日本語文字コード

- 文字と計算機上の符号(数値)を対応づけるための枠組み
- 現在，JIS,シフトJIS,EUCなどの異なった日本語文字コードが混在している
- 解釈の枠組みが異なれば記号の意味が異なってしまう例
  - プログラムが想定するコード体系と異なると，「文字化け」が起こる
- コード体系の標準化･統一化には困難も多い
  - 歴史的経緯、利害関係、処理の都合、拡張性、文字セットの違い

情報表現の歴史

- 文字(メソポタミア, BC3000+)
- 算盤(バビロニア, BC1100+)
- 印刷(中国, 2世紀)
- 0を使った記数法 (インド, 5–6世紀)
- 楽譜(グレゴリオ聖歌, 9–12世紀)
- 活版印刷(グーテンベルグ, 1450?)

オルゴール(スイス, 1770?)
モールス符号(モールス, 1830s)
ファクシミリ(1843)
電話(1870s)
レコード(1877)
テープレコーダー(1900)
2進数(アタナソフ&ベリーの計算機, 1937−42)

文字の符号化: 歴史

- 手旗信号(18世紀～): 2本の旗の向きによって表現。
- モールス符号(1830s): 長点、短点の並びで表現。文字によって長短点の個数が違う。空白が区切り。
- テレタイプライタ(1920s): 5個のON/OFFの信号の並び(5ビット=32通り)
- ASCIIコード(1960s): 7個のON/OFFの信号の並び(7ビット=128通り)

画像は以下よりhttp://www.shurdington.org/Scouts/Semaphore.htm

ブリタニカ百科辞典より

www.mssu.edu/ seg-vm/pict0475.htmlより

発展

文字の符号化の問題

- 歴史的性質: 過去に符号化された文字は読めるべき
- 転送・記録の効率: 短い符号化→速く転送・沢山記録
- 文字の量: ヨーロッパ語は数十文字・漢字は数千以上
- 複数の標準: メーカーごと、地域ごとに決定
- 細かな、しかし文化的には無視できない違い: 見た目の類似性、異体字、方言ごとに異なる文字
- 国際化: 狭いコミュニティだけの使用→世界中のコンピュータが通信をする時代、多言語の同時使用

文字の符号化: ASCII

- 米国におけるアルファベットの符号化方式 (ASCII = American Standard Code for Information Interchange)
- 7ビットで128文字を表現
- 表わされる文字: アルファベット(大文字・小文字)・数字・記号・制御文字(テレタイプライタ等への「次の行へ」「一文字訂正」という指示)

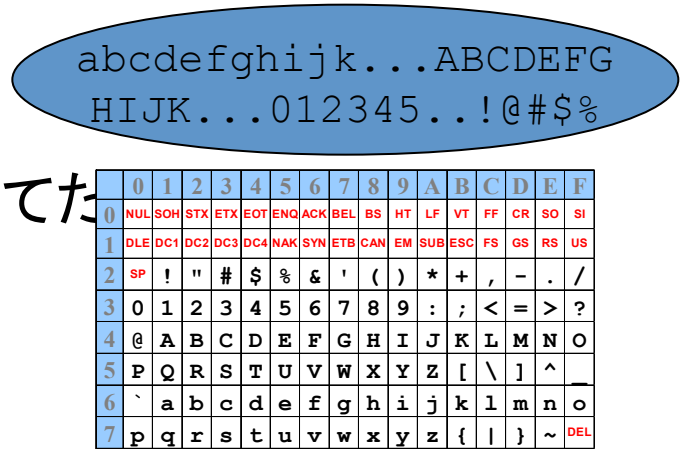
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

問題: 多文字・多国語

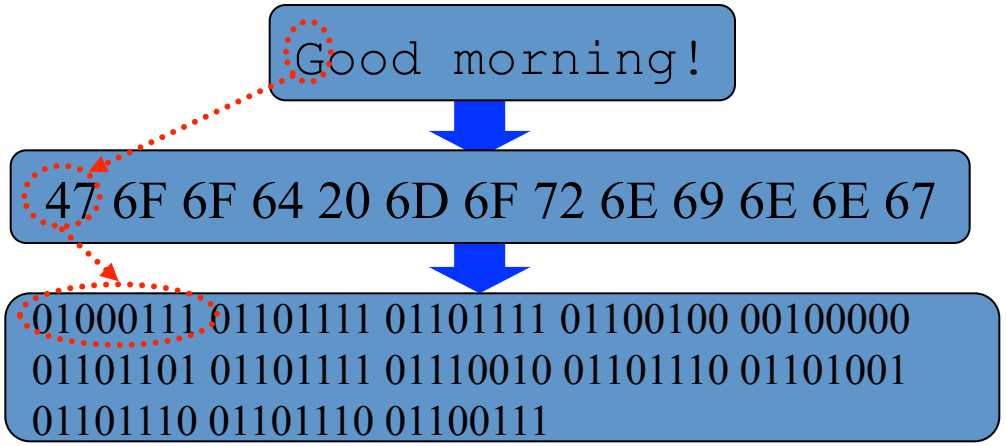
発展

文字符号化の整理

- 文字集合 ～ 符号化する文字の集まり
- 文字コード
  - 文字集合の各文字に割り当てた**番号**
  - 文字集合ごとに決める
- 符号化方式
  - 文字コードをビットの並びにする方法
  - 複数の文字集合を混在させる場合もある

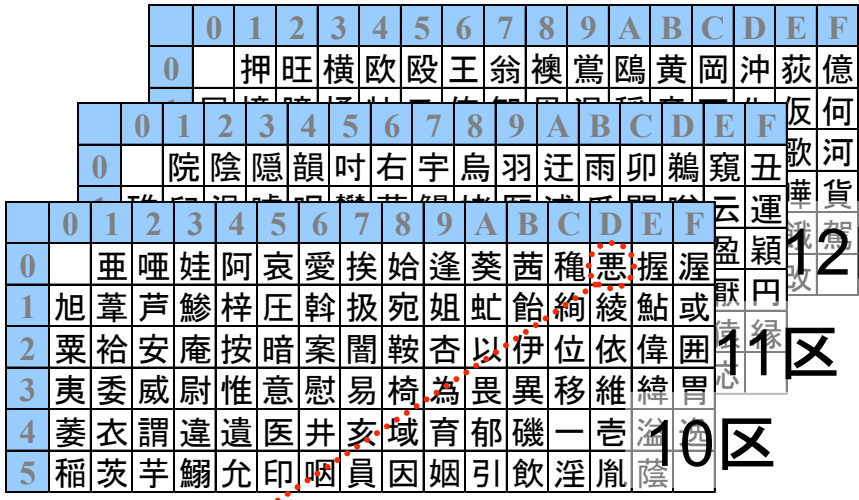


	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL



日本語の文字コードと符号化

- 文字集合: 漢字・仮名・記号約7000字
- 文字コード: JISコード (区点コード)
  - 1文字につき1～54 (1～84)までの区コードと1～5E(1～94)までの点コードを割り当て
  - (あるいは、(区+20) × 100 + (点+20) のコードを割り当てているとも言う)
- 符号化
  - 3つの方式: JIS方式、シフトJIS、EUC
    ← ASCII文字集合との共存のための異なる工夫



悪…10区0D点(16区13点)

または((10+20)×100)+(0D+10)
=301D (12317)

日本語の符号化: JIS方式

- 日本語と英語を切り替えながら7ビット単位で符号化
  - 英語はASCII符号化そのまま
  - 日本語は区+20, 点+20の2つの7ビットコード
  - 1B, 24, 42 というコードの後は日本語
  - 1B, 28, 42 というコードの後はASCII符号化に戻る
- 特徴
  - ASCII文字と同じ範囲しか使わない
    →ASCII文字を仮定して作られたシステムでそのまま使える
  - 多国語を混在させることが可能←切替コードを使うため
  - 前から順に見てゆかないと分からない

a	b	c	(日本語へ切替)			情		報		(英語へ切替)			e	f
97	98	99	27	36	66	62	112	74	115	27	40	66	101	102
61	62	63	1B	24	42	3E	70	4A	73	1B	28	42	65	66

日本語の符号化: EUC方式

- 日本語と英語のコードが重ならないように8ビット単位で符号化
  - 英語はASCII符号化そのまま
  - 日本語は、区+90, 点+90 (16進数)を8ビット2つの並びで表現
- 特徴
  - 日本語と英語、のような2ヶ国語のみ可能 (EUC韓国語, EUC中国語もあるが、混在はできない)
  - どの1バイトを見ても、英語か日本語かの区別ができる→処理が簡単
  - 切替がないので表現が短い

a	b	c	情		報		e	f
97	98	99	190	240	202	243	101	102
61	62	63	BE	F0	CA	F3	65	66

日本語の符号化: シフトJIS方式

- 日本語と英語とカナを8ビット単位で符号化
  - 英語はASCII符号化そのまま
  - 日本語は、区点コードを1バイト目が80～9FあるいはE0～EFに2バイト目が40～FCの範囲の並びで表現
  - 仮名1文字はA0からDFの1バイト (いわゆる半角仮名)
- 特徴
  - 日本語と英語の2ヶ国語のみ可能
  - 日本語の2バイト目と英語は同じコードになることがある
  - 切替がないので表現が短い

a	b	c	情		報		e	f
97	98	99	143	238	149	241	101	102
61	62	63	8F	EE	95	F1	65	66

多国語の符号化方式: Unicode

- 文字集合: 世界の多くの言語の約95,000字
- コード: 32ビット
- 符号化: UTF-8, UTF-16, UTF-32の主な3方式
  - UTF-8: ASCII文字は1バイト ヨーロッパ・アラビア文字は2バイト、それ以外(日韓中印・・・)は3バイト
  - UTF-16: ほとんどの文字は1文字を2バイトで表現

アルメニア語の大文字のPeh
コード:054A

アラビア語のDad
コード:0636

			a		b		c		情			報			e		f	
UTF-8			97		98		99		230	131	133	229	160	177	101		102	
			61		62		63		E6	83	85	E5	A0	B1	65		66	
UTF-16	255	254	97	0	98	0	99	0	197		96	49		88	101	0	102	0
	FF	FE	61	0	62	0	63	0	C5		60	31		58	65	0	66	0

発展

符号化にまつわる問題

- 「文字化け」: ある方式で符号化された文章を、別の符号化方式だと思って表示する
- 文字=文化の統一の難しさ
  - Han unification: 中日韓国語の漢字で見た目が同じ文字に同じコード(🆆英語のAとギリシャ語のAは違うコード)
  - 独自の文字: 携帯電話メールの絵文字
- 見た目が同じなのに違うコード (例: microsoft.comからのメール)