

TD1 : Codage et sous-réseaux

M1-DNR2I

Gaétan Richard

2 janvier 2012

Dans tout le cours, nous avons uniquement parlé de bits et d'octets (*i.e.* de codage binaire). Voyons un peu ce que l'on peut faire avec des 0 et des 1.

1 Codage des entiers

Pour représenter les entiers, on utilise couramment le codage avec les chiffres de 0 à 9 — c'est à dire le codage décimal. Il est bien sûr possible de coder à l'aide de 0 et de 1 (codage binaire).

Question 1 *Écrivez les chiffres de 0 à 10 en binaire.*

Question 2 *Combien de nombres est-il possible de coder avec un octet (8 bits) ?*

Il est également possible de coder en utilisant les chiffres de 0 à 9 et les lettres A à F. On parle alors de codage hexadécimal.

Question 3 *Combien vaut en décimal les nombres A0, 2C ?*

Question 4 *Combien peut-on coder de nombre avec 2 chiffres hexadécimaux ?
Comparez ce chiffre avec celui obtenu pour le codage en binaire sur un octet.*

Sur une machine, les éléments de base font soit 32, soit 64 bits.

Question 5 *Comment faire pour représenter des entiers plus grands. Existe-t-il une seule solution ?*

2 Codage des caractères

Le codage ASCII est donné par la figure 1.

Question 6 *Que représente la chaîne 42 6F 6E 6A 6F 75 72 21 ?*

Question 7 *Comment se représente la chaîne "hello world!" ?*

Ctrl	Dec	Hex	Char	Code	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
^@	0	00		NUL	32	20	sp	64	40	@	96	60	`
^A	1	01	␣	SOH	33	21	!	65	41	A	97	61	a
^B	2	02	␣	SIX	34	22	"	66	42	B	98	62	b
^C	3	03	♥	ETX	35	23	#	67	43	C	99	63	c
^D	4	04	♦	EOI	36	24	\$	68	44	D	100	64	d
^E	5	05	♣	ENQ	37	25	%	69	45	E	101	65	e
^F	6	06	♠	ACK	38	26	&	70	46	F	102	66	f
^G	7	07	+	BEL	39	27	'	71	47	G	103	67	g
^H	8	08	␣	BS	40	28	(72	48	H	104	68	h
^I	9	09	␣	HI	41	29)	73	49	I	105	69	i
^J	10	0A	␣	LF	42	2A	*	74	4A	J	106	6A	j
^K	11	0B	␣	VI	43	2B	+	75	4B	K	107	6B	k
^L	12	0C	␣	FF	44	2C	,	76	4C	L	108	6C	l
^M	13	0D	␣	CR	45	2D	-	77	4D	M	109	6D	m
^N	14	0E	␣	SO	46	2E	.	78	4E	N	110	6E	n
^O	15	0F	␣	SI	47	2F	/	79	4F	O	111	6F	o
^P	16	10	␣	SLE	48	30	0	80	50	P	112	70	p
^Q	17	11	␣	CS1	49	31	1	81	51	Q	113	71	q
^R	18	12	␣	DC2	50	32	2	82	52	R	114	72	r
^S	19	13	!!	DC3	51	33	3	83	53	S	115	73	s
^T	20	14	␣	DC4	52	34	4	84	54	T	116	74	t
^U	21	15	␣	NAK	53	35	5	85	55	U	117	75	u
^V	22	16	␣	SYN	54	36	6	86	56	V	118	76	v
^W	23	17	␣	EIB	55	37	7	87	57	W	119	77	w
^X	24	18	↑	CAN	56	38	8	88	58	X	120	78	x
^Y	25	19	↓	EM	57	39	9	89	59	Y	121	79	y
^Z	26	1A	→	SIB	58	3A	:	90	5A	Z	122	7A	z
^[_	27	1B	←	ESC	59	3B	;	91	5B	[123	7B	{
^`	28	1C	␣	FS	60	3C	<	92	5C	\	124	7C	
^]	29	1D	␣	GS	61	3D	=	93	5D]	125	7D	}
^^	30	1E	▲	RS	62	3E	>	94	5E	^	126	7E	~
^-	31	1F	▼	US	63	3F	?	95	5F	_	127	7F	Δ†

FIGURE 1 – Codage ASCII

Question 8 *Ce code suffit-il à coder l'intégralité du français ?*

Question 9 *Comment peut-on coder des chaînes de caractères ?*

Question 10 *On suppose que dans la première chaîne de cette section, on inverse le 18^{ième} bit, que devient le message ?*

3 Netmask

Le *netmask* (ou masque réseau en français) indique (en binaire) quels bits de l'adresse ipv4 appartiennent au sous-réseau.

Question 11 *Convertir en binaire le masque 255.255.255.0. Indiquer les adresses contenues dans la plage 192.168.0.0 avec un netmask de 255.255.255.0*

Il est également possible de noter directement le nombre de bits utilisés par le réseau en décimal.

Question 12 *Indiquer le netmask du réseau 172.16/12 et indiquer les adresses contenues dans cette plage.*