WIKIPÉDIA

Chaîne de caractères

En informatique, une **chaîne de caractères** est à la fois conceptuellement une suite ordonnée de caractères et physiquement une suite ordonnée d'unité de code (code unit). La chaîne de caractères est un type de donnée dans de nombreux langages informatiques. La traduction en anglais est *string*.

Sommaire

Unités d'une chaîne de caractères

À l'époque des pionniers, on a communément confondu chaîne de caractère et chaîne d'octets, ce qui prête aujourd'hui à confusion, lorsque l'on ne veut pas se limiter à 255 caractères. Par extension, on parle de *chaîne binaire* pour décrire une séquence d'octets.

Certains langages préfèrent gérer les chaînes de caractères à partir d'unités de 16 bits.

En Unicode, le type de donnée « Unicode string » est une séquence ordonnée de « code units »[1].

Dans les langages de programmation

La plupart des langages de programmation offrent une classe ou un type destiné à la représentation et à la manipulation des chaînes de caractères.

Langage	Type de donnée	Description
Python	`str, unicode`	A été modifié avec Python 3.0.
Java	`java.lang.String`	Depuis leur origine, les chaînes Java sont des chaînes Unicode.
C	`char*` et `char[]`	Le langage C n'a jamais connu de véritable type chaîne. Les chaînes de caractères sont couramment simulées par un pointeur sur une séquence de caractères mono-octets se terminant par un octet nul. Des bibliothèques existent pour gérer les chaînes, notamment pour pallier les limites des chaînes mono-octets.
C++	`char*` et `char[]`, `basic_string<>` (`string` ou `wstring`)	Les templates du C++ définissent la classe `std::string` (chaînes à base de caractères mono-octets). Néanmoins, il est souvent nécessaire de manipuler les « chaînes de style C » pour pouvoir utiliser la bibliothèque standard du C++.
C++2011	`char*` et `char[]`, `char16_t`, `char32_t`, `basic_string<>` (`string` ou `wstring`)	Les évolutions du C++2011 permettent de considérer le traitement des chaînes de caractères Unicode.
C#	`string`, `String` et `StringBuilder`	string est un alias vers la Classe String (System.String), donc le type string ne correspond pas à une référence (pointeur). Il est aussi possible d'utiliser char* et char[], mais cela implique l'utilisation du mot clé "unsafe". StringBuilder est recommandé pour les cas où le code effectue de nombreuses concaténations.
Pascal	`String`	Le type de donnée chaîne existe depuis longtemps en Pascal. Toutefois, depuis l'apparition de Delphi, plusieurs types de chaines de caractères ont été ajoutés : AnsiString, UnicodeString et WideString.
Objective C	`char *` et `NSString`	NSString permet de représenter une chaîne de caractère unicode immutable. char * permet de gérer un buffer d'octet.
Javascript	`var`	Les spécifications de l'ECMAScript versions 3 et 5 au moins déclarent toutes deux explicitement une String comme une collection d'entier 16-bit non signés qui quand ils représentent du texte sont des unités de code UTF-16[2].

Représentation numérique

Différentes techniques existent pour représenter des chaînes à l'aide d'octets. Elles nécessitent généralement de pouvoir représenter chaque caractère (encodage), mais aussi de marquer la fin de la chaîne.

La fin de la chaîne peut être connue à l'aide d'un caractère de fin de chaîne (en général 0, mais $ a également été utilisé sous MS-DOS[3]), ou en stockant simultanément le nombre de caractères ou le nombre d'octets de la chaîne.

Chaque caractère est représenté par un nombre d'octets qui dépend du codage de caractères. En fonction de l'encodage utilisé, des limites pourront exister sur l'ensemble des caractères disponibles, les algorithmes de parcours de chaînes, l'interopérabilité et/ou des performances. En particulier, les codages à base de caractères mono-octets tels que les ASCII étendus, peuvent être plus performants, mais limitant et/ou contraignants dans un contexte d'internationalisation et/ou d'interopérabilité. Les autres encodages, comme UTF-8, présentent d'autres caractéristiques.

Représentation en mémoire

Dans une mémoire informatique, l'adresse mémoire du premier caractère est connu. Pour délimiter la fin de la chaîne, soit elle est terminée par un caractère de fin de chaîne (zéro binaire en langage C, et on parle alors d'*ASCIIZ* pour indiquer « *terminé par un zéro* »), soit le nombre de caractères est stocké en parallèle (BASIC, Pascal, PL/I). Dans certains langages orientés objet, le codage interne de la chaîne n'a pas besoin d'être connu (encapsulation).

FRANK en mémoire, délimité par un caractère nul

FRANK en mémoire stocké avec la longueur

F	R	A	N	K	NUL	k	e	f	w
46	52	41	4E	4B	00	6B	65	66	77

length	F	R	A	N	K	k	e	f	w
05	46	52	41	4E	4B	6B	65	66	77

Des séquence d'échappement peuvent également être présentes.

Exemples avec diactriques

Illustration: 「Amélie」en Unicode UTF16BE formes NFC et NFD

Caractère représenté	A	m	é		l	i	e
UTF16-BE NFC	0041	006d	00e9		006c	0069	0065
UTF16-BE NFD	0041	006d	0065	0301	006c	0069	0065
UTF16-BE NFD	A	m	e	'	l	i	e

Exemples internationaux

À titre d'exemple, la table *'[3] ci-dessous décrit le codage de la chaîne 「日本語版Wikipedia」 (Wikipedia version japonaise) dans avec la convention ISO-2022-JP et Unicode.*

Exemple en ISO 2022

Le tableau d'illustration d'exemple est formaté comme suit:

- La première ligne indique chaque caractère.
- La ligne intermédiaire indique le numéro associé à chaque caractère ou le changement de codage.
- La dernière ligne indique chaque octet, sous forme ASCII en bas, et hexadécimal codé décimal en partie supérieure.

Illustration: 「日本語版Wikipedia」en ISO-2022-JP

Caractère représenté		日	本	語	版		W	i	k	i	p	e	d	i	a
機能 区点 行列	JIS X 0208 を指示	38-92	43-60	24-76	40-39	ASCII を指示	05/07	06/09	06/11	06/09	07/00	06/05	06/04	06/09	06/01
octet	01/11 02/04 04/02	04/06 07/12	04/11 05/12	03/08 06/12	04/08 04/07	01/11 02/08 04/02	05/07	06/09	06/11	06/09	07/00	06/05	06/04	06/09	06/01
	ESC $ B	*F l*	*K *	*8 l*	*H G*	*ESC (B*	*W*	*i*	*k*	*i*	*p*	*e*	*d*	*i*	*a*

Note: il se peut que la première séquence d'échappement ne soit pas nécessaire lorsque le texte commence par l'un des 96 caractères du standard américain (ascii).

Exemple en Unicode

Le codage de la chaîne 「日本語版Wikipedia」 peut être fait avec des unités de 16 bits.

- U+65E5 CJK UNIFIED IDEOGRAPH-65E5
- U+672C CJK UNIFIED IDEOGRAPH-672C
- U+8A9E CJK UNIFIED IDEOGRAPH-8A9E
- U+7248 CJK UNIFIED IDEOGRAPH-7248
- U+0057 LATIN CAPITAL LETTER W
- U+0069 LATIN SMALL LETTER I
- U+006B LATIN SMALL LETTER K
- U+0069 LATIN SMALL LETTER I
- U+0070 LATIN SMALL LETTER P
- U+0065 LATIN SMALL LETTER E

Illustration: 「日本語版Wikipedia」en Unicode UTF16BE

Caractère représenté	日	本	語	版	W	i	k	i	p	e	d	i	a

UTF16-BE	65e5	672c	8a9e	7248	0057	0069	006b	0069	0070	0065	0064	0069	0061

Sucre syntaxique

La représentation d'une chaîne de caractères dans un langage de programmation varie d'un système à un autre.

Pour représenter une chaîne de caractères dans un flux de caractères (comme un fichier texte, en particulier dans un code source), il est généralement nécessaire de marquer le début et la fin de la chaîne, et éventuellement d'utiliser des séquences d'échappement.

Généralement, pour représenter une chaîne de caractères, on l'entoure par une paire de caractères spéciaux, souvent des guillemets doubles. On notera par exemple **"Wikipédia"** pour désigner la chaîne composée des neuf caractères **W**, **i**, **k**, **i**, **p**, **é**, **d**, **i** et **a**.

Exemples :

- `"Wikipedia"`
- `'Cette phrase est une chaîne de caractères en langage Pascal qui utilise les apostrophes.'`
- `(Le langage manipule aussi des chaînes de caractères avec des parenthèses.)`
- `""` : chaîne vide, de longueur zéro
- `' '` : chaîne contenant un seul caractère : une espace

Pour pouvoir utiliser ces caractères spéciaux, il existe des conventions. Avec le langage Pascal, on double le guillemet simple pour pouvoir l'introduire dans la chaîne de caractères :

- `'Il s''agit d''un simple guillemet dans la chaîne de caractères.'`

D'autres conventions utilisent un caractère d'échappement ; la barre oblique inversée est le caractère le plus utilisé. Pour les langages Java, C, C++ (et d'autres), on note \" pour introduire un guillemet double dans une chaîne de caractères :

- `"Première solution pour contenir le délimiteur \", un caractère d'échappement"`
- `"Seconde solution pour contenir le délimiteur \\, le doublage du délimiteur"`

Algorithmes

De nombreux algorithmes font partie de l'état de l'art pour traiter les chaînes, chacun pouvant connaître différentes formes. Quelques exemples de catégories de tels algorithmes :

- recherche de sous-chaîne(s) comme celui de Boyer-Moore ;
- recherche d'expressions rationnelles ;
- tri : tri en Unicode, classement alphabétique, classement alphabétique complexe ;
- analyse syntaxique d'une chaîne ;
- conversion (en Unicode, capitalisation, transcodages…).

La prise en compte des chaînes de caractère de manière appropriée par le développeur nécessité généralement de connaître les différents usages des caractères. Les opérations qui semblent évidentes sur un alphabet de vingt-six caractères ne le sont pas nécessairement avec l'ensemble des caractères reconnus par Unicode. Pour une application visant à être diffusé à un niveau mondial, ceci est rendu difficile par les spécificités éventuelles de l'écriture liée à certaines cultures: existence d'équivalence de caractère, taille des caractères asiatiques, sens d'écriture, variante d'écriture pour une même lettre en fonction de sa position, notamment. En particulier une opération aussi simple que compter des caractères peut nécessiter d'être précisée afin de savoir si elle doit servir à compter les unités de code, les points de code ou les graphèmes.

Toutefois des bibliothèques, pour certains langages de programmation, permettent de répondre en partie à ces besoins.

Limitation

Certaines limitations de certains langages de programmations conduisent des développeurs à écrire des bugs, ou certains testeur à ne valider le bon fonctionnement que sur une plage limitée de caractères. En particulier les développeurs anglophones ont la mauvaise habitude de n'effectuer qu'un nombre de tests limités aux seuls caractères ASCII[4].

Voir aussi

- Codage de caractères
- Expression rationnelle

Sur les autres projets Wikimedia :

chaîne de caractères, sur le Wiktionnaire

Notes et références

1. Le Standard Unicode, version 6.1, §2.7 Unicode Strings https://www.unicode.org/versions/Unicode6.1.0/
2. See section 8.4 of the ECMAScript Language Specification.
3. http://www.ctyme.com/intr/rb-2562.htm
4. http://unspecified.wordpress.com/2012/04/19/the-importance-of-language-level-abstract-unicode-strings/