} = 0.$$

   So, indeed, $f \in o(n^2)$.

   Secondly, we show that for any $\epsilon > 0$, $f \notin \mathcal{O}\left(n^{2-\epsilon}\right)$. Indeed, suppose towards a contradiction that $f \in \mathcal{O}\left(n^{2-\epsilon}\right)$ for some ϵ. Then there is $c > 0$, such that for all sufficiently large n, we have

$$\frac{n^2}{\log n} \le cn^{2-\epsilon} \implies n^2 \le cn^{2-\epsilon}\log n \implies n^\epsilon \le c\log(n),$$

   and thus $n^\epsilon \in \mathcal{O}(\log(n))$, which is well-known not to be the case.

5. Suppose that the statement is true. We would have that there exists a $c > 0$ and n_0, such that for all $n > n_0$:

$$(n+1)! \le cn! \implies (n+1)\cdot n! \le cn! \implies n+1 \le c$$

   Absurd.

6. Notice that if $a = 0.5$ and $b = 2$, we have $\lceil 0.5\rceil \lceil 2\rceil = 2$ and $\lceil 0.5\cdot 2\rceil = \lceil 1\rceil$.

■

Exercise 5$*$ (40 points) The following two claims are false. Find the error(s) in the supposed proofs, and offer a brief suggestion on how to avoid each error.

Claim 1. *We have* $1+\cdots+n = \frac{1}{2}(n^2+n+2)$ *for all* $n \in \mathbb{N}$.

Proof. We proceed by induction. Suppose that $1+\cdots+n = \frac{1}{2}(n^2+n+2)$ holds for some $n \in \mathbb{N}$; we'll show that then it holds for $n+1$ too. Indeed,

$$\begin{aligned} 1+\cdots+n+(n+1) &= \tfrac{1}{2}(n^2+n+2)+(n+1) \\ &= \tfrac{1}{2}(n^2+3n+4) \\ &= \tfrac{1}{2}((n+1)^2+(n+1)+2). \end{aligned}$$

Whence $1+\cdots+n = \frac{1}{2}(n^2+n+2)$ for all $n \in \mathbb{N}$. □

Solution. The first mistake can be found already in the basis of induction. Notice that when $n = 1$ we would have that $1 = 2$. A good advice when proving things by induction: always check the induction base case before stating it is trivial. ■

Claim 2. *If $T\colon \mathbb{N} \to [0,\infty)$ obeys the recurrence relation $T(n) = T(n-1)+n$, then $T(n) = \mathcal{O}(n)$.*

Proof. By induction on n: *(induction step)* if $T(n) = \mathcal{O}(n)$ for some $n \in \mathbb{N}$, then $T(n+1) = T(n)+n+1 = \mathcal{O}(n)+\mathcal{O}(n) = \mathcal{O}(n)$; *(base case)* we have $T(0) = \Theta(1) \leq \mathcal{O}(n)$. □

Solution. The mistake is in the inductive step as it doesn't make sense to say that $T(n) = \mathcal{O}\left(n\right)$ for some specific $n \in \mathbb{N}$; such a statement only makes sense for all n. (Remember, $T(n) = \mathcal{O}\left(n\right)$ means $T \in \mathcal{O}\left(n\right)$.) When using syntactic sugar like $\mathcal{O}$-notation, it's easy to make false or non-sensical steps that look plausible; always make sure you understand what the statement actually means. ■

Exercise 6* (50 points) For each of the following recurrence relations, determine if the Master theorem (see §4.5 of the book) can be applied, and if so, write down the asymptotic solution (e.g. $T(n) = \Theta(n \lg n)$) without further explanation. If the Master theorem cannot be applied, briefly indicate why.

1. $T(n) = 9\,T(\frac{n}{3}) + n^2$
2. $T(n) = 9\,T(\frac{n}{3}) + n$
3. $T(n) = 9\,T(\frac{n}{3}) + n^3$
4. $T(n) = 9\,T(\frac{n}{3}) + n^2 \lg n$
5. $T(n) = \sqrt{3}\,T(\frac{n}{3}) + \lg(\lg n)$
6. $T(n) = 143640\,T(\frac{n}{70}) + n^2$
7. $T(n) = 2^n\,T(\frac{n}{2}) + n^{2n}$
8. $T(n) = 41592\,T(\frac{n}{31}) + n^{\pi}$
9. $T(n) = b^y\,T(\frac{n}{b}) + n^y$ for some $b \in (1,\infty)$ and $y \in [1,\infty)$
10. $T(n) = T(\frac{n}{2}) + 3^{\lceil \log_3(n) \rceil}$

Solution.

1. $\Theta(n^2 \log(n))$
2. $\Theta(n^2)$

3. $\Theta(n^3)$

4. The Master theorem cannot be applied.

   We obviously can't apply case 1.

   Case 2, would require that $n^2 \log n = \Theta\left(n^2\right)$, which is not so.

   Case 3, fails because $n^2 \lg n \notin \Omega(n^{2+\varepsilon})$ for all $\varepsilon > 0$. (It also fails in the regularity condition: We need that

$$\begin{aligned} 9(\frac{n}{3})^2 \log(n/3) &\leq cn^2 \log n \\ \log(n/3) &\leq c \log n \\ 1 - \frac{\log 3}{\log n} &\leq c \end{aligned}$$

   By taking $n \to \infty$, we see that we would need $1 \leq c$, which contradicts the requirement that $c < 1$.)

5. $\Theta(\sqrt{n})$ (Note: $\log_3(\sqrt{3}) = \frac{1}{2}$.)

6. $\Theta(n^{\log_{70}(143640)})$

   (Note: since $\log_{70}(143640) \approx 2.8 > 2$, we see that we're in case 1, but $\Theta(n^{2.8})$ is, of course, not the correct answer.)

7. The Master theorem does not apply, because a is not constant.

8. $\Theta(n^\pi)$ (Note: $\log_{31}(41592) \approx 3.097 < \pi$.)

9. $\Theta(n^y \log(n))$

10. The Master theorem does not apply, because $\log_2(1) = 0$ and $3^{\lceil \log_3(n) \rceil} \in \Omega(n)$ put us in case 3, but the 'regularity condition' does not hold.

    The regularity condition, in the form presented by the book, is that there should be be some $c > 0$ and N such that for all $n \geq N$

$$a\, f(n/b) \ \leq \ cf(n).$$

    So for the regularity condition to hold in our situation, there must be some $c < 1$ and N such that for all $n \geq N$

$$3^{\lceil \log_3(n) - \log_3(2) \rceil} \ \leq \ c3^{\lceil \log_3(n) \rceil}$$

    But this is impossible, because $\lceil \log_3(n) \rceil = \lceil \log_3(n) - \log_3(2) \rceil$ for all n of the form $n = 3^k$ (because $0 < \log_3(2) < 1$, and so $\lceil \log_3(n) - \log_3(2) \rceil = \lceil k - \log_3(2) \rceil = \lceil k \rceil = \lceil \log_3(n) \rceil$.)

■