

Pliage : recto visible (autre face), traits gris rentrants et traits rouges saillants. Découper selon le trait rouge entre les deux ●, puis achever le pliage.

Prérequis

Tous les modules qui précèdent celui-ci (notamment CODAGE et INF1), ainsi que leurs prérequis, sont des prérequis de celui-ci.

1 Puissances, logarithmes et exponentielles

Soit un entier n positif, la n-ème puissance d'un nombre réel positif x, notée x^n, est le produit de n copies de x. On appelle n l'exposant de la puissance. Cette opération est inversible, et son inverse est la racine n-ème d'un nombre x, notée n√x. On montre que x^{a+b} = x^a x^b et x^{a \times b} = (x^a)^b = (x^b)^a, et on convient par extension que x^{-n} = 1/x^n et x^{1/n} = n√x, et que x^{a/b} = b√(x^a) = (b√x)^a. Noter que (x^a)^b n'est généralement pas égal à x^{(a^b)}.

Une fonction logarithme f est définie par la propriété que f(x \times y) = f(x) + f(y). On en déduit que f(1) = 0. Si on impose que f est dérivable, alors on doit avoir f'(x) = k/x. La fonction logarithme naturel (ou népérien), ln, est la solution de ce système quand k = 1. On montre que ln est définie de 0 à +∞, et que son image croît strictement de -∞ à +∞, que ln x^a = a \times ln x et que ln (x/y) = (ln x) - (ln y). Noter que pour x suffisamment grand on a ln x << x, et que pour x voisin de 1 on a ln x ≈ x - 1.

On observe que ln est inversible, et on appelle exponentielle, e(x), son inverse. On montre que e'(x) = e(x). On appelle nombre d'Euler la valeur e(1) = 2,718..., et on la note e. On convient que e^x = e(x). On montre que e^{x+y} = e^x \times e^y et e^{x \times y} = (e^y)^x = (e^x)^y, et que e^{-x} = 1/e^x. On convient que a^x = e^{x \times \ln a}, ce qui généralise l'opération puissance à des exposants quelconques. Noter que (a^y)^x est en général différent de a^{(x^y)}.

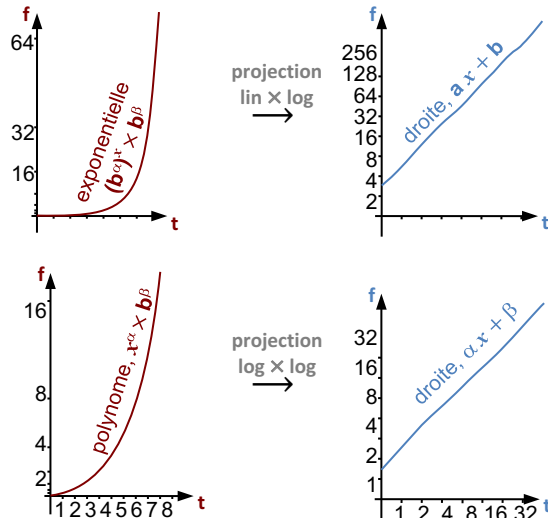
Dans le cas général, on note b = e(1/k), et on appelle la solution du système logarithme en base b, noté log_b. On montre que log_b b^x = b^{log_b x} = x, que (log_b b') \times (log_b b) = 1 et log_b x = (log_{b'} x) / (log_{b'} b). On voit que ln x = log_e x et log_b x = ln x / ln b.

On montre enfin que pour tout alpha > 0, b > 1, on a lim_{x->∞} x^alpha / log_b x = ∞ et lim_{x->∞} b^x / x^alpha = ∞ ; une puissance, aussi faible soit-elle, finit toujours par dépasser un logarithme, et une exponentielle finit toujours par dépasser une puissance. Ce sont 3 ordres de grandeurs très différents.

Le logarithme permet des changements de variable intéressants. Quand la mesure d'un phénomène croît d'un facteur constant à chaque pas de temps, on dit que sa croissance est exponentielle, car f(t) = k^{t-t_0} \times f(t_0). Ces phénomènes sont difficiles à représenter par une courbe car le rapport entre les plus petites valeurs et les plus grandes est énorme : ex. environ 8 milliards sur l'histoire des capacités des disques durs depuis leur invention en 1956, alors qu'une feuille ou un écran ne font que 1000 pixels de haut. Utiliser le logarithme dans la présentation de ce genre de données permet de mieux en rendre compte graphiquement.

Le changement de variable (t, f(t)) -> (t, log_b f(t)), c-à-d. afficher log_b f(x) plutôt que f(x) à l'abscisse x, rend le graphe de f plus lisible et l'hypothèse de croissance exponentielle facile à vérifier si les points s'alignent sur une droite ; on appelle cela la projection lin \times log.

Le changement de variable (t, f(t)) -> (log_b t, log_b f(t)), c-à-d. afficher log_b f(x) à l'abscisse log_b x plutôt que f(x) à l'abscisse x forme la projection log \times log. Son avantage est que le graphe des phénomènes qui ont une croissance polynomiale y tend vers une droite dont la pente est le degré du polynôme. C'est un moyen très simple de vérifier si un phénomène a une croissance polynomiale ou non, et si oui quel est son degré.



3 Éléments d'une théorie des systèmes

Si on veut que système judiciaire, système nerveux, système solaire et système de gestion de fichiers partagent une notion commune de système on ne peut que la définir d'une façon très vague. On dira qu'un système est caractérisé par des entités et par les relations qu'elles entretiennent. Ex. les entités du système solaire sont le Soleil, les planètes et tous les corps qui orbitent autour du Soleil et des planètes, tous entretenant des relations d'attraction gravitationnelle. De même, les entités d'un système de fichiers sont des fichiers et des répertoires qui entretiennent une relation de contenant-contenu : contient \subseteq répertoires \times {fichiers U répertoires}, car un répertoire peut contenir des fichiers et des répertoires.

Un système peut être naturel, ex. le système solaire, ou artificiel, ex. le système judiciaire. Qu'il soit naturel ou non, un système peut être une vue de l'esprit, un modèle, construit pour simplifier l'analyse, ex. le système Terre-Lune, ou le système judiciaire quand il fait abstraction des questions sociales ou de santé mentale. Un système a en général une notion de dedans et de dehors qui permet de déterminer quelles sont les entités du système. Cette frontière peut être conventionnelle, ex. les limites du système solaire, ou être concrète, ex. la membrane plasmique des cellules biologiques. De même, les entités peuvent être concrètes, ex. la Lune, ou bien abstraites, ex. une société anonyme. Même concrètement délimité, un système peut être ouvert ou fermé. Dans le premier cas, il entretient des relations avec son extérieur, dans le second cas, non. En pratique, il n'existe pas de système fermé, même si on essaye souvent d'en construire en laboratoire dans des situations d'expérimentation, ex. en thermodynamique.

Le plus souvent on décrit un système dans une phase stable, c'est-à-dire obéissant à des règles qui semblent pouvoir durer indéfiniment, et qui semblent durer depuis toujours, ex. le système solaire, la vie cellulaire, la nationalité française, le cycle du carbone en climatologie. Ce sont des vues de l'esprit commodes, mais ça ne dit rien de la création de ces systèmes dont on sait bien qu'ils n'ont pas toujours existé ainsi et qu'ils n'existeront pas toujours. On parlera de phase transitoire pour la période où de nouvelles règles se mettent en place.

Les entités d'un système peuvent elles-mêmes être analysées comme des systèmes ; on les appellera des sous-systèmes. Souvent, on attribue à des systèmes des propriétés qu'on n'attribue pas à leurs sous-systèmes. Ex. la vie, propriété qu'on attribue aux cellules et aux systèmes biologiques dont elles sont des sous-systèmes, mais qu'on n'attribue pas aux organites et biomolécules qu'elles contiennent. Il en est de même pour les propriétés macroscopiques des matériaux, ex. couleur, brillance ou dureté, qu'on ne peut pas attribuer aux atomes dont ils sont composés. On appelle ces propriétés des propriétés émergentes. Ce concept est parfois résumé dans l'adage Le tout est plus que la somme des parties, mais aussi parfois contesté en affirmant qu'il n'y a pas émergence mais simplement ignorance des mécanismes profonds de ces systèmes.

4 Un sous-système nommé disque dur magnétique

Un support de stockage de masse persistant permet de mémoriser une grande quantité de données sans autre apport d'énergie que pour lire et écrire. Le disque dur magnétique (hard drive, HD) est un support de stockage persistant qui coexiste dans les systèmes informatiques avec la bande magnétique et le disques SSD.

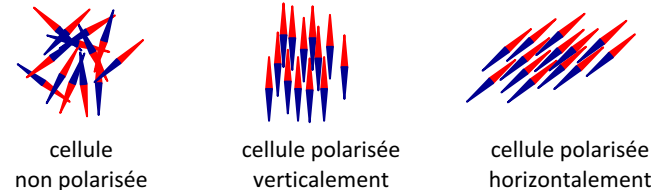
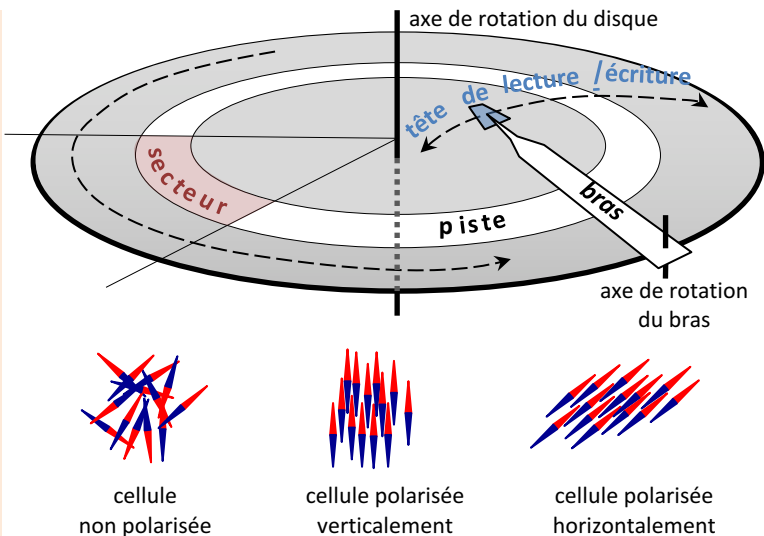
En 1956, IBM commercialisait le premier disque HD. Son principe est celui d'un disque fait d'un matériaux léger et rigide, ex. verre ou aluminium, revêtu d'une couche ferromagnétique et tournant à vitesse constante autour d'un axe normal à sa surface (aujourd'hui, environ 5000-7000 tr/min, soit env. 1 tour en 10 ms). La surface est divisée en anneaux concentriques qu'on appelle des pistes.

Soumis à un champ magnétique localisé, des groupes de particules ferromagnétiques (formant chacun une cellule) s'orientent (se polarisent) de façon cohérente alors que les groupes voisins gardent leur position initiale. Inversement, une cellule polarisée induit une variation localisée de champ magnétique. On convient donc que les deux valeurs possibles d'un bit de mémoire correspondent à deux polarisations opposées d'une cellule. Le principe de la lecture/écriture est de déplacer au-dessus du disque une tête de lecture/écriture capable d'induire ou de détecter une variation de champ magnétique. À cette fin, la tête de lecture est placée au bout d'un bras qui peut osciller dans le plan du disque de tel sorte que la tête puisse survoler à la demande n'importe quelle piste du disque.

La surface du disque est aussi divisée en secteurs géométriques, et l'intersection d'une piste et d'un secteur géométrique est simplement appelée secteur. Pour fixer les idées on peut dire qu'un disque comporte quelques dizaines de secteurs géométriques et quelques dizaines de milliers de pistes. La capacité d'un secteur est le nombre de cellules magnétiques qu'on peut y distinguer : ex. quelques millions de millions. Les nombres de cellules par secteur, nombres de secteurs par pistes et nombres de pistes sont les facteurs qui déterminent la capacité totale du disque. Depuis 1956, ces nombres n'ont fait qu'augmenter, permettant à la fois la diminution de la taille des disques et l'augmentation exponentielle de leur capacité (environ un doublement tous les deux ans).

L'unité d'échange avec un disque dur magnétique est le secteur. Le secteur désiré est identifié par ses numéros de piste et de secteur géométrique. Que se soit pour lire ou écrire un secteur il faut d'abord positionner la tête de lecture au-dessus de sa piste, puis attendre que la rotation du disque amène le secteur désiré sous la tête de lecture/écriture, puis faire l'opération demandée (lire ou écrire). Le positionnement du bras et la rotation du disque (1/2 tour en moyenne, soit environ 5 ms) font qu'un accès disque peut prendre quelques dizaines de millisecondes, alors qu'un accès mémoire ne prend que quelques dizaines de nanosecondes. C'est un million de fois plus lent.

Le disque dur à semi-conducteurs, (solid-state drive, SSD) est une autre solution de stockage de masse persistant qui remplace progressivement le disque dur magnétique. Utilisant une toute autre technologie que le disque dur magnétique, le transistor à grille flottante, il partage cependant avec lui une unité d'échange assez volumineuse (env. 4000 octets), et un temps de traitement meilleur mais encore assez long, environ 0,1 ms, soit 10 mille fois plus lent que la mémoire.



2 Quelques sommes et produits de suites finies

La somme et le produit étant des opérations associatives, commutatives, ayant chacune un élément neutre, on en dérive des opérations de sommes et produits itérés : \sum_{i \in [a; b]} t_i et \prod_{i \in [a; b]} f_i où les t_i et f_i sont des termes et facteurs dépendants de i. Si l'intervalle des indices est vide, la valeur conventionnelle de l'expression est l'élément neutre de l'opération, ex. 0 pour la somme et 1 pour le produit, mais aussi Faux pour la disjonction (V), qui est une sorte de somme, et Vrai pour la conjonction (A), qui est une sorte de produit.

Certaines sommations doivent être connues. Il est même bon de savoir les retrouver :
• \sum_{i \in [1; n]} i : la somme des n 1-ères entiers est égale n \times (n+1) / 2. Ex. \sum_{i \in [1; 17]} i = 17 \times 18 / 2 = 153.
• \sum_{i \in [0; n]} x^i : la somme des n 1-ères puissances de x est égale à (x^{n+1} - 1) / (x - 1). Ex. \sum_{i \in [0; n]} 2^i = 2^{n+1} - 1.

Il est bon aussi de se rappeler quelques transformations élémentaires :
• \sum_{i \in [0; n]} (k \times t_i) = k \times \sum_{i \in [0; n]} t_i et \sum_{i \in [0; n]} k = k \times n, si k ne dépend pas de i.
• \sum_{i \in [0; n]} (s_i + t_i) = \sum_{i \in [0; n]} s_i + \sum_{i \in [0; n]} t_i. Attention, rien de tel pour \sum_{i \in [0; n]} (s_i \times t_i) !

6 Quelques chiffres

Les chiffres qui suivent sont mondiaux (population > 7,7 milliards). Ils sont de 2018-2019. L'unité de temps est l'année, sauf quand une autre unité est spécifiée.

- Plus de 4 milliards d'internautes ; la moitié en Asie (un milliard égale 10^9 ou environ 2^30).
Environ 28 milliards d'objets connectés.
Plus de 1,9 milliard de sites web utilisant plus de 333 millions de noms de domaine.
Plus de 2000 milliards de recherches internet (65 000 par seconde) ; 92 % via Google.
Environ 260 millions de PC vendus (~ 8 par seconde), 160 millions de tablettes et 1400 millions de smartphones.
Environ 150 000 péta-octets de trafic internet (un péta vaut 10^15 ou environ 2^50).
Un système logiciel complexe (ex. Linux, Windows ou Android) compte plusieurs millions de lignes de code.
Le dépôt des logiciels de Google compte plusieurs milliards de lignes de code.
Un processeur intégré compte plusieurs milliards de transistors.

5 Quelques dates

- 1936 : publication par Alan Turing de son modèle de calcul et preuve de l'existence de machines de Turing universelles où un programme est traité comme une donnée.
1937 : démonstration par Claude Shannon de ce que l'algèbre de Boole est un bon modèle du calcul électronique.
1945 : publication par John Von Neumann de son modèle d'architecture qui réalise l'idée de programme en mémoire traité comme une donnée.
1948 : publication par Shannon de sa théorie de l'information. Cette théorie gouverne une grande partie des systèmes de communication, mais aussi des systèmes de calcul comme ceux de deep learning.
1961 : publication par Rolf Landauer de son principe qui relie calcul et énergie. Principe largement ignoré des concepteur de systèmes informatiques.
1972 : publication du principe de localité par Peter Denning. Ce principe gouverne un très grand nombre de dispositifs d'accélération des calculs et des communications. Remarquons du même auteur ce beau principe : Solidarity, not software, generates collaboration.

0 N'oubliez jamais !

Un modèle est toujours imparfait, mais peut être utile, et c'est tout ce qu'on peut lui demander (d'après George Box 1978, voir aussi les cartes à l'échelle 1/1 de Jorge Luis Borges 1946 et celles de Lewis Carroll 1893 pour des exemples de perfection inutilisable).

L'informatique ne travaille que sur la représentation des choses ; il lui faut des capteurs et des actionneurs pour toucher les choses. Par contre, travailler sur des représentations de représentations de ... est courant. Par exemple, 440 est une représentation d'un entier qui est la mesure en hertz d'une fréquence, 0x1B8 est une autre représentation du même entier, et la_3 est une autre représentation de la même fréquence. La fréquence double, l'octave, pourrait être représentée par 880, 0x370 ou la_4. Il faut toujours se demander ce qui a du sens par rapport à ce qui est modélisé et ce qu'on va en faire.