

0.1 Objectifs (Eduscol)

La Terre est habitée par une grande diversité d'êtres vivants, dont l'espèce humaine. Cette biodiversité est dynamique et issue d'une longue histoire : l'évolution des espèces. Les activités humaines se sont développées et ont des effets directs et indirects sur les écosystèmes. Une approche systémique met en lumière les relations entre la santé humaine, la santé animale et les écosystèmes. **Les mathématiques proposent des modèles pour mieux appréhender la dynamique des systèmes vivants. L'être humain a construit des machines pour traiter l'information et a créé des langages pour les commander. Avec les méthodes de l'intelligence artificielle, il continue d'étendre les capacités de traitement de données et les domaines d'application de l'informatique.**

Les thématiques abordées sont éventuellement l'occasion de remobiliser des contenus concernant les phénomènes aléatoires et les phénomènes d'évolution. Une partie de la biodiversité actuelle est encore méconnue. Diverses approches cherchent à l'estimer et à en comprendre l'évolution. Ce thème est l'occasion d'approfondir les processus permettant l'apparition de nouveautés au sein de populations. Ces dernières sont soumises à des pressions de sélection qui vont modifier la fréquence de représentation de ces nouveautés et donc participer à l'évolution des espèces. L'être humain peut être à l'origine de façon directe ou indirecte de telles pressions. L'espèce humaine, comme les autres êtres vivants, est issue d'une histoire évolutive complexe que la paléanthropologie cherche à retracer. Les archives paléontologiques sont rares, mais des techniques de biologie moléculaire les complètent.

La démarche de modélisation mathématique comporte plusieurs étapes : identification du type de modèle le mieux adapté pour traduire la réalité, détermination des paramètres du modèle, confrontation des résultats du modèle à des observations, qui peut conduire à limiter son domaine de validité ou à le modifier.

Les mathématiques et l'informatique contribuent à l'élaboration de modèles démographiques et au développement de l'intelligence artificielle dont les nombreuses potentialités et les limites, notamment éthiques, sont à discuter.

I-Les modèles démographiques Prédire l'évolution de l'effectif d'une population humaine et des ressources qui lui sont nécessaires est un enjeu majeur du développement durable. Pour prédire ces évolutions, les scientifiques utilisent des modèles mathématiques. La démarche de modélisation comporte notamment le choix d'un modèle et le contrôle de sa validité. La présentation de l'exemple historique de Malthus permet de mettre en œuvre cette démarche mathématique dans le cas discret.



Figure 1: Pascaline-1642 (Wikipedia)

II-De la machine de Turing à l'intelligence artificielle L'être humain n'a cessé d'accroître sa capacité d'action sur le monde, utilisant son intelligence pour construire des outils et des machines. Dans le contexte du traitement automatique de l'information (informatique), il a élaboré un

mode de pensée algorithmique susceptible d'être codé dans des langages permettant de commander des machines. Plus largement, le terme *intelligence artificielle* (IA) se définit comme un champ interdisciplinaire théorique et pratique qui a pour objet la compréhension de mécanismes de la cognition et de la réflexion, et leur imitation par un dispositif matériel et logiciel, à des fins d'assistance ou de substitution à des activités humaines : reconnaître et localiser des objets dans une image, conduire une voiture, traduire un texte, dialoguer, etc. Un champ de l'intelligence artificielle ayant récemment permis des applications spectaculaires est celui de l'apprentissage automatique.

0.2 Tout part du calcul

Le mot algorithme dérive du nom du grand mathématicien persan Al Khwarizmi (vers l'an 820). Il introduisit en Occident la numération décimale (rapportée d'Inde) et enseigna les règles élémentaires des calculs s'y rapportant. Avant l'ère moderne, on trouve des machines à chaque époque: la *Pascaline* (1642) est capable d'additionner et de soustraire, la *machine à calculer* de Charles Babbage, mathématicien Anglais du XIX^{eme}.

Augusta Ada King ou Ada Lovelace rencontre Charles Babbage et de cette rencontre naît le projet de la machine analytique. Ada écrit les premiers programmes vers 1840 et peut être considérée comme la première personne ayant codé. Elle développe les notions de boucles, de conditionnelles entravaillant sur la machine analytique de Charles Babbage, avec qui elle collabore. Elle écrit à propos de cette machine, théorisée mais jamais réalisée de son vivant. Il faudra attendre les années 1930 avec Alan Turing pour formaliser une telle notion de calculateur universel qui manipule des symboles généraux, et abandonner la notion de calculatrice purement numérique.

La machine tisse des motifs algébriques comme le métier de Jacquard tisse des fleurs et des feuilles.

Ada Lovelace



0.3 Des débuts aux premiers ordinateurs électroniques

- Alonzo Church (1903-1995) et Alan Turing (1912-1954), élève de John Von Neumann. Turing et son équipe dès 1939 entrent à Bletchley Park où ils finissent par décrypter la machine Enigma (avec la machine de Lorenz).

Turing et son équipe développe notamment les calculateurs Robinson et Colossus sont les premières machines qui intègrent les concepts d'arithmétique binaire, d'horloge interne, de mémoire tampon, de lecteurs de bande, d'opérateurs booléens, de sous-programmes et d'imprimantes. Tout ceci sera classé *Secret défense* jusqu'en 1975.

- Le 28 mai 1936, Alan Turing propose dans un article intitulé "On Computable Numbers", un concept qui permet à une machine d'interpréter un code et donc d'effectuer des calculs de type différents. Ce qu'on appelle la "machine de Turing" n'est alors qu'un modèle théorique précurseur de l'informatique et non une réalisation complète. Cette application visionnaire sera la base du développement informatique et du fonctionnement par algorithme. Le premier

ON COMPUTABLE NUMBERS, WITH AN APPLICATION TO
THE ENTSCHIEDUNGSPROBLEM

By A. M. TURING.

[Received 28 May, 1936.—Read 12 November, 1936.]

The "computable" numbers may be described briefly as the real numbers whose expressions as a decimal are calculable by finite means. Although the subject of this paper is ostensibly the computable numbers, it is almost equally easy to define and investigate computable functions of an integral variable or a real or computable variable, computable predicates, and so forth. The fundamental problems involved are, however, the same in each case, and I have chosen the computable numbers for explicit treatment as involving the least cumbersome technique. I hope shortly to give an account of the relations of the computable numbers, functions, and so forth to one another. This will include a development of the theory of functions of a real variable expressed in terms of computable numbers. According to my definition, a number is computable if its decimal can be written down by a machine.

In §§ 9, 10 I give some arguments with the intention of showing that the computable numbers include all numbers which could naturally be regarded as computable. In particular, I show that certain large classes of numbers are computable. They include, for instance, the real parts of all algebraic numbers, the real parts of the zeros of the Bessel functions, the numbers π , e , etc. The computable numbers do not, however, include all definable numbers, and an example is given of a definable number which is not computable.

Although the class of computable numbers is so great, and in many ways similar to the class of real numbers, it is nevertheless enumerable. In § 8 I examine certain arguments which would seem to prove the contrary. By the correct application of one of these arguments, conclusions are reached which are superficially similar to those of Gödel†. These results

† Gödel, "Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, I", *Monatsh. Math. Phys.*, 38 (1931), 173-198.

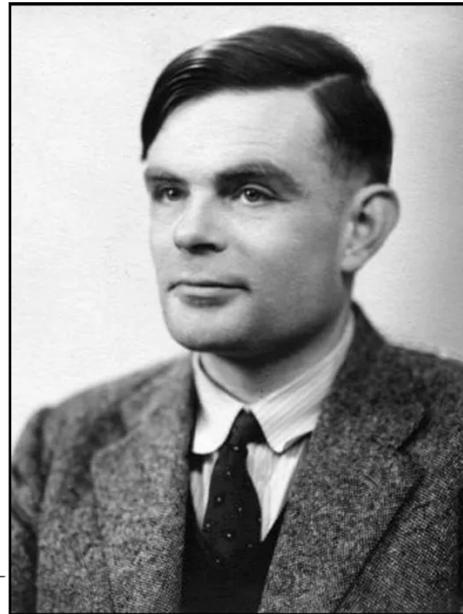


Figure 2: Alan Turing et l'article fondateur de 1936

ordinateur programmable ne sera construit qu'au cours de la Seconde Guerre mondiale et nommé "Colossus". A la même période, naît le Z3 de Konrad Zuse en Allemagne (1941). John Von Neumann (1903-1957) qui a donné son nom à l'**architecture de Von Neumann** utilisée dans la quasi-totalité des ordinateurs modernes. Cela est dû au fait qu'il est, en 1944, le rapporteur des travaux pionniers en la matière, oubliant Eckert, Mauchly. Le modèle

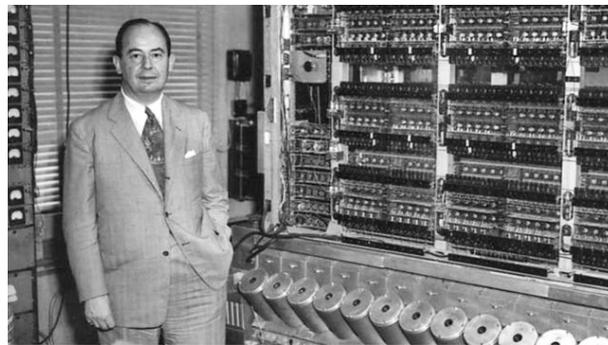


Figure 3: John Von Neumann

de calculateur à programme auquel son nom reste attaché, et qu'il attribuait lui-même à Alan Turing, possède une unique mémoire qui sert à conserver les logiciels et les données. Ce modèle extrêmement innovant pour l'époque, est à la base de la conception de nombre d'ordinateurs.

Alan Turing se suicidera en 1954, en mangeant une pomme imbibée de cyanure. Une légende souvent rapportée dit que cet épisode est à l'origine du logo d'Apple, mais le créateur du logo, Rob Janoff, a démenti dans de nombreuses interviews toute référence au mathématicien.

- 1945-1947: Marlyn Wescoff and Ruth Lichterman were two of the female programmers of ENIAC: premier ordinateur entièrement électronique pouvant être Turing-complet. Utilisé pour les calculs ayant servi à fabriquer la Bombe H. Il se programme avec des commutateurs et des câbles: 167 m², 30 tonnes.

- En 1950 est publié l'article *Computing Machinery and Intelligence* ouvert par le premier paragraphe : *The imitation game* (Faites le test de Voight-Kampff)

I.—COMPUTING MACHINERY AND INTELLIGENCE

BY A. M. TURING

1. *The Imitation Game.*

I PROPOSE to consider the question, 'Can machines think?' This should begin with definitions of the meaning of the terms 'machine' and 'think'. The definitions might be framed so as to reflect so far as possible the normal use of the words, but this attitude is dangerous. If the meaning of the words 'machine' and 'think' are to be found by examining how they are commonly used it is difficult to escape the conclusion that the meaning and the answer to the question, 'Can machines think?' is to be sought in a statistical survey such as a Gallup poll. But this is absurd. Instead of attempting such a definition I shall replace the question by another, which is closely related to it and is expressed

Figure 4: The imitation game

- 1951: Grace Hopper crée le premier compilateur: A-0 System.



Figure 5: Grace Hopper utilisant un UNIVAC

- 1954-1957 : FORTRAN (Formula Translation), premier langage de programmation dit de haut niveau.
- 1956 : GMOS, OS rudimentaire et IBM 704. Début de l'intelligence artificielle.
- 1965 : Multics Multiplexed Operating and Computing System (pour réseau telecom et électrique)
- 1969 : Margaret Hamilton pilote le projet informatique d'Apollo 11. Naissance d'Arpanet. UNICS qui devient UNIX développé par les BELL LABS conçu pour être portable, multitâche, multi utilisateur codé en assembleur.

0.4 L'arrivée du premier microprocesseur et d'Unix

- 1971 : Intel 4004, premier microprocesseur d'une puissance de calcul équivalente à l'ENIAC.
- 1972: UNIX est refondé en C pour ne pas dépendre trop du matériel. Ken Thompson et Dennis Ritchie.
- 1973: Le Micral.
- 1975: L'OS fondateur CP/M Gary Killdall et Digital Research/ à mettre en rapport avec le Intel 4004, premier microprocesseur. Une partie de l'OS qui s'occupe du matériel Basic input output system est stocké dans une puce ROM. Killdall rate l'accord avec IBM, ce qui profite à Bill Gates Le terme générique DOS: Disk operating system. Création de Microsoft par Bill Gates et Paul Allen, qui fournit un BASIC.
- 1976: Apple (Steve Jobs, Steve Wozniak), Ronald Wayne. Le microprocesseur entre dans le marché du jeu vidéo.
- 1977: Commodore achète le basic de Microsoft pour son PET mais ne paye qu'à la sortie de la machine et manque de couler Microsoft sauvé...par un contrat avec Apple.
- Atari lance la VCS 2600 ; la première console de jeu avec un succès mondial.

0.5 Diversité de machines et de système d'exploitation

- 1980-1981 : Microsoft achète à Seattle Computer le QDOS (quick & Dirty DOS).
À ce moment, IBM ne croit en effet guère à ce segment de marché. Elle approche Digital Research pour obtenir une licence de son système **CP/M**. Échec des négociations et entrée en scène de Bill Gates. Profitant de l'occasion, il propose PC-DOS 1.0 qui n'est autre qu'un système racheté à Seattle Computer Products, 86-DOS. L'**IBM PC** sort le 12 Aout 1981...
Ce dernier a un gros avantage: une compatibilité avec les API (Application Programming Interface) de CP/M. Coût de l'opération pour **Microsoft**, 50 000\$, probablement l'un des investissements les plus rentables de l'histoire. Quand sort le premier PC d'IBM en 1981, PC-DOS est devenu MS-DOS. Les années suivantes voient une importante diversité de machines et de système d'exploitation. La micro informatique entre définitivement dans les foyers. Mais la guerre du PC est déclarée. **Commodore**, Texas Instruments et **Atari** se lancent dans une guerre des prix partiellement responsable du crash des jeux vidéo de 1983. IBM est grignoté par les clones que son ouverture a encouragés. Apple traverse des crises politiques avec le Macintosh et se barricade en système fermé. Microsoft, enfin, s'érige en nouvel empire de la tech sur les ruines laissées par **Big Blue**.
- Arrivée de Nintendo, entreprise créée en 1880, sur le marché des consoles de jeu.
- 1983 :Arpanet passe au protocole TCP/IP.
- 1984 : MacOS.
- 1985/1986 : TOS, écrit par Atari et Digital Research. Avec AmigaOS, ce sont deux OS avec interface graphique: GEM et Workbench.
- Arrivée de Sega sur le marché de la console.
- 1987 : MINIX OS "pédagogique" basé sur UNIX est reprogrammé par **Linux Torvalds**. La fusion avec le projet GNU de **Richard Stallman** aboutit à **GNU/Linux** en 1991.
- 1989 : Tim Berners-Lee (né en 1955) invente le World Wide Web (WWW) pour que les chercheurs puissent partager les informations au sein du CERN.
- Le projet GNU de Richard Stallman est complété par le noyau Linux de Linus Torvalds: c'est la naissance de GNU/Linux.



Figure 6: Le système GNU/Linux naît en 1991 de la fusion du projet GNU et du noyau Linux

- 1993: Mosaic premier navigateur commercial puis Netscape.
- Arrivée de Sony sur le marché de la console.
- 1998: Création de Google.
- 2001: Arrivée de Microsoft sur le marché de la console.
- 2007: Création d'Android utilisant le noyau Linux et iOS.

Date*	Composants logiques	Mémoire interne	Mémoire externe	Programmation
1950	Tubes	Expérimentations	Carte ou bandes perforées puis bandes magnétiques	Code binaire, assembleurs
1959	Transistor	Tores de ferrite	Cartes, tambours	Premiers langages évolués
1966	Circuits intégrés	Torre de ferrite et disque rapide	Cartes, bandes, disques	Langages évolués
1973	Micro-processeurs	Semi-conducteurs	Bandes disques, disquettes, cartouches	Langages évolués et communication graphique

0.5.1 Comment définir un ordinateur

Un ordinateur est un ensemble de dispositifs mécaniques, électroniques et logiciels capable de réceptionner, de traiter et d'émettre de l'information. Nous avons vu que les travaux d'Alan Turing puis de Von Neumann et Grace Hopper ont permis de créer les premiers ordinateurs entre 1940 et 1950. Encore aujourd'hui, les ordinateurs suivent l'architecture de Von Neumann. Ce modèle décompose l'ordinateur en 4 parties distinctes :

1. l'unité arithmétique et logique (UAL) ou unité de traitement, qui effectue les opérations de base ;
2. l'unité de contrôle, qui est chargée du séquençage des opérations ;
3. la mémoire, qui contient à la fois les données et le programme qui indique à l'unité de contrôle quels calculs faire sur ces données. La mémoire se divise en mémoire vive (programmes et données en cours de fonctionnement) et mémoire de masse (programmes et données de base de la machine) ;

4. les dispositifs d'entrée-sortie, qui permettent de communiquer avec le monde extérieur.

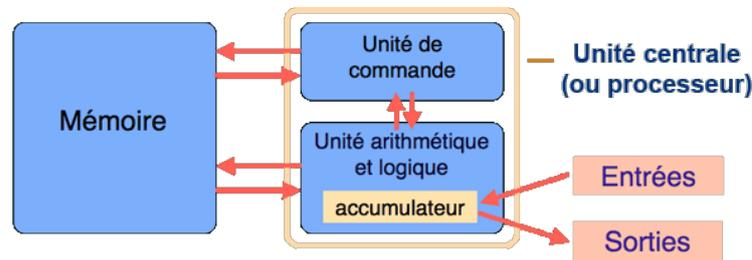


Figure 7: Architecture de Von Neumann (du site interstices.info)

Un ordinateur ne manipule que des données numériques et leur acquisition peut passer par une numérisation : scanner, micro, caméra etc. Les données sont alors converties en valeurs binaires *compréhensibles* par la machine : une suite de 0 et de 1 nommés bits pour *binary digit*. C'est la plus petite unité d'information manipulable sur une machine numérique. Il est possible de représenter physiquement cette information binaire par un signal électrique ou magnétique, qui, au-delà d'un certain seuil, correspond à la valeur 1. Une unité fondamentale en informatique est l'octet : regroupement de 8 bits. Par exemple 11110010. Il permet par exemple de stocker un caractère comme une lettre ou un chiffre. Combien de valeurs différentes peut-on coder sur un octet ?

0.5.2 Les chiffres

On peut bien sûr représenter les nombres en base 2, intéressons-nous aux nombres entiers, c'est identique à la façon dont nous comptons en base 10.

Notation positionnelle: représentation des entiers positifs En décimal, le nombre 123 est égal à $1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0$; la position des chiffres indique si on doit les considérer comme des unités, des dizaines, des centaines. On dispose des dix chiffres de 0 à 9.

Utilisons une représentation dans un tableau:

10^3	10^2	10^1	10^0
0	1	2	3

On a représenté ce nombre avec 4 bits.

En binaire, le mode de calcul est le même mais la base est 2, on ne dispose que de deux chiffres: 0 et 1. Le nombre 1101_2 est égal à $1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 8 + 4 + 1 = 13$.

C'est la représentation utilisée pour les entiers machines des processeurs modernes.

Exercices Calculer la valeur décimale des nombres binaires suivants: 11000101_2 , 11111111_2 .

Exercice: Nombres en base 10 et en base 2 Dans cet exercice, lorsque la base n'est pas indiquée, c'est la base 10.

Base 10 Nous représentons quotidiennement les nombres avec la base 10: les 9 chiffres permettent par leur position dans un nombre de représenter tous les nombres entiers.

Exemple: En base 10 ou décimale, l'entier naturel 123 se décompose de la façon suivante:

$$1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0$$

La position des chiffres indique si on doit les considérer comme des unités, des dizaines, des centaines. On dispose des dix chiffres de 0 à 9 pour multiplier chacune des puissances de 10.

Utilisons une représentation dans un tableau:

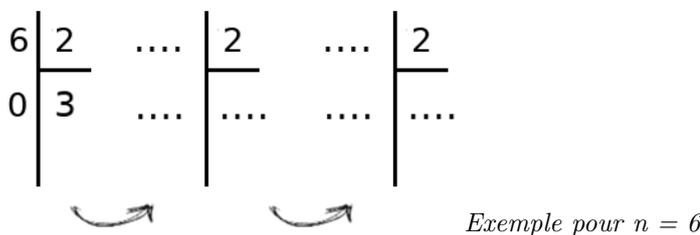
Puissance de 10 ou Poids	10^3	10^2	10^1	10^0
Coefficient	0	1	2	3

1. Donner de la même façon la décomposition du nombre 1912.

Base 2 Les ordinateurs calculent en base 2 ou binaire. Cela vient de leur fonctionnement: les constituants de base que sont leurs transistors savent ou non laisser passer le courant. Il n'y a donc que deux états. Le mode de calcul est le même mais la base est 2, on ne dispose que de deux chiffres: 0 et 1.

C'est la représentation utilisée pour les entiers machines des processeurs modernes.

1. **Conversion binaire vers décimal.** Donner la décomposition avec les puissances de 2 des nombres binaires 1101_2 , 1111_2 . En déduire leur valeur décimale.
2. **Conversion décimal vers binaire.** Pour convertir un nombre entier n écrit en base 10 vers la base 2, on effectue des divisions euclidiennes successives par 2:



En base 2, le nombre n s'écrit $r_p \dots r_2 r_1 r_0$ avec $0 \leq i \leq p$, $p + 1$ nombre de chiffres nécessaires à l'écriture de n en base 2.

Autrement dit, il suffit d'écrire les restes obtenus **de droite à gauche** dans l'ordre de leur apparition pour obtenir la conversion.

1. Compléter l'exemple pour convertir 6 en base 2 et sur 4 bits (**binary digit**).
2. Quel sont les restes possibles dans une division euclidienne lorsqu'on divise par 2 ?
3. Convertir les nombres 15 puis 192 en base 2 sur 4 bits.

Base 16 Le principe est le même, on utilise les 10 chiffres puis A=10, B=11, C= 12, D=13, E=14 et F=15 pour obtenir une base de 16 "chiffres".

1. Donnez la valeur de $5D_{16}$ en décimal.
2. Avec la méthode des divisions successives, convertissez 18 puis 75 en base 16.

0.5.3 Stockage, type et taille des fichiers

Voici les unités standardisées :

- | | |
|--|---|
| 1. Un kilooctet (Ko) = 10^3 octets | 6. Un exaooctet (Eo) = 10^{18} octets |
| 2. Un mégaooctet (Mo) = 10^6 octets | 7. Un zettaooctet (Zo) = 10^{21} octets |
| 3. Un gigaooctet (Go) = 10^9 octets | 8. Un yottaooctet (Yo) = 10^{24} octets |
| 4. Un téraooctet (To) = 10^{12} octets | 9. Un ronnaooctet (Ro) = 10^{27} octets |
| 5. Un pétaooctet (Po) = 10^{15} octets | 10. Un quettaooctet (Qo) = 10^{30} octets |

Et une photographie de l'évolution des supports :

Donner des types de fichiers et leur ordre de grandeur.

0.6 Encodage ASCII

Les caractères, comme tout le reste, doivent être transformés en séries de bits. Il y a longtemps, le codage d'un caractère se faisait uniquement sur 7 bits et seulement en caractères latins de base: pas de caractères accentués donc. On s'inquiétait surtout de pouvoir coder des textes en anglais, essentiellement techniques. Le code le plus courant sur les ordinateurs, et qui finit par s'imposer, fut le code **ASCII**. Le problème fut que de nombreux autres, concurrents, subsistaient. Un même codage ou série de bits pouvait correspondre à divers caractères selon l'encodage. Par exemple, la norme **ISO-8859-1 (latin-1)** codifie tous les caractères accentués du français ou de l'allemand. Ce qui posait évidemment problème si on ouvrait un fichier avec le mauvais encodage.

Dorénavant, le codage des caractères est autorisé sur plusieurs octets avec **Unicode**: celui-ci correspond à plusieurs codages dont le plus usuel est **UTF-8**. C'est aujourd'hui le code à utiliser ; les autres doivent être considérés comme obsolètes. Le code **UTF-8** reprend le code **ASCII** pour les caractères de base ; les autres caractères étant codés sur plusieurs octets.

ASCII TABLE

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

Figure 8: Encodage ASCII

0.7 Correction de bugs

0.8 Annexes

Evolution démographique

Savoirs	Savoir-faire
<p>L'évolution d'une population dont la variation absolue par unité de temps est presque constante est représentée par un nuage de points évoquant une droite. Cette évolution peut être modélisée par une suite arithmétique (modèle dit linéaire).</p> <p>L'évolution d'une population dont la variation relative par unité de temps (encore appelée taux d'évolution) est presque constante est représentée par un nuage de points évoquant la courbe d'une exponentielle. Cette évolution peut être modélisée par une suite géométrique (modèle dit exponentiel).</p> <p>Le modèle démographique de Malthus est un modèle exponentiel d'évolution de l'effectif de la population. Il prévoit que l'effectif de la population décroît vers 0 si le taux de mortalité est supérieur au taux de natalité et croît vers l'infini si le taux de natalité est supérieur au taux de mortalité.</p> <p>Si les prédictions du modèle de Malthus peuvent se révéler correctes sur un temps court, elles sont irréalistes sur un temps long, notamment en raison de l'insuffisance des ressources disponibles.</p> <p>Des modèles plus élaborés prévoient que la population mondiale atteindra environ 10 milliards d'humains en 2050.</p>	<p>À partir de données démographiques, calculer des variations absolues par unité de temps et des variations relatives par unité de temps d'une population afin de choisir entre un modèle linéaire et un modèle exponentiel.</p> <p>Selon le modèle de Malthus, prédire l'effectif d'une population au bout de n années à partir de son effectif initial, de son taux de natalité et de son taux de mortalité.</p> <p>À l'aide d'un tableur, d'une calculatrice ou d'une représentation graphique, calculer le temps de doublement d'une population sous l'hypothèse de croissance exponentielle.</p> <p>À partir de documents fournis, proposer un modèle de croissance de ressources alimentaires (par exemple, la production mondiale de blé ou de riz) et la comparer à une croissance exponentielle.</p> <p>À l'aide d'une calculatrice ou d'un tableur, ajuster un nuage de points par une courbe de tendance et utiliser ce modèle pour effectuer des prévisions.</p> <p>Comparer les valeurs fournies par un modèle à des données réelles afin de tester sa validité.</p> <p>↔ Calculs usuels sur les suites arithmétiques et géométriques, représentations graphiques.</p> <p>↔ Lectures graphiques.</p> <p>↔ Résolution d'équations et d'inéquations.</p>

Information

Savoirs	Savoir-faire
<p>Jusqu'au début du xx^e siècle, les machines traitant l'information étaient limitées à une ou à quelques tâches prédéterminées (tisser grâce à un ruban ou des cartes perforées, trier un jeu de cartes perforées, séparer des cartes selon un critère, sommer des valeurs indiquées sur ces cartes, etc.). Turing a été le premier à proposer le concept de machine universelle qui a été matérialisé dix ans plus tard avec les premiers ordinateurs. Ceux-ci sont constitués au minimum d'un processeur et d'une mémoire vive.</p> <p>Un ordinateur peut manipuler des données de natures diverses une fois qu'elles ont été numérisées : textes, images, sons. Les programmes sont également des données : ils peuvent être stockés, transportés et traités par des ordinateurs. En particulier, un programme écrit dans un langage de programmation de haut niveau (Python, Scratch, etc.) peut être traduit en instructions spécifiques à chaque type de processeur.</p> <p>Un programme peut comporter jusqu'à plusieurs centaines de millions de lignes de code, ce qui rend très probable la présence d'erreurs appelées bogues (ou bugs). Ces erreurs peuvent conduire un programme à avoir un comportement inattendu et entraîner des conséquences graves.</p>	<p>Analyser des documents historiques relatifs au traitement de l'information et à son automatisation.</p> <p>Recenser des outils numériques utilisés dans la vie courante, identifier ceux qui sont programmables, et par qui (ordinateur, thermostat d'ambiance, téléphone intelligent, boîte Internet, ordinateur de bord d'une voiture, etc.).</p> <p>Savoir distinguer les fichiers exécutables des autres fichiers sous un système d'exploitation donné.</p> <p>Connaître l'ordre de grandeur de la taille d'un fichier image, son, vidéo.</p> <p>Savoir calculer la taille en octets d'une page de texte (en ASCII et non compressé).</p> <p>Corriger un algorithme ou un programme bogué simple.</p> <p>↔ Ordres de grandeur.</p> <p>↔ Proportionnalité.</p> <p>↔ Logique.</p>

Intelligence artificielle

<p>L'intelligence artificielle (IA) est née en 1956. A cette époque, elle visait à simuler sur ordinateur les facultés cognitives humaines et recouvrait des approches relevant de l'informatique, des mathématiques et des sciences cognitives. L'approche symbolique (systèmes experts) initiée à la fin des années 50 n'a pas tenu ses promesses. Aujourd'hui, on a tendance à attribuer le terme d'IA à l'un de ses sous-domaines, celui de l'apprentissage automatique (apprentissage machine). Il s'agit d'un processus par lequel un algorithme évalue et améliore ses propres performances, non pas sous l'intervention d'un humain chargé de programmer la machine, mais en répétant son exécution sur des jeux de données de natures variées (mesures de capteurs pour des prévisions, textes pour la traduction, sons pour la reconnaissance vocale, images pour la reconnaissance visuelle, etc.).</p> <p>L'apprentissage machine exploite des méthodes mathématiques qui, à partir du repérage de tendances (par exemple, des corrélations ou des similarités) sur de très grandes quantités de données (données massives), permettent de faire des prédictions ou de prendre des décisions sur d'autres données.</p> <p>La qualité et la représentativité des données fournies sont essentielles pour la qualité des résultats. En effet, l'un des risques de l'apprentissage automatique réside dans l'amplification des biais des données. Par ailleurs, une interprétation trop rapide des données et un amalgame entre corrélation et causalité peuvent aboutir à des résultats erronés.</p>	<p>Analyser des documents relatifs à une application de l'intelligence artificielle.</p> <p>Utiliser une courbe de tendance (encore appelée courbe de régression) pour estimer une valeur inconnue à partir de données d'entraînement.</p> <p>Analyser un exemple d'utilisation de l'intelligence artificielle : identifier la source des données utilisées et les corrélations exploitées.</p> <p>Sur des exemples réels, reconnaître les possibles biais dans les données, les limites de la représentativité.</p> <p>Sur des exemples simples, montrer qu'une corrélation ne correspond pas toujours à une relation de causalité.</p> <p>Expliquer pourquoi certains usages de l'IA peuvent poser des problèmes éthiques.</p> <p>↔ Calcul algébrique.</p> <p>↔ Lectures graphiques.</p>
<p>L'inférence bayésienne est une méthode de calcul de probabilités de causes à partir des probabilités de leurs effets. Elle est utilisée en apprentissage automatique pour modéliser des relations au sein de systèmes complexes, notamment en vue de prononcer un diagnostic.</p>	<p>Dans le contexte d'un diagnostic médical fondé sur un test, produire un tableau de contingence afin de calculer des fréquences de faux positifs, faux négatifs, vrais positifs, vrais négatifs et en déduire le nombre de personnes malades suivant leur résultat au test.</p> <p>Utiliser cette démarche dans d'autres contextes de classification (reconnaissance d'images, détection de messages non sollicités envoyés en masse, etc.).</p> <p>↔ Probabilités conditionnelles.</p> <p>↔ Formule de Bayes.</p>