ECMA

EUROPEAN COMPUTER MANUFACTURERS ASSOCIATION

STANDARD ECMA-53

REPRESENTATION OF SOURCE PROGRAMS FOR PROGRAM INTERCHANGE

APL
COBOL
FORTRAN
Minimal BASIC
PL/1

January 1978

Free copies of this **ECMA** standard are available from
ECMA European Computer Manufacturers Association
114 Rue du Rhône - 1204 Geneva (Switzerland)

ECMA
EUROPEAN COMPUTER MANUFACTURERS ASSOCIATION

STANDARD ECMA-53

REPRESENTATION OF SOURCE PROGRAMS FOR PROGRAM INTERCHANGE

—

APL
COBOL
FORTRAN
Minimal BASIC
PL/1

January 1978

BRIEF HISTORY

This ECMA Standard has been prepared by TC 1, the Technical Committee of ECMA for Coded Character Sets, in close co-operation with the relevant language committees of the Association. It represents a deliberate attempt to combine in one standard several elements of other standards together with prescriptions on implementation on media, thereby rendering interchange of source programs easier.

This Standard was passed as Standard ECMA-53 at the General Assembly of Dec. 13, 1977.

1. SCOPE

This Standard ECMA-53 defines the representation of source programs by means of standard character sets. These are:

- Standard ECMA-6 : 7-bit Coded Character Set
- Standard ECMA-35 : Extensions of the 7-bit Coded Character Set
- Standard ECMA-43 : 8-bit Coded Character Set.

The implementation of this representation on different media is also specified.

This Standard is directed at source programs written in APL, Minimal BASIC, COBOL, FORTRAN and PL/1.

2. CHARACTER SETS

2.1 APL

This Standard defines 90 characters for general use in interchange of APL programs. Fifty-seven of these characters are contained in the 7-bit Coded Character Set. The use of yet other characters, in particular of so-called combined characters, requires agreement between sender and recipient. One of the 90 characters, the character AMPERSAND, does not belong to the APL language character set. It is used only in the representation of APL source programs on media for program interchange (see 6.1).

2.2 Minimal BASIC

The Minimal BASIC language character set comprises 60 characters, all contained in the 7-bit Coded Character Set.

2.3 COBOL

The COBOL language character set comprises 51 characters, all contained in the 7-bit Coded Character Set. Beside these, the COBOL language allows an implementation to provide for other characters to be used in "non-numeric literals", "comment-entries" and "comment lines". In order to interchange such data, agreement is required between sender and recipient.

2.4 FORTRAN

The FORTRAN language character set comprises 49 characters, all contained in the 7-bit Coded Character Set. Beside these, the FORTRAN language allows an implementation to provide for other characters to be used in "character constants" and in "comments". In order to interchange such data, agreement is required between sender and recipient.

2.5 PL/1

The PL/1 language character set comprises 57 characters, two of which are not contained in the 7-bit Coded Character Set. Beside these, the PL/1 language allows an implementation to

provide for other characters to be used in "character-string-constants" and in "comments". In order to interchange such data, agreement is required between sender and recipient.

3. CODING

3.1 APL

The characters of the APL language defined in this Standard shall be coded according to Table 2.1.

3.2 Minimal BASIC

The characters of the Minimal BASIC language character set shall be coded according to Table 2.2.

3.3 COBOL

The characters of the COBOL language character set shall be coded according to Table 2.3.

3.4 FORTRAN

The characters of the FORTRAN language character set shall be coded according to Table 2.4.

3.5 PL/1

The characters of the PL/1 language character set shall be coded according to Table 2.5.

4. NOTES ON PARTICULAR CHARACTERS

4.1 No printed graphic corresponds to the character BLANK (symbol ␣ in PL/1) which is used in the FORTRAN and PL/1 language character sets.

The abbreviation SP is used hereafter to indicate the non-printing character SPACE of the standard Coded Character Sets.

4.2 In the 7-bit and in the 8-bit code tables, two characters are allocated to pos. 2/4, namely $ and ¤. In any version of the codes a single character is to be allocated to this position. The character of the 7-bit or of the 8-bit Coded Character Set, which corresponds to the character $ of the language character sets of Minimal BASIC, COBOL, FORTRAN and PL/1 is either $ or ¤ (¤ in the International Reference Version).

The same applies to pos. 2/3 for the characters £ and #, the latter being the character of the International Reference Version of the 7-bit Coded Character Set.

5. CHARACTER SETS RELATIONSHIP

The tables indicate:

Table 1 : the correspondence between the language character sets and the 7-bit Coded Character Set,

Table 2 : the 7-bit coding of the characters of the language character sets. All characters contained in the 7-bit Coded Character Set have retained their coding.

The correspondence with the 8-bit Coded Character Set and the 8-bit coding of the language character sets is to be derived from the relationship between 7-bit and 8-bit Coded Character Sets.

6. SOURCE PROGRAM INTERCHANGE

6.1 APL

An APL source program consists of statements, which may be of different lengths. A statement comprises one or more lines. Each statement begins on a new line.

For program interchange a line shall have a maximum length of 72 characters. If a statement extends over more than one line, the last character of each line, with the exception of the last line of the statement, shall be the character AMPERSAND. It shall be considered as not being part of the APL statement. The character preceding the character AMPERSAND shall be treated as if it immediately preceded the first character of the next line. The representation on media for interchange shall be as follows:

6.1.1 Punched Tape

The representation of the characters is given in ECMA-10. A source program shall be preceded and followed by at least ten characters NUL. Lines shall be separated by the characters CR and LF.

6.1.2 Punched Cards

The representation of the characters is given in ECMA-44. Each line shall have a fixed length of 71 characters and be recorded in columns 2 to 72. Column 1 shall be left unpunched. Columns 73 to 80 may be used for identification or other purposes.

6.1.3 Magnetic Tapes

The representation of the characters is given in ECMA-5, ECMA-12 and ECMA-36. A source program shall form one file. Each line shall form a variable-length record; or shall form one fixed-length record of not more than 72 characters; or shall be contained in a fixed-length record of 80 characters.

If a fixed-length record of 80 characters is used, a line shall comprise 71 characters recorded in positions 2 to 72 of the record. In position 1 the character SPACE shall be recorded. Positions 73 to 80 may be used for identification or other purposes.

Labelling shall be according to ECMA-13.

6.1.4 Magnetic Tape Cassettes and Cartridges

The representation of the characters is given in ECMA-34 and ECMA-46. A source program shall form one file. Each line shall

form a variable-length record; or shall form one fixed-length record of not more than 72 characters; or shall be contained in a fixed-length record of 80 characters.

Labelling shall be according to ECMA-41.

If a fixed-length record of 80 characters is used, a line shall comprise 71 characters recorded in positions 2 to 72 of the record. In position 1 the character SPACE shall be recorded. Positions 73 to 80 may be used for identification or other purposes.

6.2 Minimal BASIC

A Minimal BASIC source program consists of statements, one per line, beginning with a line number. The maximum length of a line is 72 characters. The representation on media for interchange shall be as follows:

6.2.1 Punched Tape

The representation of the characters is given in ECMA-10. A source program shall be preceded and followed by at least ten characters NUL. Lines shall be separated by the characters CR and LF.

6.2.2 Punched Cards

The representation of the characters is given in ECMA-44. Each line shall have a fixed length of 72 characters and be recorded in columns 1 to 72. The remaining columns shall be left unpunched.

6.2.3 Magnetic Tapes

The representation of the characters is given in ECMA-5, ECMA-12 and ECMA-36. A source program shall form one file. Each line shall form one fixed-length record of either 72 or 80 characters. If the record length is 80 characters, only the first 72 characters are part of the source program; the last eight characters shall be SPACEs.

Labelling shall be according to ECMA-13.

6.2.4 Magnetic Tape Cassettes and Cartridges

The representation of the characters is given in ECMA-34 and ECMA-46. A source program shall form one file. Each line shall form one fixed-length record of either 72 or 80 characters. If the record length is 80 characters, only the first 72 characters are part of the source program; the last eight characters shall be SPACEs.

Labelling shall be according to ECMA-41.

6.3 COBOL

A COBOL source program consists of lines, each containing 72 characters. The representation on media for interchange shall be as follows:

6.3.1 Punched Tape

The representation of the characters is given in ECMA-10. A

source program shall be preceded and followed by at least ten characters NUL. Lines shall be separated by the characters CR and LF.

6.3.2 Punched Cards

The representation of the characters is given in ECMA-44. Each line shall be recorded in columns 1 to 72. The remaining columns may be used for identification or other purposes.

6.3.3 Magnetic Tapes

The representation of the characters is given in ECMA-5, ECMA-12 and ECMA-36. A source program shall form one file. Each line shall form one fixed-length record of either 72 or 80 characters. If the record length is 80 characters, only the first 72 characters are part of the source program; the last eight characters may be used for identification or other purposes.

Labelling shall be according to ECMA-13.

6.3.4 Magnetic Tape Cassettes and Cartridges

The representation of the characters is given in ECMA-34 and ECMA-46. A source program shall form one file. Each line shall form one fixed-length record of either 72 or 80 characters. If the record length is 80 characters, only the first 72 characters are part of the source program; the last eight characters may be used for identification or other purposes.

Labelling shall be according to ECMA-41.

6.4 FORTRAN

A FORTRAN source program consists of lines each having a maximum length of 72 characters. The representation on media for interchange shall be as follows:

6.4.1 Punched Tape

The representation of the characters is given in ECMA-10. A source program shall be preceded and followed by at least ten characters NUL. Lines shall be separated by the characters CR and LF.

6.4.2 Punched Cards

The representation of the characters is given in ECMA-44. Each line shall have a fixed-length of 72 characters and be recorded in columns 1 to 72. The remaining columns may be used for identification or other purposes.

6.4.3 Magnetic Tapes

The representation of the characters is given in ECMA-5, ECMA-12 and ECMA-36. A source program shall form one file. Each line shall form one fixed-length record of either 72 or 80 characters. If the record length is 80 characters,

only the first 72 characters are part of the source program; the last eight characters may be used for identification or other purposes.

Labelling shall be according to ECMA-13.

6.4.4 Magnetic Tape Cassettes and Cartridges

The representation of the characters is given in ECMA-34 and ECMA-46. A source program shall form one file. Each line shall form one fixed-length record of either 72 or 80 characters. If the record length is 80 characters, only the first 72 characters are part of the source program; the last eight characters may be used for identification or other purposes.

Labelling shall be according to ECMA-41.

6.5 PL/1

A PL/1 source program in accordance with ECMA-50 consists of one or more "external procedures". An "external procedure" consists of a continuous sequence of characters. Although a line concept is not defined in the PL/1 language, this sequence of characters may be divided into lines to permit the representation on media. The last character of a line shall be treated as if it immediately preceded the first character of the next line. For program interchange the maximum length of a line shall be 72 characters. The representation on media for interchange shall be as follows:

6.5.1 Punched Tape

The representation of the characters is given in ECMA-10. An "external procedure" shall be preceded and followed by at least ten characters NUL. Lines shall be separated by the characters CR and LF.

6.5.2 Punched Cards

The representation of the characters is given in ECMA-44. Each line shall have a fixed length of 71 characters and be recorded in columns 2 to 72. Column 1 shall be left unpunched. Columns 73 to 80 may be used for identification or other purposes.

6.5.3 Magnetic Tapes

The representation of the characters is given in ECMA-5, ECMA-12 and ECMA-36. An "external procedure" shall form one file. Each line shall form a variable-length record; or shall form one fixed-length record of not more than 72 characters; or shall be contained in a fixed-length record of 80 characters.

If a fixed-length record of 80 characters is used, a line shall comprise 71 characters recorded in positions 2 to 72 of the record. In position 1 the character SPACE shall be recorded. Positions 73 to 80 may be used for identification or other purposes.

Labelling shall be according to ECMA-13.

6.5.4 Magnetic Tape Cassettes and Cartridges

The representation of the characters is given in ECMA-34 and ECMA-46. An "external procedure" shall form one file. Each line shall form a variable-length record; or shall form one fixed-length record of not more than 72 characters; or shall be contained in a fixed-length record of 80 characters.

If a fixed-length record of 80 characters is used, a line shall comprise 71 characters recorded in positions 2 to 72 of the record. In position 1 the character SPACE shall be recorded. Positions 73 to 80 may be used for identification or other purposes.

Labelling shall be according to ECMA-41.

TABLE 1

ECMA 7-bit Code		Minimal BASIC		COBOL		FORTRAN		PL/1	
Name	Graphic	Name	Graphic	Name	Graphic	Name	Graphic	Name	Graphic
Space	SP	Space		Space		Blank		Blank	b
Exclamation mark	!	Exclamation mark	!					Or	|
Quotation mark, Diaeresis	"	Quotation mark	"	Quotation mark	"				
Pound sign or Number sign	£ #	Number sign	#						
Dollar sign or Currency sign	$ ¤	Dollar sign	$	Currency sign	$	Currency symbol	$	Dollar	$
Percent sign	%	Percent sign	%					Percent	%
Ampersand	&	Ampersand	&					And	&
Apostrophe, Acute accent	'	Apostrophe	'			Apostrophe	'	Single quote	'
Left parenthesis	(	Left parenthesis	(	Left parenthesis	(	Left parenthesis	(	Left parenthesis	(
Right parenthesis	)	Right parenthesis	)	Right parenthesis	)	Right parenthesis	)	Right parenthesis	)
Asterisk	*	Asterisk	*	Asterisk	*	Asterisk	*	Asterisk	*
Plus sign	+	Plus sign	+	Plus sign	+	Plus	+	Plus	+
Comma, Cedilla	,	Comma	,	Comma	,	Comma	,	Comma	,
Hyphen, Minus sign	-	Minus sign	-	Minus sign	-	Minus	-	Minus	-
Full stop, Period	.	Full stop	.	Period	.	Decimal point	.	Period	.
Solidus	/	Solidus	/	Stroke	/	Slash	/	Slash	/
Digits	0-9	Digits	0-9	Digit	0-9	Digits	0-9	Digits	0-9
Colon	:	Colon	:			Colon	:	Colon	:
Semi-colon	;	Semi-colon	;	Semicolon	;			Semicolon	;
Less than sign	<	Less than sign	<	Less than symbol	<			Less than	<
Equals sign	=	Equals sign	=	Equal sign	=	Equals	=	Equal	=
Greater than sign	>	Greater than sign	>	Greater than symbol	>			Greater than	>
Question mark	?	Question mark	?						
Commercial at	@								
Capital letters	A-Z	Letters	A-Z	Letter	A-Z	Letters	A-Z	Letters	A-Z
Upward arrow head, Circumflex accent	^	Circumflex accent	^					Not	¬
Underline	_	Underline	_					Break	-

TABLE 1 continued

ECMA 7-bit Code		APL	
Name	Graphic	Name	Graphic
Space	SP	Space	
Exclamation mark	!		
Quotation mark, Diaeresis	"	Diaeresis	"
Pound sign or number sign	£ #		
Dollar sign or Currency sign	$ ¤		
Percent sign	%		
Ampersand	&	Ampersand	&
Apostrophe, Acute accent	'	Single quote	'
Left parenthesis	(	Left parenthesis	(
Right parenthesis	)	Right parenthesis	)
Asterisk	*	Asterisk	*
Plus sign	+	Plus sign	+
Comma, Cedilla	,	Comma	,
Hyphen, Minus sign	-	Minus sign	-
Full stop, Period	.	Full stop	.
Solidus	/	Solidus	/
Digits	0-9	Digits	0-9
Colon	:	Colon	:
Semi-colon	;	Semi-colon	;
Less than sign	<	Less than sign	<
Equals sign	=	Equals sign	=
Greater than sign	>	Greater than sign	>
Question mark	?	Question mark	?
Commercial at	@		
Capital letters	A-Z	Letters	A-Z
Left square bracket	[	L. square bracket	[
Reverse solidus	\	Reverse solidus	\
Right square bracket	]	R. square bracket	]
Upward arrow head, Circumflex accent	^	Upward arrow	↑
Underline	_	Underline	_
Grave accent	`	Left arrow	←

ECMA 7-bit Code		APL	
Name	Graphic	Name	Graphic
Small letter a	a	Alpha	⍺
Small letter b	b	Decode	⊥
Small letter c	c	Intersection	∩
Small letter d	d	Floor	⌊
Small letter e	e	Element of	∈
Small letter f	f	Multiply sign	×
Small letter g	g	Nabla	∇
Small letter h	h	Delta	∆
Small letter i	i	Iota	⍳
Small letter j	j	Small circle	∘
Small letter k	k	Divide sign	÷
Small letter l	l	Quad	⎕
Small letter m	m	Not equal to	≠
Small letter n	n	Encode	⊤
Small letter o	o	Large circle	○
Small letter p	p	Right arrow	→
Small letter q	q	Logical Or	∨
Small letter r	r	Rho	⍴
Small letter s	s	Ceiling	⌈
Small letter t	t	Tilde	~
Small letter u	u	Downward arrow	↓
Small letter v	v	Union	∪
Small letter w	w	Omega	⍵
Small letter x	x	Super-Set	⊃
Small letter y	y	Logical And	∧
Small letter z	z	Sub-Set	⊂
Left curly bracket	{	Less th. or eq. to	≤
Vertical line	|	Vertical line	|
Right curly bracket	}	Gr. th. or eq. to	≥
Overline	‾	Overline	‾

TABLE 2.1

APL

b_7	0	0	0	0	1	1	1	1
b_6	0	0	1	1	0	0	1	1
b_5	0	1	0	1	0	1	0	1
	0	1	2	3	4	5	6	7
b_4 b_3 b_2 b_1								
0 0 0 0 0			SP	0		P	←	→
0 0 0 1 1				1	A	Q	⍺	∨
0 0 1 0 2			"	2	B	R	⊥	⍴
0 0 1 1 3				3	C	S	∩	⌈
0 1 0 0 4				4	D	T	⌊	~
0 1 0 1 5				5	E	U	∈	⍋
0 1 1 0 6			&	6	F	V	×	∪
0 1 1 1 7			'	7	G	W	⍛	⍵
1 0 0 0 8			(	8	H	X	⍙	⊃
1 0 0 1 9			)	9	I	Y	⍳	∧
1 0 1 0 10			*	:	J	Z	○	⊂
1 0 1 1 11			+	;	K	[	÷	≤
1 1 0 0 12			,	<	L	\	⌷	∣
1 1 0 1 13			-	=	M	]	≠	≥
1 1 1 0 14			.	>	N	↑	⍎	−
1 1 1 1 15			/	?	O	_	⍕	

TABLE 2.2

Minimal BASIC

b_4	b_3	b_2	b_1		b_7 0	0	0	0	1	1	1	1
					b_6 0	0	1	1	0	0	1	1
					b_5 0	1	0	1	0	1	0	1
					0	1	2	3	4	5	6	7
0	0	0	0	0			SP	0		P		
0	0	0	1	1			!	1	A	Q		
0	0	1	0	2			"	2	B	R		
0	0	1	1	3			#	3	C	S		
0	1	0	0	4			$	4	D	T		
0	1	0	1	5			%	5	E	U		
0	1	1	0	6			&	6	F	V		
0	1	1	1	7			'	7	G	W		
1	0	0	0	8			(	8	H	X		
1	0	0	1	9			)	9	I	Y		
1	0	1	0	10			*	:	J	Z		
1	0	1	1	11			+	;	K			
1	1	0	0	12			,	<	L			
1	1	0	1	13			-	=	M			
1	1	1	0	14			.	>	N	^		
1	1	1	1	15			/	?	O	_		

TABLE 2.3

COBOL

b_7	0	0	0	0	1	1	1	1
b_6	0	0	1	1	0	0	1	1
b_5	0	1	0	1	0	1	0	1
	0	1	2	3	4	5	6	7
b_4 b_3 b_2 b_1								
0 0 0 0	0			SP	0		P		
0 0 0 1	1				1	A	Q		
0 0 1 0	2			"	2	B	R		
0 0 1 1	3				3	C	S		
0 1 0 0	4			$	4	D	T		
0 1 0 1	5				5	E	U		
0 1 1 0	6				6	F	V		
0 1 1 1	7				7	G	W		
1 0 0 0	8			(	8	H	X		
1 0 0 1	9			)	9	I	Y		
1 0 1 0	10			*		J	Z		
1 0 1 1	11			+	;	K			
1 1 0 0	12			,	<	L			
1 1 0 1	13			-	=	M			
1 1 1 0	14			.	>	N			
1 1 1 1	15			/		O			

TABLE 2.4

FORTRAN

b_4	b_3	b_2	b_1		b_7 0	0	0	0	1	1	1	1
					b_6 0	0	1	1	0	0	1	1
					b_5 0	1	0	1	0	1	0	1
					0	1	2	3	4	5	6	7
0	0	0	0	0			SP	0		P		
0	0	0	1	1				1	A	Q		
0	0	1	0	2				2	B	R		
0	0	1	1	3				3	C	S		
0	1	0	0	4			$	4	D	T		
0	1	0	1	5				5	E	U		
0	1	1	0	6				6	F	V		
0	1	1	1	7			'	7	G	W		
1	0	0	0	8			(	8	H	X		
1	0	0	1	9			)	9	I	Y		
1	0	1	0	10			*	:	J	Z		
1	0	1	1	11			+		K			
1	1	0	0	12			,		L			
1	1	0	1	13			-	=	M			
1	1	1	0	14			.		N			
1	1	1	1	15			/		O			

TABLE 2.5

PL/1

b_7	0	0	0	0	1	1	1	1
b_6	0	0	1	1	0	0	1	1
b_5	0	1	0	1	0	1	0	1

b_4	b_3	b_2	b_1		0	1	2	3	4	5	6	7
0	0	0	0	0			ъ	0		P		
0	0	0	1	1			ı	1	A	Q		
0	0	1	0	2				2	B	R		
0	0	1	1	3				3	C	S		
0	1	0	0	4			$	4	D	T		
0	1	0	1	5			%	5	E	U		
0	1	1	0	6			&	6	F	V		
0	1	1	1	7			'	7	G	W		
1	0	0	0	8			(	8	H	X		
1	0	0	1	9			)	9	I	Y		
1	0	1	0	10			*	:	J	Z		
1	0	1	1	11			+	;	K			
1	1	0	0	12			,	<	L			
1	1	0	1	13			-	=	M			
1	1	1	0	14			.	>	N	¬		
1	1	1	1	15			/		O	_		