

Programmikkeeled. APL

Programming languages - APL

EESTI STANDARDI EESSÕNA

NATIONAL FOREWORD

See Eesti standard EVS-EN 28485:1999 sisaldab Euroopa standardi EN 28485:1993 ingliskeelset teksti.	This Estonian standard EVS-EN 28485:1999 consists of the English text of the European standard EN 28485:1993.
Standard on jõustunud sellekohase teate avaldamisega EVS Teatajas.	This standard has been endorsed with a notification published in the official bulletin of the Estonian Centre for Standardisation.
Euroopa standardimisorganisatsioonid on teinud Euroopa standardi rahvuslikele liikmetele kättesaadavaks 16.08.1993.	Date of Availability of the European standard is 16.08.1993.
Standard on kättesaadav Eesti Standardikeskusest.	The standard is available from the Estonian Centre for Standardisation.

Tagasisidet standardi sisu kohta on võimalik edastada, kasutades EVS-i veebilehel asuvat tagasiside vormi või saates e-kirja meiliaadressile standardiosakond@evs.ee.

ICS 35.060

Võtmesõnad: apl, data processing, programming (computers), programming languages,

Standardite reprodutseerimise ja levitamise õigus kuulub Eesti Standardikeskusele

Andmete paljundamine, taastekitamine, kopeerimine, salvestamine elektroonsesse süsteemi või edastamine ükskõik millises vormis või millisel teel ilma Eesti Standardikeskuse kirjaliku loata on keelatud.

Kui Teil on küsimusi standardite autorikaitse kohta, võtke palun ühendust Eesti Standardikeskusega:
Aru 10, 10317 Tallinn, Eesti; www.evs.ee; telefon 605 5050; e-post info@evs.ee

The right to reproduce and distribute standards belongs to the Estonian Centre for Standardisation

No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, without a written permission from the Estonian Centre for Standardisation.

If you have any questions about copyright, please contact Estonian Centre for Standardisation:
Aru 10, 10317 Tallinn, Estonia; www.evs.ee; phone 605 5050; e-mail info@evs.ee

UDC 681.3.04:800.92

Descriptors: Data processing, programming: computers, programming languages, APL

English version

Programming languages - APL (ISO 8485:1989)

Langages de programmation - APL (ISO 8485:1989)

Programmiersprachen - APL (ISO 8485:1989)

This European Standard was approved by CEN on 1993-08-13. CEN members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the Central Secretariat or to any CEN member.

The European Standards exist in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CEN member into its own language and notified to the Central Secretariat has the same status as the official versions.

CEN members are the national standards bodies of Austria, Belgium, Denmark, Finland, France, Germany, Greece, Iceland, Ireland, Italy, Luxembourg, Netherlands, Norway, Portugal, Spain, Sweden, Switzerland and United Kingdom.

CEN

European Committee for Standardization
Comité Européen de Normalisation
Europäisches Komitee für Normung

Central Secretariat: rue de Stassart, 36 B-1050 Brussels

Foreword

On the proposal of the CEN Central Secretariat, the Technical Board has decided to submit the International Standard:

"Programming languages - APL (ISO 8485:1989)"

to the formal vote.

The result of the formal vote was positive.

For the time being, this document exists only in English and in French.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by February 1994, and conflicting national standards shall be withdrawn at the latest by February 1994.

In accordance with the CEN/CENELEC Internal Regulations, the following countries are bound to implement this European Standard:

Austria, Belgium, Denmark, Finland, France, Germany, Greece, Iceland, Ireland, Italy, Luxembourg, Netherlands, Norway, Portugal, Spain, Sweden, Switzerland and United Kingdom.

Endorsement notice

The text of this European Standard is identical to the text of the International Standard ISO 8485:1989 without any modification.

INTERNATIONAL STANDARD

**ISO
8485**

First edition
1989-11-01

Programming languages — APL

Langages de programmation — APL



Reference number
ISO 8485 : 1989 (E)

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

Draft International Standards adopted by the technical committees are circulated to the member bodies for approval before their acceptance as International Standards by the ISO Council. They are approved in accordance with ISO procedures requiring at least 75 % approval by the member bodies voting.

International Standard ISO 8485 was prepared by Technical Committee ISO/TC 97, *Information processing systems*.

Users should note that all International Standards undergo revision from time to time and that any reference made herein to any other International Standard implies its latest edition, unless otherwise stated.

Annexes A and B are for information only.

CONTENTS

0	Introduction	3
1	Scope and Field of Application	5
2	References	7
3	Form of the Standard	9
3.1	Form of Definitions	9
3.2	Named Arrays in Examples	10
3.3	Notes	10
3.4	Cross-References	10
3.5	General Definitions	11
4	Compliance	13
4.1	Conforming Implementations	13
4.1.1	Required Behaviour for Conforming Implementations	13
4.1.2	Required Documentation for Conforming Implementations	14
4.1.2.1	Documentation of Optional-Facilities	14
4.1.2.2	Documentation of Implementation-Defined-Facilities	14
4.1.2.3	Consistent Extensions	14
4.2	Conforming Programs	15
4.2.1	Required Behaviour for Conforming Programs	15
4.2.2	Required Documentation for Conforming Programs	15
5	Definitions	17
5.1	Characters	17
5.2	Numbers	19
5.2.1	Elementary Operations	19
5.2.2	Number Constants	20
5.2.3	Subsets of the Set of Numbers	21
5.2.4	Implementation Algorithms	23
5.2.5	Defined Operations	24
5.3	Objects	26
5.3.1	Lists	26
5.3.2	Arrays	27
5.3.3	Defined-Functions	29
5.3.4	Tokens	30
5.3.4.1	Metaclasses	32
5.3.4.2	Index-List	33
5.3.5	Symbols	33
5.3.6	Contexts	34
5.3.7	Workspaces	34
5.3.8	Sessions	35
5.3.9	Shared-Variables	36
5.3.10	Systems	37
5.4	Evaluation Sequences	39
5.4.1	Evaluation Sequence Phrases	39
5.4.2	Diagrams	41
5.5	Other Terms	42

6 Syntax and Evaluation		43
6.1 Introduction		43
6.1.1 Evaluate-Line		43
6.1.2 Character-Diagrams		45
6.1.3 Evaluate-Statement		52
6.1.4 Bind-Token-Class		53
6.1.5 Literal-Conversion		54
6.1.6 Statement-Analysis Token-Diagrams		55
6.2 Reduce-Statement		60
6.3 The Phrase Evaluators		63
6.3.1 Diagrams		64
6.3.2 Remove-Parentheses		65
6.3.3 Evaluate-Niladic-Function		65
6.3.4 Evaluate-Monadic-Function		66
6.3.5 Evaluate-Monadic-Operator		68
6.3.6 Evaluate-Dyadic-Function		69
6.3.7 Evaluate-Dyadic-Operator		71
6.3.8 Evaluate-Indexed-Reference		72
6.3.9 Evaluate-Assignment		73
6.3.10 Evaluate-Indexed-Assignment		74
6.3.11 Evaluate-Variable		75
6.3.12 Build-Index-List		76
6.3.13 Process-End-of-Statement		77
6.4 The Form Table		78
7 Scalar Functions		83
7.1 Monadic Scalar Functions		84
7.1.1 Conjugate	$Z \leftarrow + B$	84
7.1.2 Negative	$Z \leftarrow - B$	84
7.1.3 Signum	$Z \leftarrow \times B$	85
7.1.4 Reciprocal	$Z \leftarrow \div B$	85
7.1.5 Floor	$Z \leftarrow \lfloor B$	86
7.1.6 Ceiling	$Z \leftarrow \lceil B$	86
7.1.7 Exponential	$Z \leftarrow * B$	87
7.1.8 Natural Logarithm	$Z \leftarrow \otimes B$	87
7.1.9 Magnitude	$Z \leftarrow B$	88
7.1.10 Factorial	$Z \leftarrow ! B$	88
7.1.11 Pi times	$Z \leftarrow \circ B$	89
7.1.12 Not	$Z \leftarrow \sim B$	89
7.2 Dyadic Scalar Functions		90
7.2.1 Plus	$Z \leftarrow A + B$	91
7.2.2 Minus	$Z \leftarrow A - B$	91
7.2.3 Times	$Z \leftarrow A \times B$	92
7.2.4 Divide	$Z \leftarrow A \div B$	92
7.2.5 Maximum	$Z \leftarrow A \lceil B$	93
7.2.6 Minimum	$Z \leftarrow A \lfloor B$	93
7.2.7 Power	$Z \leftarrow A * B$	94
7.2.8 Logarithm	$Z \leftarrow A \otimes B$	94
7.2.9 Residue	$Z \leftarrow A B$	95
7.2.10 Binomial	$Z \leftarrow A ! B$	96
7.2.11 Circular Functions	$Z \leftarrow A \circ B$	97
7.2.12 And	$Z \leftarrow A \wedge B$	99
7.2.13 Or	$Z \leftarrow A \vee B$	99
7.2.14 Nand	$Z \leftarrow A \nabla B$	100
7.2.15 Nor	$Z \leftarrow A \nabla B$	100
7.2.16 Equal	$Z \leftarrow A = B$	101
7.2.17 Less than	$Z \leftarrow A < B$	103
7.2.18 Less than or equal to	$Z \leftarrow A \leq B$	103
7.2.19 Not equal	$Z \leftarrow A \neq B$	104
7.2.20 Greater than or equal to	$Z \leftarrow A \geq B$	104
7.2.21 Greater than	$Z \leftarrow A > B$	105

8 Structural Primitive Functions		107
8.1 Introduction		107
8.2 Monadic Structural Primitive Functions		107
8.2.1 Ravel	$Z \leftarrow , B$	107
8.2.2 Shape	$Z \leftarrow \rho B$	108
8.2.3 Index Generator	$Z \leftarrow \iota B$	109
8.3 Dyadic Structural Primitive Functions		110
8.3.1 Reshape	$Z \leftarrow A \rho B$	110
8.3.2 Join	$Z \leftarrow A , B$	111
9 Operators		113
9.1 Introduction		113
9.2 Monadic Operators		114
9.2.1 Reduction	$Z \leftarrow f/ B$	114
9.2.1 Reduction	$Z \leftarrow f/[K] B$	114
9.2.1 Reduction	$Z \leftarrow f\neq B$	114
9.2.1 Reduction	$Z \leftarrow f\neq[K] B$	114
9.2.2 Scan	$Z \leftarrow f\backslash B$	116
9.2.2 Scan	$Z \leftarrow f\backslash[K] B$	116
9.2.2 Scan	$Z \leftarrow f\backslash B$	116
9.2.2 Scan	$Z \leftarrow f\backslash[K] B$	116
9.3 Dyadic Operators		117
9.3.1 Outer Product	$Z \leftarrow A \circ . f B$	117
9.3.2 Inner Product	$Z \leftarrow A f . g B$	118
10 Mixed Functions		119
10.1 Monadic Mixed Functions		119
10.1.1 Roll	$Z \leftarrow ? B$	119
10.1.2 Grade Up	$Z \leftarrow \Delta B$	121
10.1.3 Grade Down	$Z \leftarrow \Psi B$	122
10.1.4 Reverse	$Z \leftarrow \Phi B$	123
10.1.4 Reverse	$Z \leftarrow \Theta B$	123
10.1.4 Reverse	$Z \leftarrow \Phi[K] B$	123
10.1.4 Reverse	$Z \leftarrow \Theta[K] B$	123
10.1.5 Monadic Transpose	$Z \leftarrow \mathfrak{B} B$	124
10.1.6 Matrix Inverse	$Z \leftarrow \mathbb{B} B$	125
10.1.7 Execute	$Z \leftarrow \mathfrak{E} B$	126
10.2 Dyadic Mixed Functions		127
10.2.1 Join Along an Axis	$Z \leftarrow A , [K] B$	127
10.2.2 Index of	$Z \leftarrow A \iota B$	130
10.2.3 Member of	$Z \leftarrow A \in B$	131
10.2.4 Deal	$Z \leftarrow A ? B$	132
10.2.5 Compress	$Z \leftarrow A / B$	133
10.2.5 Compress	$Z \leftarrow A \neq B$	133
10.2.5 Compress	$Z \leftarrow A / [K] B$	133
10.2.5 Compress	$Z \leftarrow A \neq [K] B$	133
10.2.6 Expand	$Z \leftarrow A \backslash B$	135
10.2.6 Expand	$Z \leftarrow A \backslash B$	135
10.2.6 Expand	$Z \leftarrow A \backslash [K] B$	135
10.2.6 Expand	$Z \leftarrow A \backslash [K] B$	135
10.2.7 Rotate	$Z \leftarrow A \Phi B$	137
10.2.7 Rotate	$Z \leftarrow A \Theta B$	137
10.2.7 Rotate	$Z \leftarrow A \Phi [K] B$	137
10.2.7 Rotate	$Z \leftarrow A \Theta [K] B$	137
10.2.8 Base Value	$Z \leftarrow A \perp B$	139
10.2.9 Representation	$Z \leftarrow A \tau B$	140
10.2.10 Dyadic Transpose	$Z \leftarrow A \mathfrak{B} B$	142
10.2.11 Take	$Z \leftarrow A \uparrow B$	144
10.2.12 Drop	$Z \leftarrow A \downarrow B$	145
10.2.13 Matrix Divide	$Z \leftarrow A \mathbb{B} B$	146
10.2.14 Indexed Reference	$Z \leftarrow A [I]$	147
10.2.15 Indexed Assignment	$Z \leftarrow V [I] \leftarrow B$	148

11 System Functions		151
11.1 Introduction		151
11.2 Definition		151
11.3 Diagram		151
11.4 Niladic System Functions		152
11.4.1 Time Stamp	$Z \leftarrow \square TS$	152
11.4.2 Atomic Vector	$Z \leftarrow \square AV$	153
11.4.3 Line Counter	$Z \leftarrow \square LC$	153
11.5 Monadic System Functions		154
11.5.1 Delay	$Z \leftarrow \square DL \ B$	154
11.5.2 Name Class	$Z \leftarrow \square NC \ B$	155
11.5.3 Expunge	$Z \leftarrow \square EX \ B$	156
11.5.4 Name List	$Z \leftarrow \square NL \ B$	156
11.5.5 Query Stop	$Z \leftarrow \square STOP \ B$	157
11.5.6 Query Trace	$Z \leftarrow \square TRACE \ B$	157
11.6 Dyadic System Functions		158
11.6.1 Name List	$Z \leftarrow A \ \square NL \ B$	158
11.6.2 Set Stop	$Z \leftarrow A \ \square STOP \ B$	158
11.6.3 Set Trace	$Z \leftarrow A \ \square TRACE \ B$	159
12 System Variables		161
12.1 Definitions		161
12.2 Evaluation Sequences		162
12.2.1 Comparison Tolerance	$Z \leftarrow \square CT \leftarrow B$	162
12.2.1 Comparison Tolerance	$Z \leftarrow \square CT$	162
12.2.2 Random Link	$Z \leftarrow \square RL \leftarrow B$	163
12.2.2 Random Link	$Z \leftarrow \square RL$	163
12.2.3 Print Precision	$Z \leftarrow \square PP \leftarrow B$	164
12.2.3 Print Precision	$Z \leftarrow \square PP$	164
12.2.4 Index Origin	$Z \leftarrow \square IO \leftarrow B$	165
12.2.4 Index Origin	$Z \leftarrow \square IO$	165
12.2.5 Latent Expression	$Z \leftarrow \square LX \leftarrow B$	166
12.2.5 Latent Expression	$Z \leftarrow \square LX$	166
12.2.5 Latent Expression	$Z \leftarrow \square LX[I] \leftarrow B$	166
13 Defined Functions		167
13.1 Introduction		167
13.2 Definitions		168
13.3 Diagrams		171
13.4 Operations		174
13.4.1 Call-Defined-Function	$Z \leftarrow DFN$	174
13.4.1 Call-Defined-Function	$Z \leftarrow DFN \ B$	174
13.4.1 Call-Defined-Function	$Z \leftarrow A \ DFN \ B$	174
13.4.2 Defined-Function-Control		175
13.4.3 Function Fix	$Z \leftarrow \square FX \ B$	176
13.4.4 Character Representation	$Z \leftarrow \square CR \ B$	177
13.5 Function Editing		178
13.5.1 Evaluate-Function-Definition-Request		178
13.5.2 Evaluate-Editing-Request		180
13.5.3 Diagrams		181
14 Shared Variables		183
14.1 Informal Introduction		183
14.2 Definitions		186
14.3 Diagrams		186
14.4 Operations		186
14.4.1 Primary-Name		186
14.4.2 Surrogate-Name		186
14.4.3 Degree-of-Coupling		186
14.4.4 Access-Control-Vector		186
14.4.5 Offer		187
14.4.6 Retract		188
14.4.7 Shared-Variable-Reset		188

14.5 Shared Variable Forms		189
14.5.1 Shared Variable Reference	$Z \leftarrow SHV$	189
14.5.2 Shared Variable Assignment	$Z \leftarrow SHV \leftarrow B$	189
14.5.3 Shared Variable Indexed Assignment	$Z \leftarrow SHV[I] \leftarrow B$	190
14.6 Shared Variable System Functions		191
14.6.1 Shared Variable Access Control Inquiry	$Z \leftarrow \square SVC B$	191
14.6.2 Shared Variable Query	$Z \leftarrow \square SVQ B$	192
14.6.3 Shared Variable Degree of Coupling	$Z \leftarrow \square SVO B$	193
14.6.4 Shared Variable Offer	$Z \leftarrow A \square SVO B$	194
14.6.5 Shared Variable Retraction	$Z \leftarrow \square SVR B$	195
14.6.6 Shared Variable Access Control Set	$Z \leftarrow A \square SVC B$	196
15 Formatting and Numeric Conversion		197
15.1 Introduction		197
15.2 Numeric Conversion		197
15.2.1 Numeric-Input-Conversion		198
15.2.2 Numeric-Output-Conversion		199
15.3 Diagrams		199
15.4 Operations		203
15.4.1 Monadic Format	$Z \leftarrow \Phi B$	203
15.4.2 Dyadic Format	$Z \leftarrow A \Phi B$	206
16 Input and Output		209
16.1 Introduction		209
16.2 Definitions		209
16.2.1 User Facilities		209
16.2.2 Implementation Algorithms		210
16.2.3 Prompts		211
16.3 Diagrams		211
16.4 Operations		212
16.4.1 Immediate-Execution		212
16.4.2 Quad Input	$Z \leftarrow \square$	213
16.4.3 Quote Quad Input	$Z \leftarrow \square$	214
16.4.4 Quad Output	$Z \leftarrow \square \leftarrow B$	214
16.4.5 Quote Quad Output	$Z \leftarrow \square \leftarrow B$	215
17 System Commands		217
17.1 Introduction		217
17.2 Definitions		217
17.3 Diagrams		218
17.4 Operations		219
17.4.1 Evaluate-System-Command		219
17.5 Diagrams and Evaluation Sequences		220
Annex A. The APL Character Set for Workspace Interchange		225
Annex B: The Workspace Interchange Convention		233
B.1 Introduction		233
B.1.1 Purpose of the Convention		233
B.1.2 Organisation of the Convention		233
B.2 Level 1 Description		234
B.2.1 Canonical Representation Vectors		234
B.2.2 Representation of Variables		235
B.2.3 Representation of Functions		236
B.2.4 Representation of Executable Expressions		237
B.3 Level 2 Description		238
B.3.1 Multiple Canonical Representation Vectors		238
B.3.2 Workspace Environment		238
B.3.3 Pseudovariables		239
B.3.4 Interchange Convention Identifier		239
B.3.5 Escape		239
B.3.6 Bits		239
B.3.7 Translate		239
B.3.8 Note		240
B.3.9 Workspace Identifier		240
B.3.10 End of the Stream		240

B.3.11 Inclusion of Nonstandard Material	240
B.3.12 Use of Other Types	241
B.4 Level 3 Description	242
B.4.1 Atoms	242
B.4.2 Registered character sets	242
B.4.3 Escape sequences	242
B.4.4 Registered characters	242
B.4.5 ISO-APL characters	242
B.4.6 Characters	243
B.4.7 The Translate Table	243
B.4.8 Example	244
B.4.9 Escape	244
B.4.10 Bits	244
B.4.11 Translate	244
B.4.12 Example	245
B.4.13 Mapping of Characters to Bits	245
B.4.14 Stage 1 Translation	245
B.4.15 Stage 2 Translation	245
B.4.16 Example	246
B.4.17 Summary	246
B.5 Level 4 Description	247
B.5.1 Physical Media	247
B.5.2 Nine-track Magnetic Tape	247
B.6 Extended Example	248
Bibliography	253
INDEX	255

Programming languages — APL

This document is a preview generated by EVS

This page intentionally left blank

0 INTRODUCTION

APL stands for **A Programming Language**. It is a notation invented by K. E. Iverson in the late 1950s for the description of algorithms, and expanded on and made into the programming system *APL\360* by Iverson and his colleagues Adin Falkoff, Larry Breed, Dick Lathwell, and Roger Moore in the mid-1960s.

Throughout this document

- the term “this standard” is understood to mean “this International Standard”;
- the words “chapter”, “section” and “subsection” are understood to mean “clause”, “subclause” and “sub-subclause”, respectively.

This document is a preview generated by EVS

This page intentionally left blank

1 SCOPE AND FIELD OF APPLICATION

This standard defines the programming language APL and the environment in which APL programs are executed. Its purpose is to facilitate interchange and promote portability of APL programs and programming skills.

This standard specifies the syntax and semantics of APL programs and the characteristics of the environment in which APL programs are executed.

It also specifies requirements for conformance to this standard, including the publication of values and characteristics of implementation properties so that conforming implementations can be meaningfully compared.

This standard does not specify:

- implementation properties that are likely to vary with the particular equipment or operating system used;

- required values for implementation limits such as APL workspace size or numeric precision;

- the data structures used to represent APL objects;

- the facilities available through shared variables.

This document is a preview generated by EVS

This page intentionally left blank

2 REFERENCES

ISO 2375 : 1985, *Data processing — Procedure for registration of escape sequences.*

ISO 2382-15 : 1985, *Data processing — Vocabulary — Part 15: Programming languages.*