Encodage des caractères

Christophe Viroulaud

Première - NSI

DonRep 14

Encodage des caractères

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

En première approche il semble simple de représenter une chaîne de caractère en mémoire : il suffit d'associer un numéro (un code binaire) à chaque lettre.

01100001	01100010	01100011	01100100	01100101
97	98	99	100	101
a	b	c	d	e

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

En pratique plusieurs contraintes apparaissent. Il faut par exemple que chaque système respecte le même encodage. De plus tous les caractères doivent être représentés.

97	98	99	100	101
a	b	c	d	e

Tableau 1 – machine 1

97	98	99	100	101
!	@	:	#	é

Tableau 2 – machine 2

Héloïse $\Rightarrow$ H@elo&ése

Comment encoder les caractères en mémoire ?

Encodage des caractères

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

Sommaire

1. Première tentative de normalisation : ASCII

2. Prise en compte des différents langages : ISO 8859

3. Encodage universel

Première tentative de normalisation : ASCII

- Années 50, lors de l'apparition des premiers ordinateurs, chaque matériel utilisait son propre système de codage.

Première tentative de normalisation : ASCII

- Années 50, lors de l'apparition des premiers ordinateurs, chaque matériel utilisait son propre système de codage.
- Début des années 60, l'ANSI (American National Standards Institute) propose une première tentative de normalisation : **l'ASCII**.

À retenir

- ASCII : American Standard Code for Information Interchange (Code américain normalisé pour l'échange d'information)
- Un caractère est encodé sur 7 bits. En pratique 1 octet est utilisé ; le bit de poids fort (bit de gauche) sert de somme de contrôle.

Encodage des caractères
Première tentative de normalisation : ASCII
Prise en compte des différents langages : ISO 8859
Encodage universel
Nouvelle norme
Représentation en mémoire

Encodage des caractères

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

Activité 1 :

1. Combien de caractères peut-on représenter en ASCII ?
2. La lettre **A** est représenté par le code binaire : 01000001. Calculer la valeur décimale représentant la lettre A.
3. Calculer la valeur hexadécimale représentant la lettre A.

Correction

$$2^7 = 128$$

Encodage des caractères

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

$$01000001$$

$$1 \times 2^0 + 1 \times 2^6 = 65$$

Encodage des caractères

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

caractère	a	
binaire	0100	0001
hexadécimal	4	1

Decimal	Hex	Char
64	40	@
65	41	A
66	42	B
67	43	C
68	44	D
69	45	E
70	46	F
71	47	G
72	48	H

FIGURE 1 – Extrait de la table ASCII

Encodage des caractères

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

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]

Encodage des caractères

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

Remarque

La table ASCII ne permet pas de représenter les caractères accentués, les idéogrammes...

Sommaire

1. Première tentative de normalisation : ASCII
2. Prise en compte des différents langages : ISO 8859
3. Encodage universel

Prise en compte des différents langages : ISO 8859

La norme ASCII est suffisante pour écrire l'anglais.
Cependant de nombreuses langues utilisent des caractères additionnels non présents dans cette norme.

À retenir

- L'International Organization for Standardization (ISO) a proposé une extension de l'encodage ASCII : **ISO 8859**.
- Encodage sur 8 bits
- Assure une compatibilité avec l'ASCII : les 128 premiers caractères de la norme ISO 8859 sont ceux de la norme ASCII.

Activité 2 :

1. Combien de caractères peut-on encoder dans une table ISO 8859 ?
2. Combien de tables ISO 8859 existe-t-il ?
3. Quelle est la table utilisée pour le français ?
4. Encoder le mot français suivant en utilisant la norme ISO 8859 (en héxadécimal) :

Héloïse

Correction

- $2^8 = 256$ caractères
- Il existe 16 tables.
- La table ISO 8859-1 (Latin-1) est utilisé pour le français. On peut également se servir de sa révision ISO 8859-15 qui ajoute notamment le signe €.

Encodage des caractères

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

H	é	l	o	ï	s	e
48	E9	6C	6F	EF	73	65

Tableau 3 – Encodage avec ISO 8859-1

Encodage des caractères

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

Remarque

Si la table utilisée n'est pas la bonne, le décodage renvoie un texte illisible.

ISO 8859-1	H	é	l	o	ï	s	e
Encodage	48	E9	6C	6F	EF	73	65
ISO 8859-5	H	Щ	l	o	Я	s	e

Sommaire

1. Première tentative de normalisation : ASCII

2. Prise en compte des différents langages : ISO 8859

3. Encodage universel
3.1 Nouvelle norme
3.2 Représentation en mémoire

Encodage universel - nouvelle norme

- Norme ISO-10646
- Chaque caractère, signe ou idéogramme est associé à un numéro unique appelé **point de code**.
- Capacité maximale : 32 bits.

- $2^{32} = 4294967296$ caractères possibles
- exemple : lettre **é** point de code **U+00E9**

Encodage des caractères

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

Sommaire

1. Première tentative de normalisation : ASCII

2. Prise en compte des différents langages : ISO 8859

3. Encodage universel
3.1 Nouvelle norme
3.2 Représentation en mémoire

Représentation en mémoire

Encodage des caractères

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

À retenir

Quatre octets (32 bits) pour chaque caractère est très coûteux en mémoire. La norme **Unicode** (et particulièrement **UTF-8 pour Universal Transformation Format**) définit plusieurs techniques pour économiser de l'espace.

- Si le *bit de poids fort* (le plus à gauche) est 0, il s'agit d'un caractère ASCII codé sur les 7 bits suivants.
- Sinon les premiers bits de poids fort de l'octet indiquent le nombre d'octets utilisés à l'aide de 1 et se terminant par 0.

Suite d'octets (en binaire)	Bits codant
0xxxxxxx	7 bits
110xxxxx 10xxxxxx	11 bits
1110xxxx 10xxxxxx 10xxxxxx	16 bits
11110xxx 10xxxxxx 10xxxxxx 10xxxxxx	21 bits

Tableau 4 – Encodage UTF-8

Suite d'octets (en binaire)	Bits codant
0xxxxxxx	7 bits
110xxxxx 10xxxxxx	11 bits
1110xxxx 10xxxxxx 10xxxxxx	16 bits
11110xxx 10xxxxxx 10xxxxxx 10xxxxxx	21 bits

Tableau 5 – Encodage UTF-8

Remarque

Tous les caractères ASCII sont encodés sur 1 octet.

Suite d'octets (en binaire)	Bits codant
0xxxxxxx	7 bits
110xxxxx 10xxxxxx	11 bits
1110xxxx 10xxxxxx 10xxxxxx	16 bits
11110xxx 10xxxxxx 10xxxxxx 10xxxxxx	21 bits

Tableau 6 – Encodage UTF-8

- lettre **é** point de code **U+00E9**

Suite d'octets (en binaire)	Bits codant
0xxxxxxx	7 bits
110xxxxx 10xxxxxx	11 bits
1110xxxx 10xxxxxx 10xxxxxx	16 bits
11110xxx 10xxxxxx 10xxxxxx 10xxxxxx	21 bits

Tableau 6 – Encodage UTF-8

- lettre **é** point de code **U+00E9**
- $00E9_{hex} = 11101001_{bin}$, il faut 8 bits pour encoder la lettre **é**

$00E9_{hex} = 11101001_{bin}$, il faut 8 bits pour encoder la lettre **é**

110xxxxx	10xxxxxx
110xxx**11**	10**101001**

Encodage des caractères

Première tentative de normalisation : ASCII

Prise en compte des différents langages : ISO 8859

Encodage universel

Nouvelle norme

Représentation en mémoire

$00E9_{hex} = 11101001_{bin}$, il faut 8 bits pour encoder la lettre **é**

110xxxxx	10xxxxxx
11000**11**	10**101001**

Pour encoder la lettre **é** la norme UTF-8 utilise 2 octets.

Remarques

La norme UTF-8 est utilisée dans :

- plus de 95% des sites web,
- de nombreux langages de programmation (Python).