Writing in Computer Science: Math

Javier Larrosa

UPC Barcelona Tech

.

Writing Math

Consider the following three different ways to say the same thing,

1. Let C be a set of cities. The population size of a city $c \in C$ is noted $size(c)$. The average size of cities in C is,

$$m = \frac{\sum_{c \in C} size(c)}{|C|}$$

2. Suppose we have a set of n cities each one identified with an index $1 \leq i \leq n$. The population of the i-th city is noted s_i. The average size of the cities is,

$$m = \frac{1}{n} \sum_{i=1}^{n} s_i$$

3. Let C be a set of cities where each city is a pair $c = (id, s)$ with id being its name and s being its population. The average size of the cities in C is,

$$m = \frac{\sum_{(n,s) \in C} s}{n}$$

where $n = |C|$.

Symbols

Mathematical expressions are made out of symbols. A **symbol** is a letter in a given style. The usual symbol styles are:

- lower case italics: x, y, p, q
- lower case boldface: $\mathbf{x}, \mathbf{y}, \mathbf{p}, \mathbf{q}$
- upper case italics: P, Q, R
- upper case boldface: $\mathbf{P}, \mathbf{Q}, \mathbf{R}$
- Calligraphy: $\mathcal{P}, \mathcal{Q}, \mathcal{R}, \mathcal{X}$
- Greek alphabet (lower and upper case) $\alpha, \beta, \Delta, \Gamma$

Plain text

Warning

Plain text is never a valid style in mathematics because it is important to distinguish between the English and the Math. (e.g. It is wrong to write that n is the number of elements in set C; it is right to write that n is the number of elements in set C).

Styles

Warning

In English, you can write a word in plain text, *italics* or **boldface**, lower or UPPER case and it is still the same word.
In math writing, if you define a symbol (e.g. n) the definition does not extend to other fonts or styles (e.g. $\mathrm{n}, \mathbf{n}, N, \mathcal{N} \ldots$). In other words, you cannot say that the size of a given set is n and later on use **n** as a synonym because they are different symbols.

Styles

Hint

Do not to use the same letter in different, but similar, styles to denote different things, because it may be confusing.
For example, it is a bad idea to denote the size of a set of numbers m and their mean m.

The previous rule does not apply between lower and upper case. For instance, it is not unusual to name a set M and an arbitrary element m.

I would avoid naming a set C and an arbitrary element c because they look much more similar.

Consistency

Hint

Be consistent with the notation.
For instance, I have seen many papers about logic that use upper-case letters to denote formula (e.g. F, G) and many others that use lower-case greek letters (e.g. α, γ). This is Okay, but do not mix them in the same paper.

Notational Entity

Notational Entity

Upper case has greater *notational entity* than lower case, and calligraphic upper case has greater notational entity than upper case. Additional **notational entity** can also be obtained using boldface.

A notation of greater entity is to be used to name more complex elements.

Example:

- Let P be a set of integers
- Let p be an integer
- Let $\mathcal{P}$ be a partition of a set of integers

Modifiers

Modifiers

Sometimes we need to name several mathematical objects that are very related. To emphasize their relation one can use the same letter with a **modifier**. Modifiers are sub-indexes, super-indexes, accents and priming. These elements can be very useful in some situations, but one should use with care.

- Accents and priming: useful to denote variations (e.g. derivative $f(x), f'(x)$) or special elements (e.g. optimum x^*)
- Indexes are useful for enumeration (e.g. $X = \{x_1, x_2, \ldots, x_i, \ldots, x_n\}$) or identification ($s_i$ and w_i are the size and wealth of city i)

Words as symbols

Warning

For complex definitions (specially functions) words are sometimes a good idea, because they are easy to memorize (e.g. ,...).

- E.g.sin for *sinus*, max for *maximum*
- E.g. $MC(G)$ set of maximal cliques of a graph G

Computer Scientists use words too often (probably because of our habit to name variables and algorithms). So it is good advise to restrict our notation to single letter symbols unless there is a good reason against it.

Notational Conventions

Some usual naming conventions:

- **scalars** a, b, x, y, p, q
- **indexes** i, j, k
- **vectors** (i.e, fixed-size ordered sequences) $\vec{v} = (v_1, v_2, \dots, v_n)$.
- In Computer Science we often talk about **arrays** and **lists**, which are finite (but not necessarily fixed-size) ordered sequences. Usual notation for arrays is $s = (s[1], s[2], \dots, s[n])$, and for lists is $l = (l_1, l_2, \dots, l_n)$
- **sets**, **multisets** $S = \{x_1, x_2, \dots, x_n\}$
- **tuples** $P = (A, B)$
- **functions** f, g, h
  - $f : A \longrightarrow B$ (that means $x \in A$, then $f(x) \in B$)
  - $f : A \times A \longrightarrow B$ (that means $x, y \in A$, then $f(x, y) \in B$)

Some hints from Donald Knuth

- Symbols in different formulas must be separated by words
  - **Bad**: Consider $S_q, q < p$.
  - **Good**: Consider S_q, where $q < p$.
- Do not start a sentence with a symbol.
  - **Bad**: $x^n - a$ has n distinct zeroes.
  - **Good**: The polynomial $x^n - a$ has n distinct zeroes.
- Do not use the symbols $\Rightarrow, \forall, \exists, \ldots$ (replace them by the corresponding words)

Some hints from Donald Knuth

- Facilitate memorization (e.g. $\vec{v}$ for vectors, p for points, S for sets, ...)
- Avoid **unnecessary subscripts** (e.g. x and y is better than x_1 and x_2, or x and x')
- Avoid **piling subscripts** (e.g. $x_{i'}^{p_j}$)
- Avoid **unfamiliar symbols** (e.g. Ξ, Θ) and **unfamiliar accents** (e.g. $\hat{a}$, $\check{a}$, $\dot{a}$)

Some hints from Donald Knuth

On notation consistency

Do not use the same notation for two different things.
Use the same notation for the same thing when it appears in several places.

For example, do not say "A_j for $1 \leq j \leq n$" in one place and "A_k for $1 \leq k \leq n$" in another.
Choose and use always the same indexes when varying on the same ranges

Some hints from Donald Knuth

On notation revision

When you first establish notation you may not be fully aware of what are you going to need from it. So be ready (and willing) to make changes

- Is a bad idea to start with a definition like "let $X = \{x_1, x_2, \ldots, x_n\}$" if you are going to need subsets of X. Also you will need to be speaking of elements x_i and x_j all the time.
- Do not name the elements of a set X unless necessary. Thus you can alway refer to elements x and y of X, and a subset Y of X.

Some hints from Donald Knuth

- Do not use **programming language notation** in your mathematical writing (e.g. $*$ for multiplication, or $:=$ for assignment)

Example

2 Definitions and Preliminaries

Note that all the lemmas presented in this section can be found in a book on treewidth by Ton Kloks [Kloks, 1994]. All the graphs used in this paper are finite, undirected, connected and simple. We denote an undirected graph by $G(V_G, E_G)$ where V_G is the set of vertices of the graph and E_G is the set of edges of the graph. The *neighborhood* of a vertex v denoted $N_G(v)$ is the set of vertices that are neighbors of v. A chord of a cycle C is an edge not in C whose endpoints lie in C. A *chordless cycle* in G is a cycle of length at least 4 in G that has no chord.

Definition 2.1. *A graph is* **chordal** *or triangulated if it has no chordless cycle.*

A graph $G' = (V', E')$ is called a *subgraph* of G if $V' \subseteq V$ and $E' \subseteq E$. If $W \subseteq V$ is a subset of vertices then $G[W]$ denotes the subgraph induced by W. A clique is a graph G that is completely connected. A *triangulation* of a graph G is a graph H such that G is a subgraph of H, H has the same set of vertices as G and H is chordal.

Definitions

In Scientific Writing, we often have to write **definitions**

- Definitions can be given in **words** or in **mathematical writing**
- Sometimes we give both
- Sometimes we add an example

Example: mean, median, mode

in words

Given a list of numerical values,

- The **mean** is the sum of the values divided by the number of values
- The **median** is the "middle" value after sorting the list. If the number of values is even, the median is the mean of the two "middle" values.
- The **mode** is the value that occurs most often. In case of ties, there is no mode.

Mean (math writing)

Mean

Given n numerical values $x_1, x_2, \ldots, x_n$, the **mean** is

$$\frac{\sum_{i=1}^{n} x_i}{n}$$

Median (math writing)

Median

Given n **sorted** numerical values $x_1, x_2, \ldots, x_n$, the **median** is,

- $x_{(n/2)+1}$, if n is even
- $\frac{x_{n/2}+x_{(n/2)+1}}{2}$, if n is odd

Mode (math writing)

Mode

Consider a list of numerical values $L = (x_1, x_2, \ldots, x_n)$, possibly with repetitions. Let $Y = \{y_1, y_2, \ldots, y_m\}$ the set of distinct values appearing in L. Let f_i (with $1 \leq i \leq m$) be the number of occurrences of y_i in L,

$$f_i = |\{j|\ x_j = y_i, 1 \leq j \leq n\}|$$

The **median** of L is y_{i^*} where i^* is defined as,

$$i^* = \mathtt{argmax}_{i=1}^{m}\{f_i\}$$

Example: Auction (first version)

Given a set of items S, a **bid** is a pair (B, v) with $B \subset S$ and v being a positive number. Element B represents the items requested by the bid and v is the offering price.
An **auction** $\mathcal{A}$ is a set of bids. Let $\mathcal{S}$ be a subset of $\mathcal{A}$. We say that $\mathcal{S}$ is **feasible** if there are no items appearing in more than one of its bids,

$$\cap_{(B,v)\in\mathcal{S}} B = \emptyset$$

The **value** of $\mathcal{S}$, noted $f(\mathcal{S})$, is the sum of its offering prices,

$$f(\mathcal{S}) = \sum_{(B,v)\in\mathcal{S}} v$$

The goal is to find the feasible subset of $\mathcal{A}$ with the highest value.

Example: Auction (second version)

Given a set of items S, a **bid** is a pair $b = (B, v)$ with $B \subset S$ and v being a positive number. Element B represents the items requested by the bid and v is the offering price.

An **auction** is a set of bids $b_1, b_2, \ldots, b_n$, with $b_i = (B_i, v_i)$. Let I be a subset of $\{1, 2, \ldots, n\}$ which represents a subset of the auction. We say that I is **feasible** if there are no items appearing in more than one of its bids,

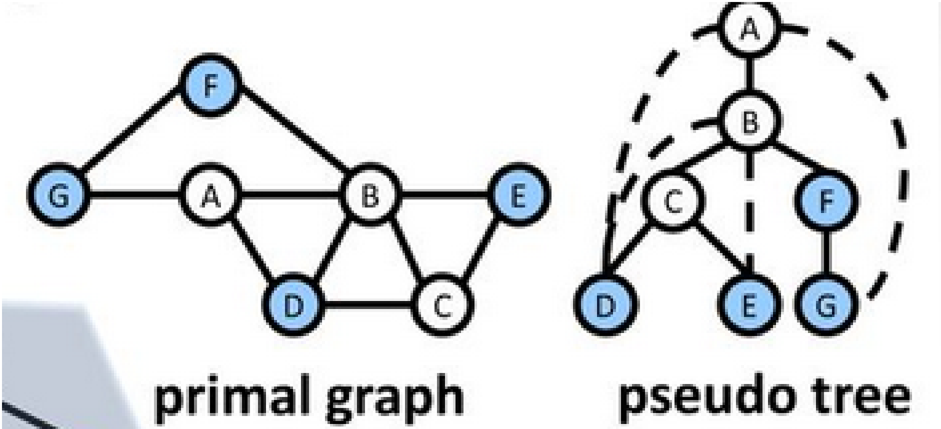
$$\forall_{i,j \in I} B_i \cap B_j = \emptyset$$

The **value** of I, noted $f(I)$, is the sum of its offering prices,

$$f(I) = \sum_{i \in I} v_i$$

The goal is to find the feasible set I with the highest value.

Example: Pseudo-tree



[Freuder and Quinn 1985]

Example: Pseudo-tree

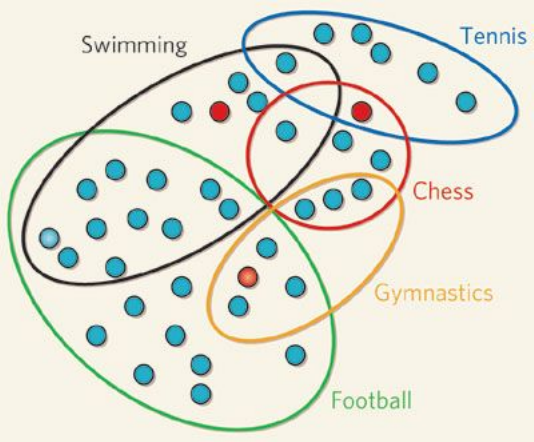
A **tree** is a cycle-free graph.... The **ancestors** of a vertex u are all the vertices in the **path** from u to the tree **root**.

Definition

We say that a tree $T = (V', E')$ with root $r \in V'$ is a **pseudo-tree** of a graph $G = (V, E)$ if: both have the same set of vertices (e.g. $V = V'$) and every edge in G does not cross branches in T (e.g. for all $(u, v) \in E$ we have that $u \in anc(T, v)$ or $v \in anc(T, u)$

Obs: I do not like the prime appearing before the non-prime

Example: Hitting Set



Example: Hitting Set

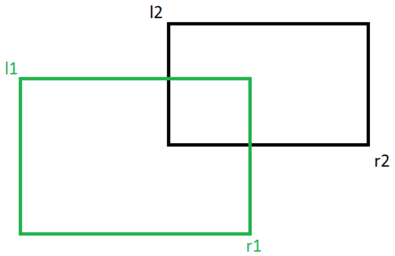
Hitting Set

Let $S_1, S_2, \ldots, S_n$ be subsets of a set U. We say that a set $H \subseteq U$ is a **hitting set** if every S_i has at least one element in H,

$$\forall_{1 \leq i \leq n}, |S_i \cap H| > 0$$

In the **hitting set problem** the goal is to find a hitting set of minimum size.

Example: Non-overlapping Rectangles



Example: Non-overlapping Rectangles

We represent a rectangle as a pair of two-dimension points $r = ((x_1, y_1), (x_2, y_2))$ where the first point indicates the top-left corner and the second point indicates the bottom-right corner.
Two rectangles $r = ((x_1, y_1), (x_2, y_2))$ and $r' = ((x'_1, y'_1), (x'_2, y'_2))$ **do not overlap** iff $(x_2 < x'_1)$ or $(x'_2 < x_1)$ or $(y_1 < y'_2)$ or $(y'1 < y_2)$