

ISO/IEC 13816:2007-10 (E)

Information technology - Programming languages, their environments and system software interfaces - Programming language ISLISP

Contents		Page
1	Scope 1 2 Normative references 1 3 Compliance of ISLisp processors and text 2 4 Terms and definitions 2 5 Notation and conventions 7 6 Lexemes 9 6.1 Separators	10
6.2	Comments	10
7	Textual representation 10 8 Reserved identifiers 11 9 Errors 11 9.1 Classes of error specification	12
9.2	Pervasive error types	12
10	Classes 13 10.1 Metaclasses	13
10.2	Predefined classes	15
10.3	Standard classes	16
10.3.1	Slots	16
10.3.2	Creating instances of classes	16
11	Scope and extent 17 11.1 The lexical principle	17
11.2	Scope of identifiers	17
11.3	Some specific scope rules	18
11.4	Extent	18
12	Forms and evaluation 19 12.1 Forms	19
12.2	Function application forms	20
12.3	Special forms	20
12.4	Defining forms	21
12.5	Macro forms	21
12.6	The evaluation model	21
12.7	Functions	22
12.8	Defining operators	26
13	Predicates 28 13.1 Boolean values	28
13.2	Class predicates	28
13.3	Equality	29
13.4	Logical connectives	31
14	Control structure 33 14.1 Constants	33
14.2	Variables	34
14.3	Dynamic variables	37
14.4	Conditional expressions	38
14.5	Sequencing forms	40
14.6	Iteration	41
14.7	Non-local exits	42
14.7.1	Establishing and invoking non-local exits	42
14.7.2	Assuring data consistency during non-local exits	46
15	Objects 47 15.1 Defining classes	47
15.1.1	Determining the class precedence list	50
15.1.2	Accessing slots	51
15.1.3	Inheritance of slots and slot options	51
15.2	Generic functions	52
15.2.1	Defining generic functions	52

15.2.2	Defining methods for generic functions	54
15.2.2.1	Agreement on parameter specializers and qualifiers	55
15.2.2.2	Congruent lambda-lists for all methods of a generic function	55
15.2.3	Inheritance of methods	55
15.3	Calling generic functions	56
15.3.1	Selecting the applicable methods	56
15.3.2	Sorting the applicable methods	56
15.3.3	Applying methods	57
15.3.3.1	Simple method combination	57
15.3.3.2	Standard method combination	57
15.3.4	Calling more general methods	59
15.4	Object creation and initialization	60
15.4.1	Initialize-object	60
15.5	Class enquiry	61
16	Macros 61 17 Declarations and coercions 63 18 Symbol class 65 18.1 Symbol names	65
18.1.1	Notation for symbols	65
18.1.2	Alphabetic case in symbol names	66
18.1.3	nil and ()	67
18.2	Symbol properties	67
18.3	Unnamed symbols	68
19	Number class 68 19.1 Number class	69
19.2	Float class	77
19.3	Integer class	80
20	Character class 83 21 List class 84 21.1 Cons	84
21.2	Null class	87
21.3	List operations	87
22	Arrays 91 22.1 Array classes	91
22.2	General arrays	92
22.3	Array operations	93
23	Vectors 95 24 String class 96 25 Sequence functions 99 26 Stream class 102 26.1 Streams to files	103
26.2	Other streams	105
27	Input and output 107 27.1 Argument conventions for input functions	107
27.2	Character I/O	107
27.3	Binary I/O	111
28	Files 112 29 Condition system 114 29.1 Conditions	115
29.2	Signaling and handling conditions	115
29.2.1	Operations relating to condition signaling	116
29.2.2	Operations relating to condition handling	117
29.3	Data associated with condition classes	118
29.3.1	Arithmetic errors	118
29.3.2	Domain errors	118
29.3.3	Parse errors	118
29.3.4	Simple errors	119
29.3.5	Stream errors	119
29.3.6	Undefined entity errors	119
29.4	Error identification	120