DMA 2021

– Notes for Week 3 –

1 Sequences

We will think of a **sequence** as an infinite, linearly ordered collection of numbers. An example would be

$$1, 4, 9, 16, 25, 36, 49, 64, 81, 100, \ldots \tag{1}$$

or

$$0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, \ldots \tag{2}$$

We will often use the notation (a_n) to describe the elements of a sequence and refer to a_n as the n'th element of the sequence. The number n is an index and we often start a sequence at index 0 or 1.

We can succinctly specify a sequence via a function f that is defined on, say, the positive integers $\{1, 2, 3, 4 \ldots\}$. In that case, we can write an explicit expression for a sequence by setting

$$a_n = f(n).$$

For example, the sequence in (1) is defined by the function $f(x) = x^2$ and we would write the sequence as

$$b_n = n^2 \text{ for } n \geq 1.$$

In a similar way, we can explicitly specify sequence (2) as follows:

$$c_n = n - 2\left\lfloor \frac{n}{2} \right\rfloor \text{ for } n \geq 0.$$

We can also define sequences **recursively**. A recursive definition works by explicitly specifying some of the first elements in a sequence and then defining the rest by referring back to previous elements one or more indices away. One example would be to define

$$\begin{aligned} c_0 &= 0 \\ c_1 &= 1 \\ c_n &= c_{n-2} \text{ for } n \geq 2 \end{aligned}$$

The above recovers the sequence (2) since we can use the rule $c_n = c_{n-2}$ to convince ourselves that c_2 must be equal to $c_0 = 0$ and that c_3 must be equal to $c_1 = 1$ etc.

Example 1. We define a sequence f_n by setting $f_0 = 1$ and

$$f_n = nf_{n-1} \text{ for } n \geq 1.$$

The first elements of the sequence are

$$1, 1, 2, 6, 24, 120, 720, 5040$$

This sequence is referred to as the **factorial sequence** and we usually write f_n as $n!$. Note that

$$n! = 1 \cdot 2 \cdot 3 \cdots n.$$

A special class of recursively defined sequences consists of **series** (sums of sequences). A series is defined from some sequence (a_n) by summing its elements. If n_0 is the first index of a sequence (a_n), we define the series (s_n) recursively as

$$\begin{aligned} s_{n_0} &= a_{n_0} \\ s_n &= s_{n-1} + a_n \text{ for } n > n_0 \end{aligned}$$

We will often write

$$s_n = a_{n_0} + a_{n_0+1} + \cdots + a_n$$

or

$$s_n = \sum_{k=n_0}^{n} a_k$$

to define a series.

A **summation formula** is the explicit expression of a series.

Example 2. Let (a_n) be the constant sequence 1, i.e. $a_n = 1$, $n \geq 1$. The series of (a_n) is then given by the sequence

$$1, 2, 3, 4, 5, 6, 7, 8, 9, \ldots$$

and we have obtained our first summation formula:

$$\sum_{k=1}^{n} 1 = n \tag{3}$$

The derivation and proof of summation formulas are very connected to the concept of **mathematical induction**, which we will return to later in the course. For now, we postulate some commonly used summation formulas without proofs:

Theorem 3.

$$\begin{aligned}\sum_{k=1}^{n} k &= \frac{n^2+n}{2}\\ \sum_{k=1}^{n} k^2 &= \frac{2n^3+3n^2+n}{6}\\ \sum_{k=1}^{n} c^k &= \frac{c^{n+1}-c}{c-1}\end{aligned}$$

The last formula is valid for all $c \neq 1$.

When $c = 1$ in the last summation formula, one can use formula from (3) instead.

As we will discover in the exercises, one can often go a long way by combing the four summation formulas above with the general sum rules:

Theorem 4.

$$\begin{aligned}\sum_{k=1}^{n}(a_k+b_k) &= \sum_{k=1}^{n} a_k + \sum_{k=1}^{n} b_k\\ \sum_{k=1}^{n} c a_k &= c\sum_{k=1}^{n} a_k\\ \sum_{k=1}^{n} a_k &= \sum_{k=1}^{m-1} a_k + \sum_{k=m}^{n} a_k\end{aligned}$$

2 Functions and graphs

In this section, we recall the concept of **functions** and introduce some relevant functions for the course.

- A **power function** is of the form

$$f(x) = x^a$$

where the exponent a is a constant. The expression is well-defined for all x when $a \in \{1, 2, 3, \dots\}$ but it is sometimes necessary to restrict to $x > 0$ for other values of a (recall the notation $\sqrt{x} = x^{1/2}$).

- An **exponential function** is of the form

$$f(x) = b^x$$

where the base $b > 0$ is a constant. The expression is well-defined for all $x \in \mathbb{R}$.

- A **logarithmic function** is of the form

$$f(x) = \log_b(x)$$

where the base $b > 1$ is a constant. The expression is well-defined for all $x > 0$. Logarithmic functions are the inverse functions of exponential functions with the same base. Thus,

$$b^x = y \iff \log_b(y) = x$$

Theorem 5 (Properties of logarithms)**.** *For all $a, b, c > 0$ and all $r \in \mathbb{R}$ we have*

$$\log_c(ab) = \log_c(a) + \log_c(b) \tag{4}$$

$$\log_b(a^r) = r \log_b(a) \tag{5}$$

$$\log_b(a) = \frac{\log_c(a)}{\log_c(b)} \tag{6}$$

where, in each equation above, logarithm bases are not 1.

- The **absolute value** is given by

$$|x| = \begin{cases} x & \text{if } x \geq 0 \\ -x & \text{if } x < 0 \end{cases}$$

- **Floor** is the function $\lfloor x \rfloor$, which rounds a real number x to the largest integer less than or equal to x. Thus,

$$\left\lfloor \frac{1}{2} \right\rfloor = 0 \qquad \left\lfloor -\frac{1}{2} \right\rfloor = -1 \qquad \lfloor \pi \rfloor = 3 \qquad \lfloor 7 \rfloor = 7$$

We can define the **ceiling** function in a similar way; $\lceil x \rceil$ is the smallest integer larger than or equal to x.

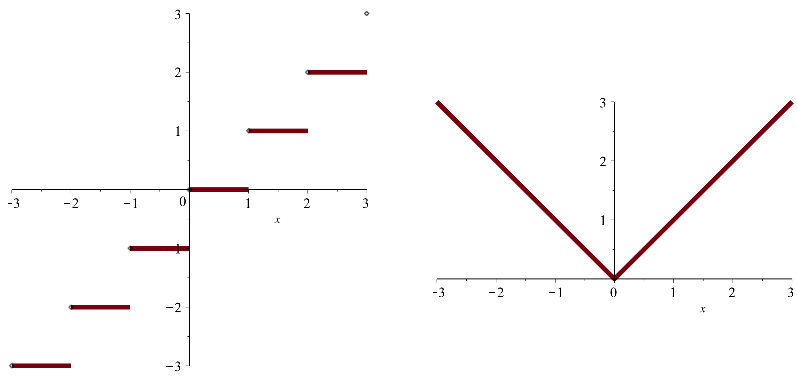

Figure 1: Graphs of $\lfloor x \rfloor$ and $|x|$

In computer science, it is often convenient to combine other functions with the floor and ceiling functions. In order to perform calculations in such situations, it is advantageous to notice that

$$\lfloor x \rfloor = n \Longleftrightarrow n \leq x < n + 1$$

and

$$\lceil x \rceil = n \Longleftrightarrow n - 1 < x \leq n.$$

Example 6. We have that

$$\lceil \log_2 x \rceil = n$$

when

$$n - 1 < \log_2 x \leq n,$$

which gives

$$2^{n-1} < 2^{\log_2 x} \leq 2^n$$

or just

$$2^{n-1} < x \leq 2^n$$

Thus, we see that $\lceil \log_2 x \rceil = n$ exactly when $2^{n-1} < x \leq 2^n$.

Recall that functions that map numbers to numbers can be shown in a **graph**, which is often the easiest way to understand the most important properties of a function. See figures 1 and 2 for some examples of graphs.

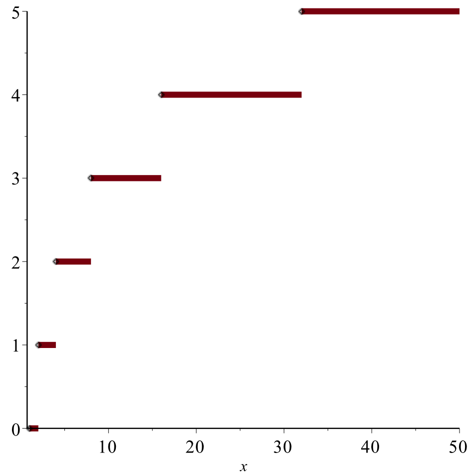

Figure 2: The graph of $\lfloor \log_2 x \rfloor$.

3 Asymptotic growth of functions

3.1 Collection of definitions

In this section we collect the definitions regarding asymptotic growth of functions. Further explanations are available in the [CLRS] book and during lectures.

Definition 7. We say that a function $f : \mathbb{R}^+ \to \mathbb{R}$ is *asymptotically positive* if there exists $x_0 \in \mathbb{R}^+$ such that $0 < f(x)$ for all $x \geq x_0$.

We will also apply the above definition for functions that are defined on some subset of the positive reals. A common choice of such a subset will be positive integers $\mathbb{Z}^+$ or natural numbers $\mathbb{N}$.

Definition 8 (Asymptotic notation)**.** Let f and g be asymptotically positive functions.

- We say that $f(x)$ *is* $O(g(x))$ if there exists a constant $c > 0$ and x_0 such that
$$f(x) \leq cg(x)$$
for all $x \geq x_0$. Think of this as **"g grows at least as fast as f asymptotically"**.

- We say that $f(x)$ *is* $\Theta(g(x))$ if f is $O(g(x))$ and $g(x)$ is $O(f(x))$. Think of this as **"g and f grow at the same rate asymptotically"**.
- We say that $f(x)$ *is* $o(g(x))$ if for any constant $c > 0$ we can find x_0 such that
$$f(x) < cg(x)$$
for all $x \geq x_0$. Think of this as **"g grows (strictly) faster than f asymptotically"**.

One can use the definition of big-O to show that big-Θ can equivalently be defined in the following manner.

Definition 9 (Second definition of big-Θ)**.** Let f and g be asymptotically positive functions. We say that $f(x)$ *is* $\Theta(g(x))$ if there exist constants $c_1, c_2 > 0$ and x_0 such that

$$c_1 g(x) \leq f(x) \leq c_2 g(x)$$

for all $x \geq x_0$.

It can be useful to *informally*(!) think of the above defined asymptotic notions as being analogous to comparison of numbers. Specifically,

$$\begin{array}{lll} f(x) \text{ is } O(g(x)) & \text{is like} & \text{“}f \leq g\text{”} \\ f(x) \text{ is } o(g(x)) & \text{is like} & \text{“}f < g\text{”} \\ f(x) \text{ is } \Theta(g(x)) & \text{is like} & \text{“}f = g\text{”} \end{array}$$

One must be careful and only use the above analogy to build intuition as some properties that hold for comparison of numbers do not carry over for functions. For instance, if a and b are numbers, then we have that $a \leq b$ or $b \leq a$. For functions, however, we can have a situation where f is not $O(g)$ and g is not $O(f)$. Can you think of such an example?

Both little-o and big-O give upper bounds. Intuitively, little-o gives a strict upper bound while big-O gives an upper bound that is potentially not strict. More formally, we have the following.

Theorem 10. *Let $f, g : \mathbb{R}^+ \to \mathbb{R}$ be asymptotically positive functions such that $f(x)$ is $o(g(x))$. Then we have that*

1. *$f(x)$ is $O(g(x))$ and*
2. *$g(x)$ is* not *$O(f(x))$.*

3.2 Collection of rules

Theorem 11. *Let $f, g, h, p : \mathbb{R}^+ \to \mathbb{R}$ be asymptotically positive functions.*

(R1) **"Overall constant factors can be ignored"**
If $c > 0$ is a constant then $cf(x)$ is $\Theta(f(x))$.

(R2) **"For polynomials only the highest-order term matters"**
If $p(x)$ is a polynomial of degree d, then $p(x)$ is $\Theta(x^d)$.

(R3) **"The fastest growing term determines the growth rate"**
If $f(x)$ is $o(g(x))$ then $c_1 g(x) + c_2 f(x)$ is $\Theta(g(x))$, where $c_1 > 0$ and $c_2 \in \mathbb{R}$ are constants.

(R4) **"Logarithms grow faster than constants"**
If $c > 0$ is a constant then c is $o(\log_a(x))$ for all $a > 1$.

(R5) **"Powers (and polynomials) grow faster than logarithms"**
$\log_a(x)$ *is $o(x^b)$ for all $a > 1$ and $b > 0$.*

(R6) **"Exponentials grow faster than powers (and polynomials)"**
x^a is $o(b^x)$ for all a and all $b > 1$.

(R7) **"Larger powers grow faster"**
x^a is $o(x^b)$ if $a < b$.

(R8) **"Exponentials with a bigger base grow faster"**
a^x is $o(b^x)$ if $0 < a < b$.

Informally, we can summarize rules (R4)–(R6) from above as

Constants < Logarithms < Polynomials < Exponentials

Example 12. Let us use the above rules to show that $2^x + x$ grows asymptotically faster than $3x^2 + 5x$ (*i.e.* $3x^2 + 5x$ is $o(2^x + x)$).

(1) First, use *(R2)* to conclude that $\boldsymbol{3x^2 + 5x}$ **and** $\boldsymbol{x^2}$ **grow at the same rate** asymptotically (*i.e.* $3x^2 + 5x$ is $\Theta(x^2)$).

(2) Use *(R6)* observe that $\boldsymbol{2^x}$ **grows faster** $\boldsymbol{x^2}$ (*i.e.* x^2 is $o(2^x)$).

(3) Combine[1] the bold statements from (1) and (2), to conclude that $\boldsymbol{2^x}$ **grows faster than** $\boldsymbol{3x^2 + 5x}$ (*i.e.* $3x^2 + 5x$ is $o(2^x)$).

[1]Note that even though we don't have a formal rule saying this, we are using the fact that if f_1 grows faster than f_2 which grows at the same rate as f_3 then f_1 grows faster than f_3. The formal statement is that if f_2 is $o(f_1)$ and f_2 is $\Theta(f_3)$ then f_3 is $o(f_1)$.

(4) Use *(R6)* again to observe that 2^x grows faster than x (*i.e.* x is $o(2^x)$)). By *(R3)*, we can now say that **$2^x + x$ grows at the same rate as 2^x** (*i.e.* $2^x + x$ is $\Theta(2^x)$).

(5) Finally, combine the bold statements from (3) and (4), to get that **$2^x + x$ grows faster than $3x^2 + 5x$** asymptotically (*i.e.* $3x^2 + 5x$ is $o(2^x + x)$).