

American National Standard

*INCITS/ISO/IEC 1989:2014 [2014]
(ISO/IEC 1989:2014, IDT)*

Reaffirmed as INCITS/ISO/IEC 1989:2014
(R2019)

*Information technology — Programming
languages, their environments and system
software interfaces — Programming language
COBOL*

Developed by



Where IT all begins



INCITS/ISO/IEC 1989:2014 [2014]

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

Adopted by INCITS (InterNational Committee for Information Technology Standards) as an American National Standard.

Date of ANSI Approval: 10/31/2014

Published by American National Standards Institute,
25 West 43rd Street, New York, New York 10036

Copyright 2014 by Information Technology Industry Council
(ITI). All rights reserved.

These materials are subject to copyright claims of International Standardization Organization (ISO), International Electrotechnical Commission (IEC), American National Standards Institute (ANSI), and Information Technology Industry Council (ITI). Not for resale. No part of this publication may be reproduced in any form, including an electronic retrieval system, without the prior written permission of ITI. All requests pertaining to this standard should be submitted to ITI, 1101 K Street NW, Suite 610, Washington DC 20005.

Printed in the United States of America

ACKNOWLEDGEMENT

Any organization interested in reproducing the COBOL standard and specifications in whole or in part, using ideas from this document as the basis for an instruction manual or for any other purpose, is free to do so. However, all such organizations are requested to reproduce the following acknowledgment paragraphs in their entirety as part of the preface to any such publication (any organization using a short passage from this document, such as in a book review, is requested to mention "COBOL" in acknowledgment of the source, but need not quote the acknowledgment):

COBOL is an industry language and is not the property of any company or group of companies, or of any organization or group of organizations.

No warranty, expressed or implied, is made by any contributor or by the CODASYL COBOL Committee as to the accuracy and functioning of the programming system and language. Moreover, no responsibility is assumed by any contributor, or by the committee, in connection therewith. The authors and copyright holders of the copyrighted materials used herein FLOW-MATIC (trademark of Sperry Rand Corporation), Programming for the UNIVAC (R) I and II, Data Automation Systems copyrighted 1958, 1959, by Sperry Rand Corporation; IBM Commercial Translator Form No F 28-8013, copyrighted 1959 by IBM; FACT, DSI 27A5260-2760, copyrighted 1960 by Minneapolis-Honeywell have specifically authorized the use of this material in whole or in part, in the COBOL specifications. Such authorization extends to the reproduction and use of COBOL specifications in programming manuals or similar publications.

Contents

Contents	iii
Tables	xx
Figures	xxi
Foreword	xxii
Introduction	xxiii
1 Scope	1
2 Normative references	2
3 Conformance to this International Standard	3
3.1 General	3
3.2 A conforming implementation	3
3.2.1 Acceptance of standard language elements	3
3.2.2 Interaction with non-COBOL runtime modules	3
3.2.3 Interaction between COBOL implementations	3
3.2.4 Implementor-defined language elements	3
3.2.5 Processor-dependent language elements	4
3.2.6 Optional language elements	4
3.2.7 Reserved words	4
3.2.8 Standard extensions	4
3.2.9 Nonstandard extensions	4
3.2.10 Substitute or additional language elements	5
3.2.11 Archaic language elements	5
3.2.12 Obsolete language elements	5
3.2.13 Externally-provided functionality	5
3.2.14 Limits	6
3.2.15 User documentation	6
3.2.16 Character substitution	6
3.3 A conforming compilation group	6
3.4 A conforming run unit	6
3.5 Relationship of a conforming compilation group to a conforming implementation	7
3.6 Relationship of a conforming run unit to a conforming implementation	7
4 Terms and definitions	8
5 Description techniques	24
5.1 General	24
5.2 General formats	24

5.2.1	Keywords	24
5.2.2	Optional words	24
5.2.3	Operands	24
5.2.4	Level numbers	25
5.2.5	Options	25
5.2.5.1	Brackets	25
5.2.5.2	Braces	25
5.2.5.3	Choice indicators	25
5.2.6	Ellipses	25
5.2.7	Punctuation	26
5.2.8	Special characters	26
5.2.9	Meta-terms	26
5.3	Rules	26
5.3.1	Syntax rules	26
5.3.2	General rules	26
5.3.3	Argument rules	26
5.3.4	Returned value rules	26
5.4	Arithmetic expressions	27
5.4.1	Textually subscripted operands	27
5.4.2	Ellipses	27
5.5	Integer operands	27
5.6	Informal description	27
5.7	Hyphens in text	27
5.8	Forms of common verbs used in this International Standard	28
5.9	Notes and examples	28
6	Reference format	29
6.1	General	29
6.2	Indicators	29
6.2.1	Fixed indicators	29
6.2.2	Floating indicators	30
6.3	Fixed-form reference format	31
6.4	Free-form reference format	33
6.5	Logical conversion	34
7	Compiler directing facility	36
7.1	General	36
7.2	Text manipulation	36
7.2.1	Text manipulation elements	37
7.2.2	COPY statement	39
7.2.3	REPLACE statement	43
7.3	Compiler directives	46
7.3.1	General format	46
7.3.2	Syntax rules	46
7.3.3	General rules	46
7.3.4	Conditional compilation	47

7.3.5	Compile-time arithmetic expressions	47
7.3.6	Compile-time boolean expressions	47
7.3.7	Constant conditional expression	48
7.3.8	CALL-CONVENTION directive	50
7.3.9	DEFINE directive	51
7.3.10	EVALUATE directive	52
7.3.11	FLAG-02 directive	55
7.3.12	FLAG-85 directive	57
7.3.13	FLAG-NATIVE-ARITHMETIC directive	59
7.3.14	IF directive	60
7.3.15	LEAP-SECOND directive	61
7.3.16	LISTING directive	62
7.3.17	PAGE directive	63
7.3.18	PROPAGATE directive	64
7.3.19	SOURCE FORMAT directive	65
7.3.20	TURN directive	66
8	Language fundamentals	68
8.1	Character sets	68
8.1.1	Computer's coded character set	68
8.1.2	COBOL character repertoire	70
8.1.3	Alphabets	72
8.1.4	Collating sequences	73
8.2	Locales	74
8.2.1	Locale field names	75
8.3	Lexical elements	76
8.3.1	Character-strings	76
8.3.1.1	COBOL words	76
8.3.1.2	Literals	84
8.3.1.3	Picture character-strings	92
8.3.2	Separators	92
8.4	References	94
8.4.1	Uniqueness of reference	94
8.4.1.1	Qualification	94
8.4.1.2	Subscripts	96
8.4.2	Identifiers	98
8.4.2.1	Identifier	98
8.4.2.2	Function-identifier	100
8.4.2.3	Reference-modification	103
8.4.2.4	Inline method invocation	105
8.4.2.5	Object-view	106
8.4.2.6	EXCEPTION-OBJECT	107
8.4.2.7	NULL object reference	107
8.4.2.8	SELF and SUPER	107
8.4.2.9	Object property	108
8.4.2.10	NULL address pointer	109

8.4.2.11	Data-address-identifier	110
8.4.2.12	Function-address-identifier	111
8.4.2.13	Program-address-identifier	112
8.4.2.14	LINAGE-COUNTER	112
8.4.2.15	Report counters	113
8.4.3	Condition-name	114
8.4.4	Explicit and implicit data item references	114
8.4.5	Scope of names	115
8.4.5.1	Local and global names	116
8.4.5.2	Scope of program-names	117
8.4.5.3	Scope of class-names and interface-names	118
8.4.5.4	Scope of method-names	118
8.4.5.5	Scope of function-prototype-names	118
8.4.5.6	Scope of user-function-names	118
8.4.5.7	Scope of program-prototype-names	118
8.4.5.8	Scope of compilation-variable-names	118
8.4.5.9	Scope of parameter-names	118
8.4.5.10	Scope of property-names	119
8.5	Data description and representation	120
8.5.1	Computer independent data description	120
8.5.1.1	Files and records	120
8.5.1.2	Levels	120
8.5.1.3	Limitations of character handling	121
8.5.1.4	Algebraic signs	121
8.5.1.5	Alignment of data items in storage	122
8.5.1.5.1	Alignment of alphanumeric groups and of data items of usage display	122
8.5.1.5.2	Alignment of data items of usage national	122
8.5.1.5.3	Alignment of data items of usage bit	122
8.5.1.5.4	Item alignment for increased object-code efficiency	123
8.5.1.5.5	Alignment of strongly-typed group items	123
8.5.1.6	Tables	123
8.5.1.6.1	Fixed-capacity tables	123
8.5.1.6.2	Occurs-depending tables	124
8.5.1.6.3	Dynamic-capacity tables	124
8.5.1.7	Dynamic-length elementary items	126
8.5.1.7.1	Structure of a dynamic-length elementary item	126
8.5.1.7.2	Location of dynamic-length elementary items	126
8.5.1.7.3	Operations on dynamic-length elementary items	126
8.5.1.8	Variable-length data items	127
8.5.1.8.1	Contiguity of data items	127
8.5.1.8.2	Availability and persistence of data items	127
8.5.1.9	Variable-length groups	127
8.5.1.9.1	Positional correspondence	128
8.5.1.9.2	Matching	128
8.5.2	Class and category of data items and literals	128
8.5.2.1	Alphabetic category	129

8.5.2.2	Alphanumeric category	129
8.5.2.3	Alphanumeric-edited category	130
8.5.2.4	Boolean category	130
8.5.2.5	Data-pointer category	130
8.5.2.6	Function-pointer category	130
8.5.2.7	Index category	130
8.5.2.8	National category	130
8.5.2.9	National-edited category	131
8.5.2.10	Numeric category	131
8.5.2.11	Numeric-edited category	131
8.5.2.12	Object-reference category	131
8.5.2.13	Program-pointer category	131
8.5.3	Types	131
8.5.3.1	Weakly-typed items	132
8.5.3.2	Strongly-typed group items	132
8.5.4	Zero-length items	133
8.6	Scope and life cycle of data	133
8.6.1	Global names and local names	133
8.6.2	External and internal items	133
8.6.3	Automatic, initial, and static internal items	134
8.6.4	Based entries and based data items	135
8.6.5	Common, initial, and recursive attributes	135
8.6.6	Sharing data items	136
8.7	Operators	137
8.7.1	Arithmetic operators	137
8.7.2	Boolean operators	137
8.7.3	Concatenation operator	137
8.7.4	Invocation operator	137
8.7.5	Relational operators	137
8.7.6	Logical operators	139
8.8	Expressions	140
8.8.1	Arithmetic expressions	140
8.8.1.1	Native, standard, standard-binary, and standard-decimal arithmetic	140
8.8.1.2	Native arithmetic	141
8.8.1.3	Standard arithmetic	141
8.8.1.3.1	Standard intermediate data item	141
8.8.1.3.2	Addition	143
8.8.1.3.3	Subtraction	143
8.8.1.3.4	Multiplication	144
8.8.1.3.5	Division	144
8.8.1.3.6	Exponentiation	144
8.8.1.3.7	Unary plus	145
8.8.1.3.8	Unary minus	145
8.8.1.4	Standard-binary arithmetic	145
8.8.1.4.1	Standard-binary intermediate data item	145
8.8.1.4.2	Basic arithmetic operations in standard-binary arithmetic	146

8.8.1.4.3 Exponentiation in standard-binary arithmetic	146
8.8.1.5 Standard-decimal arithmetic	147
8.8.1.5.1 Standard-decimal intermediate data item	147
8.8.1.5.2 Basic arithmetic operations in standard-decimal arithmetic	148
8.8.1.5.3 Exponentiation in standard-decimal arithmetic	148
8.8.2 Boolean expressions	149
8.8.3 Concatenation expressions	152
8.8.4 Conditional expressions	152
8.8.4.1 Simple conditions	152
8.8.4.1.1 Relation conditions	153
8.8.4.1.2 Boolean condition	158
8.8.4.1.3 Class condition	159
8.8.4.1.4 Condition-name condition (conditional variable)	161
8.8.4.1.5 Switch-status condition	162
8.8.4.1.6 Sign condition	162
8.8.4.1.7 Omitted-argument condition	163
8.8.4.2 Complex conditions	164
8.8.4.2.1 Negated conditions	164
8.8.4.2.2 Combined conditions	164
8.8.4.2.4 Abbreviated combined relation conditions	165
8.8.4.3 Order of evaluation of conditions	166
8.9 Reserved words	167
8.10 Context-sensitive words	170
8.11 Intrinsic function names	173
8.12 Compiler-directive words	174
8.13 External repository	175
9 I-O, objects, and user-defined functions	176
9.1 Files	176
9.1.1 Physical and logical files	176
9.1.2 Record area	176
9.1.3 File connector	176
9.1.4 Open mode	177
9.1.5 Sharing file connectors	177
9.1.6 Fixed file attributes	177
9.1.7 Organization	177
9.1.7.1 Sequential	177
9.1.7.2 Relative	178
9.1.7.3 Indexed	178
9.1.8 Access modes	178
9.1.8.1 Sequential access mode	179
9.1.8.2 Random access mode	179
9.1.8.3 Dynamic access mode	179
9.1.9 Reel and unit	179
9.1.10 Current volume pointer	179
9.1.11 File position indicator	179

9.1.12 I-O status	180
9.1.12.1 Successful completion	181
9.1.12.2 Implementor-defined successful completion	181
9.1.12.3 At end condition with unsuccessful completion	182
9.1.12.4 Invalid key condition with unsuccessful completion	182
9.1.12.5 Permanent error condition with unsuccessful completion	182
9.1.12.6 Logic error condition with unsuccessful completion	183
9.1.12.7 Record operation conflict condition with unsuccessful completion	184
9.1.12.8 File sharing conflict condition with unsuccessful completion	184
9.1.12.9 Implementor-defined condition with unsuccessful completion	184
9.1.13 Invalid key condition	184
9.1.14 Sharing mode	185
9.1.15 Record locking	186
9.1.16 Sort file	187
9.1.17 Merge file	187
9.1.18 Dynamic file assignment	187
9.1.19 Report file	187
9.2 Screens	188
9.2.1 Terminal screen	188
9.2.2 Function keys	188
9.2.3 CRT status	188
9.2.4 Cursor	189
9.2.5 Cursor locator	189
9.2.6 Current screen item	190
9.2.7 Color number	190
9.3 Objects	191
9.3.1 Objects and classes	191
9.3.2 Object references	191
9.3.3 Predefined object references	191
9.3.4 Methods	191
9.3.5 Polymorphism	191
9.3.5.1 Class polymorphism	191
9.3.5.2 Parametric polymorphism	191
9.3.6 Method invocation	192
9.3.7 Method prototypes	196
9.3.8 Conformance and interfaces	196
9.3.8.1 Conformance for object orientation	196
9.3.8.1.1 Interfaces	196
9.3.8.1.2 Conformance between interfaces	196
9.3.8.1.3 Conformance for parameterized classes and parameterized interfaces	199
9.3.9 Class inheritance	199
9.3.10 Interface inheritance	199
9.3.11 Interface implementation	200
9.3.12 Parameterized classes	200
9.3.13 Parameterized interfaces	200
9.3.14 Object life cycle	200

9.3.14.1	Life cycle for factory objects	200
9.3.14.2	Life cycle for instance objects	201
9.4	User-defined functions	201
10	Structured compilation group	202
10.1	General	202
10.2	Compilation units	202
10.3	Source units	202
10.4	Contained source units	203
10.5	Source elements and runtime elements	203
10.6	COBOL compilation group	204
10.7	End markers	209
11	Identification division	210
11.1	General	210
11.2	Identification division structure	210
11.3	CLASS-ID paragraph	211
11.4	FACTORY paragraph	212
11.5	FUNCTION-ID paragraph	213
11.6	INTERFACE-ID paragraph	214
11.7	METHOD-ID paragraph	215
11.8	OBJECT paragraph	217
11.9	OPTIONS paragraph	218
11.9.1	General format	218
11.9.2	Syntax rules	218
11.9.3	General Rules	218
11.9.4	ARITHMETIC clause	218
11.9.5	DEFAULT ROUNDED clause	219
11.9.6	ENTRY-CONVENTION clause	219
11.9.7	FLOAT-BINARY clause	220
11.9.8	FLOAT-DECIMAL clause	220
11.9.9	INTERMEDIATE ROUNDING clause	221
11.10	PROGRAM-ID paragraph	223
12	Environment division	225
12.1	General	225
12.2	Environment division structure	225
12.3	Configuration section	226
12.3.1	General format	226
12.3.2	Syntax rules	226
12.3.3	General rules	226
12.3.4	SOURCE-COMPUTER paragraph	227
12.3.5	OBJECT-COMPUTER paragraph	228
12.3.6	SPECIAL-NAMES paragraph	231
12.3.7	REPOSITORY paragraph	242
12.4	Input-output section	247

12.4.1	General format	247
12.4.2	Syntax rules	247
12.4.3	FILE-CONTROL paragraph	248
12.4.4	File control entry	248
12.4.4.4	ACCESS MODE clause	254
12.4.4.5	ALTERNATE RECORD KEY clause	255
12.4.4.6	COLLATING SEQUENCE clause	256
12.4.4.7	FILE STATUS clause	258
12.4.4.8	LOCK MODE clause	259
12.4.4.9	ORGANIZATION clause	261
12.4.4.10	RECORD DELIMITER clause	262
12.4.4.11	RECORD KEY clause	263
12.4.4.12	RELATIVE KEY clause	264
12.4.4.13	RESERVE clause	265
12.4.4.14	SHARING clause	266
12.4.5	I-O-CONTROL paragraph	267
12.4.6	SAME clause	267
13	Data division	269
13.1	General	269
13.2	Data division structure	269
13.3	Explicit and implicit attributes	269
13.4	File section	270
13.4.1	General format	270
13.4.2	Syntax rules	270
13.4.3	General Rules	270
13.4.4	File description entry	271
13.4.5	Sort-merge file description entry	274
13.5	Working-storage section	275
13.6	Local-storage section	276
13.7	Linkage section	277
13.8	Report section	279
13.8.1	General format	279
13.8.2	Syntax rules	279
13.8.3	Report description entry	279
13.8.4	Report group description entry	279
13.8.5	Report subdivisions	279
13.8.5.1	Physical subdivisions of a report	279
13.8.5.1.1	Pages	279
13.8.5.1.2	Lines	280
13.8.5.1.3	Report Items	280
13.8.5.2	Logical Subdivisions of a Report	280
13.9	Screen section	281
13.10	Constant entry	282
13.11	Record description entry	284
13.12	Type declaration entry	284

13.13	77-level data description entry	284
13.14	Report description entry	284
13.15	Report group description entry	285
13.16	Data description entry	288
13.17	Screen description entry	293
13.18	Data division clauses	297
13.18.1	ALIGNED clause	297
13.18.2	ANY LENGTH clause	298
13.18.3	AUTO clause	299
13.18.4	BACKGROUND-COLOR clause	300
13.18.5	BASED clause	301
13.18.6	BELL clause	302
13.18.7	BLANK clause	303
13.18.8	BLANK WHEN ZERO clause	304
13.18.9	BLINK clause	305
13.18.10	BLOCK CONTAINS clause	306
13.18.11	CLASS clause	307
13.18.12	CODE clause	308
13.18.13	CODE-SET clause	309
13.18.14	COLUMN clause	311
13.18.15	CONSTANT RECORD clause	315
13.18.16	CONTROL clause	316
13.18.17	DEFAULT clause	318
13.18.18	DESTINATION clause	319
13.18.19	DYNAMIC LENGTH clause	320
13.18.20	Entry-name clause	321
13.18.21	ERASE clause	322
13.18.22	EXTERNAL clause	323
13.18.23	FOREGROUND-COLOR clause	324
13.18.24	FORMAT clause	325
13.18.25	FROM clause	327
13.18.26	FULL clause	328
13.18.27	GLOBAL clause	329
13.18.28	GROUP INDICATE clause	330
13.18.29	GROUP-USAGE clause	331
13.18.30	HIGHLIGHT clause	332
13.18.31	INVALID clause	333
13.18.32	JUSTIFIED clause	334
13.18.33	Level-number	335
13.18.34	LINAGE clause	336
13.18.35	LINE clause	338
13.18.36	LOWLIGHT clause	343
13.18.37	NEXT GROUP clause	344
13.18.38	OCCURS clause	346
13.18.39	PAGE clause	352
13.18.40	PICTURE clause	354

13.18.41	PRESENT WHEN clause	369
13.18.42	PROPERTY clause	371
13.18.43	RECORD clause	374
13.18.44	REDEFINES clause	377
13.18.45	RENAMES clause	379
13.18.46	REPORT clause	381
13.18.47	REQUIRED clause	382
13.18.48	REVERSE-VIDEO clause	383
13.18.49	SAME AS clause	384
13.18.50	SECURE clause	386
13.18.51	SELECT WHEN clause	387
13.18.52	SIGN clause	388
13.18.53	SOURCE clause	389
13.18.54	SUM clause	391
13.18.55	SYNCHRONIZED clause	395
13.18.56	TO clause	397
13.18.57	TYPE clause	398
13.18.58	TYPDEF clause	403
13.18.59	UNDERLINE clause	404
13.18.60	USAGE clause	405
13.18.61	USING clause	412
13.18.62	VALIDATE-STATUS clause	413
13.18.63	VALUE clause	415
13.18.64	VARYING clause	422
14	Procedure division	424
14.1	General	424
14.2	Procedure division structure	424
14.3	Declaratives	427
14.4	Procedures	428
14.4.1	Sections	428
14.4.2	Paragraphs	428
14.5	Procedural statements and sentences	428
14.5.1	Conditional phrase	430
14.5.2	Scope of statements	430
14.5.2.1	Explicit scope termination	430
14.5.2.2	Implicit scope termination	430
14.6	Execution	431
14.6.1	Run unit organization	431
14.6.2	State of a function, method, object, or program	432
14.6.2.1	State of a function, method, or program	432
14.6.2.1.1	Active state	432
14.6.2.1.2	Initial and last-used states of data	432
14.6.2.2	Initial state of object data	433
14.6.3	Explicit and implicit transfers of control	433
14.6.4	Item identification	434

14.6.5	Results of runtime element execution	435
14.6.6	Locale identification	435
14.6.7	Sending and receiving operands	436
14.6.8	Alignment and transfer of data into data items	436
14.6.8.1	Fixed-point numeric and fixed-point numeric-edited receiving data items	436
14.6.8.2	Floating-point numeric receiving data items	436
14.6.8.3	Floating-point numeric-edited receiving data items	436
14.6.8.4	Receiving data items of categories alphabetic, alphanumeric, alphanumeric-edited, national, and national edited	437
14.6.8.5	Receiving data items of category boolean	437
14.6.9	Overlapping operands	437
14.6.10	Normal run unit termination	437
14.6.11	Abnormal run unit termination	438
14.6.12	Exception condition handling	438
14.6.12.1	Exception conditions	438
14.6.12.1.1	Normal completion of a declarative procedure	439
14.6.12.1.2	Fatal exception conditions	439
14.6.12.1.3	Non fatal exception conditions	440
14.6.12.1.4	Exception objects	440
14.6.12.1.5	Exception-names and exception conditions	441
14.6.12.2	Incompatible data	446
14.7	Common phrases and features for statements	447
14.7.1	At end condition	447
14.7.2	Invalid key condition	447
14.7.3	ROUNDED phrase	447
14.7.4	SIZE ERROR phrase and size error condition	448
14.7.5	CORRESPONDING phrase	450
14.7.6	Arithmetic statements	451
14.7.7	THROUGH phrase	452
14.7.8	RETRY phrase	453
14.8	Conformance for parameters and returning items	454
14.8.1	Parameters	454
14.8.1.1	Group items	454
14.8.1.2	Elementary items	455
14.8.1.2.1	Elementary items passed by reference	455
14.8.1.2.2	Elementary items passed by content or by value	456
14.8.2	Returning items	457
14.8.2.1	Group items	457
14.8.2.2	Elementary items	458
14.9	Statements	460
14.9.1	ACCEPT statement	460
14.9.2	ADD statement	466
14.9.3	ALLOCATE statement	469
14.9.4	CALL statement	471
14.9.5	CANCEL statement	477
14.9.6	CLOSE statement	479

14.9.7 COMPUTE statement	482
14.9.8 CONTINUE statement	484
14.9.9 DELETE statement	485
14.9.10 DISPLAY statement	487
14.9.11 DIVIDE statement	490
14.9.12 EVALUATE statement	493
14.9.13 EXIT statement	498
14.9.14 FREE statement	502
14.9.15 GENERATE statement	503
14.9.16 GO TO statement	505
14.9.17 GOBACK statement	506
14.9.18 IF statement	508
14.9.19 INITIALIZE statement	510
14.9.20 INITIATE statement	514
14.9.21 INSPECT statement	515
14.9.22 INVOKE statement	521
14.9.23 MERGE statement	525
14.9.24 MOVE statement	530
14.9.25 MULTIPLY statement	537
14.9.26 OPEN statement	539
14.9.27 PERFORM statement	544
14.9.28 RAISE statement	550
14.9.29 READ statement	551
14.9.30 RELEASE statement	558
14.9.31 RESUME statement	560
14.9.32 RETURN statement	561
14.9.33 REWRITE statement	563
14.9.34 SEARCH statement	568
14.9.35 SET statement	573
14.9.36 SORT statement	585
14.9.37 START statement	592
14.9.38 STOP statement	596
14.9.39 STRING statement	597
14.9.40 SUBTRACT statement	600
14.9.41 SUPPRESS statement	603
14.9.42 TERMINATE statement	604
14.9.43 UNLOCK statement	605
14.9.44 UNSTRING statement	606
14.9.45 USE statement	610
14.9.46 VALIDATE statement	615
14.9.47 WRITE statement	619
15 Intrinsic functions	627
15.1 General	627
15.2 Types of functions	627
15.3 Arguments	627

15.3.1	Format arguments to international date and time functions	629
15.3.1.1	Date formats	629
15.3.1.1.1	Calendar date formats	629
15.3.1.1.2	Permissible values for data associated with calendar date formats	629
15.3.1.1.3	Ordinal date formats	630
15.3.1.1.4	Permissible values for data associated with ordinal date formats	630
15.3.1.1.5	Week date formats	630
15.3.1.1.6	Permissible values for data associated with week date formats	630
15.3.1.2	Time formats	630
15.3.1.2.1	Common time formats	631
15.3.1.2.2	Local time formats	632
15.3.1.2.3	UTC time formats	632
15.3.1.2.4	Offset time formats	632
15.3.1.2.5	Permissible values for data associated with offset time formats	633
15.3.1.3	Combined date and time formats	633
15.4	Returned values	633
15.4.1	Numeric and integer functions	633
15.5	Date and time conversion functions	634
15.5.1	Integer date form	634
15.5.2	Standard date form	634
15.5.3	Julian date form	634
15.5.4	Standard numeric time form	635
15.6	Summary of functions	635
15.7	ABS function	642
15.8	ACOS function	643
15.9	ANNUITY function	644
15.10	ASIN function	645
15.11	ATAN function	646
15.12	BOOLEAN-OF-INTEGERS function	647
15.13	BYTE-LENGTH function	648
15.14	CHAR function	649
15.15	CHAR-NATIONAL function	650
15.16	COMBINED-DATETIME function	651
15.17	COS function	652
15.18	CURRENT-DATE function	653
15.19	DATE-OF-INTEGERS function	654
15.20	DATE-TO-YYYYMMDD function	655
15.21	DAY-OF-INTEGERS function	656
15.22	DAY-TO-YYYYDDD function	657
15.23	DISPLAY-OF function	658
15.24	E function	659
15.25	EXCEPTION-FILE function	660
15.26	EXCEPTION-FILE-N function	661
15.27	EXCEPTION-LOCATION function	662
15.28	EXCEPTION-LOCATION-N function	663
15.29	EXCEPTION-STATEMENT function	664

15.30 EXCEPTION-STATUS function	665
15.31 EXP function	666
15.32 EXP10 function	667
15.33 FACTORIAL function	668
15.34 FORMATTED-CURRENT-DATE function	669
15.35 FORMATTED-DATE function	670
15.36 FORMATTED-DATETIME function	671
15.37 FORMATTED-TIME function	672
15.38 FRACTION-PART function	673
15.39 HIGHEST-ALGEBRAIC function	674
15.40 INTEGER function	676
15.41 INTEGER-OF-BOOLEAN function	677
15.42 INTEGER-OF-DATE function	678
15.43 INTEGER-OF-DAY function	679
15.44 INTEGER-OF-FORMATTED-DATE function	680
15.45 INTEGER-PART function	681
15.46 LENGTH function	682
15.47 LOCALE-COMPARE function	684
15.48 LOCALE-DATE function	685
15.49 LOCALE-TIME function	686
15.50 LOCALE-TIME-FROM-SECONDS function	687
15.51 LOG function	688
15.52 LOG10 function	689
15.53 LOWER-CASE function	690
15.54 LOWEST-ALGEBRAIC function	691
15.55 MAX function	693
15.56 MEAN function	694
15.57 MEDIAN function	695
15.58 MIDRANGE function	696
15.59 MIN function	697
15.60 MOD function	698
15.61 NATIONAL-OF function	699
15.62 NUMVAL function	700
15.63 NUMVAL-C function	702
15.64 NUMVAL-F function	705
15.65 ORD function	706
15.66 ORD-MAX function	707
15.67 ORD-MIN function	708
15.68 PI function	709
15.69 PRESENT-VALUE function	710
15.70 RANDOM function	711
15.71 RANGE function	712
15.72 REM function	713
15.73 REVERSE function	714
15.74 SECONDS-FROM-FORMATTED-TIME function	715
15.75 SECONDS-PAST-MIDNIGHT function	716

15.76	SIGN function	717
15.77	SIN function	718
15.78	SQRT function	719
15.79	STANDARD-COMPARE function	720
15.80	STANDARD-DEVIATION function	721
15.81	SUM function	722
15.82	TAN function	723
15.83	TEST-DATE-YYYYMMDD function	724
15.84	TEST-DAY-YYYYDDD function	725
15.85	TEST-FORMATTED-DATETIME function	726
15.86	TEST-NUMVAL function	727
15.87	TEST-NUMVAL-C function	728
15.88	TEST-NUMVAL-F function	729
15.89	TRIM function	730
15.90	UPPER-CASE function	731
15.91	VARIANCE function	732
15.92	WHEN-COMPILED function	733
15.93	YEAR-TO-YYYY function	734
16	Standard classes	735
16.1	General	735
16.2	BASE class	735
16.2.1	New method	735
16.2.2	FactoryObject method	735
A	Language element lists	737
A.1	Implementor-defined language element list	737
A.2	Undefined language element list	751
A.3	Processor-dependent language element list	756
A.4	Optional language element list	759
B	Characters permitted in user-defined words	765
B.1	General	765
B.2	Notation	765
B.3	Repertoire of characters permitted in user-defined words	765
C	Mapping of uppercase letters to lowercase letters	769
C.1	Notations	769
C.2	General case mappings	769
C.3	Additional case mappings	772
D	Concepts	773
D.1	General	773
D.2	Files	773
D.3	Tables and dynamic-length elementary items	782
D.4	Shared memory area	789

D.5	Sharing of storage among data items	790
D.6	Compilation group and run unit organization and communication	792
D.7	Intrinsic function facility	805
D.8	Types	806
D.9	Addresses and pointers	808
D.10	Boolean support and bit manipulation	810
D.11	Character sets	813
D.12	Collating sequences	816
D.13	Culturally-specific, culturally-adaptable, and multilingual applications	819
D.14	External switches	824
D.15	Common exception processing	824
D.16	Rounding	826
D.17	Forms of arithmetic	828
D.18	Object oriented concepts	837
D.19	Report writer	855
D.20	Structured constant	862
D.21	Validate facility	863
D.22	Conditional expressions	866
D.23	Examples of the execution of the INSPECT statement	870
D.24	Examples of the execution of the PERFORM statement	873
D.25	Example of free-form reference format	877
D.26	Conditional compilation	878
D.27	CALL-CONVENTION directive	879
D.28	ENTRY-CONVENTION clause	879
D.29	Date and time handling	879
D.30	Alternatives to HIGHEST-ALGEBRAIC and LOWEST-ALGEBRAIC FUNCTIONS	885
E	Substantive changes list	886
E.1	General	886
E.2	Substantive changes potentially affecting existing programs	886
E.3	Substantive changes not affecting existing programs	894
F	Archaic and obsolete language element lists	899
F.1	Archaic language elements	899
F.2	Obsolete language elements	900
G	Known errors in the standard	901
G.1	Rationale	901
G.2	List of errors	901
	BIBLIOGRAPHY	902
	Index	903

Tables

1 Verbal forms	28
2 COBOL character repertoire	70
3 Class and category relationships for elementary data items	129
4 Combinations of symbols in arithmetic expressions	140
5 Combination of symbols in boolean expressions	150
6 Combinations of conditions, logical operators, and parentheses	165
7 Relationship of alphabet-name to coded character set and collating sequence	237
8 Category and type of editing	362
9 Results of fixed insertion editing	363
10 Results of floating insertion editing	364
11 Format 1 picture symbol order of precedence	367
12 Format 2 picture symbol order of precedence	368
13 Procedural statements	429
14 Exception-names and exception conditions	441
15 Relationship of categories of physical files and the format of the CLOSE statement	479
16 Combination of operands in the EVALUATE statement	495
17 Validity of types of MOVE statements	531
18 Category of figurative constants used in the MOVE statement	535
19 Opening available and unavailable files (file not currently open)	540
20 Opening available shared files that are currently open by another file connector	540
21 Permissible I-O statements by access mode and open mode	541
22 Table of functions	636
D.1 Summary of record lock acquisition and release	781
D.2 Examples of boolean operations	811
D.3 ROUNDED MODE examples	827

Figures

1 Fixed-form reference format	31
D.1 Format 1 SEARCH statement having two WHEN phrases	786
D.2 Compilation group sample structure example	793
D.3 Compilation group and run unit structures	794
D.4 Manager class	842
D.5 Banking hierarchy	843
D.6 Example of page layout	856
D.7 Evaluation of the condition-1 AND condition-2 AND ... condition-n	867
D.8 Evaluation of the condition-1 OR condition-2 OR ... condition-n	867
D.9 Evaluation of condition-1 OR condition-2 AND condition-3	868
D.10 Evaluation of (condition-1 OR NOT condition-2) AND condition-3 AND condition-4	869
D.11 The VARYING phrase of a PERFORM statement with the TEST BEFORE phrase having one condition ...	873
D.12 The VARYING phrase of a PERFORM statement with the TEST BEFORE phrase having two conditions ..	874
D.13 The VARYING phrase of a PERFORM statement with the TEST AFTER phrase having one condition ...	875
D.14 The VARYING phrase of a PERFORM statement with the TEST AFTER phrase having two conditions ...	876

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 1989 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 22, *Programming languages, their environments and system software interfaces*, in collaboration with INCITS PL22.4 - Programming Language COBOL.

This second edition cancels and replaces the first edition (ISO/IEC 1989:2002), which has been technically revised. It also incorporates the Technical Corrigenda ISO/IEC 1989:2002/Cor.1:2006, ISO/IEC 1989:2002/Cor.2:2006 and ISO/IEC 1989:2002/Cor.3:2009.

Introduction

COBOL began as a business programming language, but its present use has spread well beyond that to a general purpose programming language. Significant enhancements in this International Standard include:

- Dynamic-capacity tables
- Dynamic-length elementary items
- Enhanced locale support in functions
- Function pointers
- Increased size limit on alphanumeric, boolean, and national literals
- Parametric polymorphism (also known as method overloading)
- Structured constants
- Support for industry-standard arithmetic rules
- Support for industry-standard date and time formats
- Support for industry-standard floating-point formats
- Support for multiple rounding options

Annexes A, B, and C form a normative part of this International Standard. Annexes D through G are for information only.

Annex D, Concepts, includes an explanation of major features as well as the more complicated prior features and is the suggested starting point for the reading of this document.

A complete list of technical changes is given in Annex E.

The previous COBOL standard was published in 2002. Implementors have provided language extensions in response to the demands of their users. Several changes and extensions have, therefore, been made in this International Standard to prevent further divergence, and to ensure consistency among, and coherence within, various implementations.

Development of the COBOL language began before the invention of formal techniques for specification of programming languages. Hence, the COBOL standard uses its own description techniques, which are described in 5, Description techniques. These techniques involve general formats, which describe the syntax, and natural language.

During the development of this International Standard, great care was taken to minimize changes that would affect existing programs. Most substantive changes that potentially affect existing programs were introduced to resolve ambiguities in the previous COBOL standard. Details of the substantive changes are given in Annex E, Substantive changes list.

This page intentionally left blank.

Information technology — Programming languages, their environments and system software interfaces — Programming language COBOL

1 Scope

This International Standard specifies the syntax and semantics of COBOL. Its purpose is to promote a high degree of machine independence to permit the use of COBOL on a variety of data processing systems.

This International Standard specifies:

- The form of a compilation group written in COBOL.
- The effect of compiling a compilation group.
- The effect of executing run units.
- The elements of the language for which a conforming implementation is required to supply a definition.
- The elements of the language for which meaning is explicitly undefined.
- The elements of the language that are dependent on the capabilities of the processor.

This International Standard does not specify:

- The means whereby a compilation group written in COBOL is compiled into code executable by a processor.
- The time at which method, function, or program runtime modules are linked or bound to an activating statement, except that runtime binding occurs of necessity when the identification of the appropriate program or method is not known at compile time.
- The time at which parameterized classes and interfaces are expanded.
- The mechanism by which locales are defined and made available on a processor.
- The form or content of error, flagging, or warning messages.
- The form and content of listings produced during compilation, if any.
- The form of documentation produced by an implementor of products conforming to this International Standard.
- The sharing of resources other than files among run units.