

Date: January 8, 1973

To: Members of FIPS TG-3, TG-12 and ANSI X3L2

From: John L. Little
Office of Information Processing Standards
National Bureau of Standards

Subject: ASCII and EBCDIC Sequencing

Table A has been prepared to illustrate the fundamental significance of file sequencing as determined by computer character coding. A sort key of three positions is shown, containing decimal digits (10), lower case letters (26) and upper case letters (26). Special symbols, such as punctuation marks are not included in the keys, to simplify the example. ASCII and EBCDIC code tables are enclosed for convenience in reviewing Table A.

Each position in the key can assume $10 + 26 + 26 = 62$ different values. Sequence positions 62, $(62)^2=3,844$ and $(62)^3=238,328$ correspond to complete cycles through 1, 2 and 3 key positions in either code. For ASCII coding, the progression is digits, upper case, lower case, while for EBCDIC coding the progression is lower case, upper case, digits.

Classes of symbols, such as decimal digits, are scattered about in both sequences. Few, if any, applications would prefer either of these sequences for use by people. Nor can a suitable sequence for people be provided by a code table placement of characters. Special provision must be made to yield a reasonable "display sequence."

If the sort key were restricted to digits only, then the progression from 000 to 999 would be the same in both ASCII and EBCDIC, but only 1000 items could be represented by a 3-character key. If the key were restricted to upper case letters, then the progression from AAA to ZZZ would be the same in ASCII and EBCDIC, and $(26)^3=17,576$ items could be represented. Likewise, if the key were restricted to lower case letters, the progression from aaa to zzz would be the same for 17,576 items.

If each position in the key were restricted to one class, then a progression such as OaA to 9zZ would also sequence the same in ASCII or EBCDIC, but only $10 \times 26 \times 26 = 6760$ items could be represented by that 3-character key.

An NCR Century Series or other ASCII computer would find it cumbersome to use the EBCDIC sequence in large files. Likewise, an IBM 370 Series or other EBCDIC computer finds it cumbersome to maintain large files in ASCII sequence (unless sort key restrictions make it the same as EBCDIC sequence).

ASCII and EBCDIC Sequencing

It should be clear from the illustration of Table A that standardization of a file sequence is vital in today's computer-based world. FIPS Task Group 12 and ANSI Technical Committee X3L2 each have a responsibility of great magnitude in resolving this matter which cannot be postponed much longer.

BITS E ₄ E ₃ E ₂ E ₁ ↓ ↓ ↓ ↓					COLUMN															
					0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	NUL	DLE	SP	0	@	P	\	p								
0	0	0	1	0	SOH	DC1	!	1	A	Q	a	q								
0	0	1	0	0	STX	DC2	"	2	B	R	b	r								
0	0	1	1	0	ETX	DC3	#	3	C	S	c	s								
0	1	0	0	0	EOT	DC4	\$	4	D	T	d	t								
0	1	0	1	0	ENQ	NAK	%	5	E	U	e	u								
0	1	1	0	0	ACK	SYN	&	6	F	V	f	v								
0	1	1	1	0	BEL	ETB	/	7	G	W	g	w								
1	0	0	0	0	BS	CAN	(8	H	X	h	x								
1	0	0	1	0	HT	EM)	9	I	Y	i	y								
1	0	1	0	0	LF	SUB	*	:	J	Z	j	z								
1	0	1	1	0	VT	ESC	+	;	K	[k	{								
1	1	0	0	0	FF	FS	,	<	L	\	l	!								
1	1	0	1	0	CR	GS	-	=	M]	m	}								
1	1	1	0	0	SO	RS	.	>	N	^	n	~								
1	1	1	1	0	SI	US	/	?	O	_	o	DEL								

STANDARD REPRESENTATION OF ASCII IN 8 BITS

