Contents

1 Overview 1

1.1 Scope 1
1.2 Processor 1
1.3 Inclusions 1
1.4 Exclusions 1
1.5 Conformance 2
1.5.1 Fortran 95 compatibility 3
1.5.2 Fortran 90 compatibility 3
1.5.3 FORTRAN 77 compatibility 3
1.6 Notation used in this standard 4
1.6.1 Informative notes 4
1.6.2 Syntax rules 4
1.6.3 Constraints 5
1.6.4 Assumed syntax rules 6
1.6.5 Syntax conventions and characteristics 6
1.6.6 Text conventions 6
1.7 Deleted and obsolescent features 6
1.7.1 Nature of deleted features 7
1.7.2 Nature of obsolescent features 7
1.8 Normative references 7

2 Fortran terms and concepts 9

2.1 High level syntax 9
2.2 Program unit concepts 11
2.2.1 Program 12
2.2.2 Main program 12
2.2.3 Procedure 12
2.2.4 Module 13
2.3 Execution concepts 13
2.3.1 Executable/nonexecutable statements 13
2.3.2 Statement order 13
2.3.3 The END statement 14
2.3.4 Execution sequence 15
2.4 Data concepts 15
2.4.1 Data type 15
2.4.2 Data value 16
2.4.3 Data entity 16
2.4.4 Scalar 17
2.4.5 Array 17
2.4.6 Pointer 18
2.4.7 Storage 18
2.5 Fundamental terms 18
2.5.1 Name and designator 18
2.5.2 Keyword 19
2.5.3 Declaration 19
2.5.4 Definition 19

2.5.5 Reference . . . 19
2.5.6 Association . . . 19
2.5.7 Intrinsic . . . 20
2.5.8 Operator . . . 20
2.5.9 Sequence . . . 20
2.5.10 Companion processors . . . 20

3 Characters, lexical tokens, and source form . . . 21

3.1 Processor character set . . . 21
3.1.1 Letters . . . 21
3.1.2 Digits . . . 21
3.1.3 Underscore . . . 22
3.1.4 Special characters . . . 22
3.1.5 Other characters . . . 22
3.2 Low-level syntax . . . 22
3.2.1 Names . . . 23
3.2.2 Constants . . . 23
3.2.3 Operators . . . 23
3.2.4 Statement labels . . . 24
3.2.5 Delimiters . . . 24
3.3 Source form . . . 25
3.3.1 Free source form . . . 25
3.3.2 Fixed source form . . . 27
3.4 Including source text . . . 28

4 Data types . . . 29

4.1 The concept of data type . . . 29
4.1.1 Set of values . . . 29
4.1.2 Constants . . . 29
4.1.3 Operations . . . 30
4.2 Type parameters . . . 30
4.3 Relationship of types and values to objects . . . 31
4.4 Intrinsic data types . . . 31
4.4.1 Integer type . . . 32
4.4.2 Real type . . . 33
4.4.3 Complex type . . . 35
4.4.4 Character type . . . 35
4.4.5 Logical type . . . 38
4.5 Derived types . . . 38
4.5.1 Derived-type definition . . . 39
4.5.2 Determination of derived types . . . 50
4.5.3 Extensible types . . . 51
4.5.4 Component order . . . 53
4.5.5 Type parameter order . . . 53
4.5.6 Derived-type values . . . 54
4.5.7 Derived-type specifier . . . 54
4.5.8 Construction of derived-type values . . . 54
4.5.9 Derived-type operations and assignment . . . 56
4.5.10 The finalization process . . . 56
4.5.11 When finalization occurs . . . 56
4.5.12 Entities that are not finalized . . . 57
4.6 Type aliases . . . 57

4.7 Enumerations and enumerators 58
4.8 Construction of array values 60

5 Data object declarations and specifications 63

5.1 Type declaration statements 63
5.1.1 Type specifiers 66
5.1.2 Attributes 69
5.2 Attribute specification statements 77
5.2.1 Accessibility statements 78
5.2.2 ALLOCATABLE statement 78
5.2.3 ASYNCHRONOUS statement 78
5.2.4 BIND statement 78
5.2.5 DATA statement 79
5.2.6 DIMENSION statement 81
5.2.7 INTENT statement 81
5.2.8 OPTIONAL statement 81
5.2.9 PARAMETER statement 82
5.2.10 POINTER statement 82
5.2.11 SAVE statement 82
5.2.12 TARGET statement 83
5.2.13 VALUE statement 83
5.2.14 VOLATILE statement 83
5.3 IMPLICIT statement 83
5.4 NAMELIST statement 86
5.5 Storage association of data objects 87
5.5.1 EQUIVALENCE statement 87
5.5.2 COMMON statement 89

6 Use of data objects 93

6.1 Scalars 93
6.1.1 Substrings 94
6.1.2 Structure components 94
6.1.3 Type parameter inquiry 95
6.2 Arrays 96
6.2.1 Whole arrays 96
6.2.2 Array elements and array sections 96
6.3 Dynamic association 100
6.3.1 ALLOCATE statement 100
6.3.2 NULLIFY statement 104
6.3.3 DEALLOCATE statement 104

7 Expressions and assignment 107

7.1 Expressions 107
7.1.1 Form of an expression 107
7.1.2 Intrinsic operations 111
7.1.3 Defined operations 112
7.1.4 Data type, type parameters, and shape of an expression 112
7.1.5 Conformability rules for elemental operations 114
7.1.6 Specification expression 114
7.1.7 Initialization expression 116
7.1.8 Evaluation of operations 117

7.2 Interpretation of intrinsic operations 122
7.2.1 Numeric intrinsic operations 122
7.2.2 Character intrinsic operation 123
7.2.3 Relational intrinsic operations 123
7.2.4 Logical intrinsic operations 125
7.3 Interpretation of defined operations 125
7.3.1 Unary defined operation 125
7.3.2 Binary defined operation 126
7.4 Precedence of operators 126
7.5 Assignment 129
7.5.1 Assignment statement 129
7.5.2 Pointer assignment 133
7.5.3 Masked array assignment - WHERE 135
7.5.4 FORALL 138

8 Execution control 145

8.1 Executable constructs containing blocks 145
8.1.1 Rules governing blocks 145
8.1.2 IF construct 146
8.1.3 CASE construct 147
8.1.4 SELECT TYPE and ASSOCIATE constructs 150
8.1.5 DO construct 154
8.2 Branching 158
8.2.1 Statement labels 159
8.2.2 GO TO statement 159
8.2.3 Computed GO TO statement 159
8.2.4 Arithmetic IF statement 159
8.3 CONTINUE statement 159
8.4 STOP statement 159

9 Input/output statements 161

9.1 Records 161
9.1.1 Formatted record 161
9.1.2 Unformatted record 162
9.1.3 Endfile record 162
9.2 External files 162
9.2.1 File existence 162
9.2.2 File access 163
9.2.3 File position 165
9.2.4 File storage units 166
9.3 Internal files 167
9.4 File connection 168
9.4.1 Connection modes 169
9.4.2 Unit existence 169
9.4.3 Connection of a file to a unit 169
9.4.4 Preconnection 170
9.4.5 The OPEN statement 170
9.4.6 The CLOSE statement 174
9.5 Data transfer statements 175
9.5.1 Control information list 175
9.5.2 Data transfer input/output list 180
9.5.3 Error, end-of-record, and end-of-file conditions 182

9.5.4 Execution of a data transfer input/output statement 183
9.5.5 Termination of data transfer statements 192
9.6 Waiting on pending data transfer 193
9.6.1 WAIT statement 193
9.6.2 Wait operation 194
9.7 File positioning statements 194
9.7.1 BACKSPACE statement. 195
9.7.2 ENDFILE statement. 195
9.7.3 REWIND statement 195
9.8 File inquiry. 196
9.8.1 Inquiry specifiers 196
9.8.2 Restrictions on inquiry specifiers. 202
9.8.3 Inquire by output list. 202
9.9 Error processing 202
9.9.1 IOSTAT= specifier 202
9.9.2 IOMSG= specifier. 203
9.9.3 ERR= specifier 203
9.9.4 END= specifier. 203
9.9.5 EOR= specifier 203
9.10 Restriction on input/output statements. 204

10 Input/output editing 205

10.1 Explicit format specification methods 205
10.1.1 FORMAT statement. 205
10.1.2 Character format specification 205
10.2 Form of a format item list. 206
10.2.1 Edit descriptors 206
10.2.2 Fields 208
10.3 Interaction between input/output list and format 208
10.4 Positioning by format control 209
10.5 Decimal symbol. 209
10.6 Data edit descriptors. 209
10.6.1 Numeric editing. 210
10.6.2 Logical editing 214
10.6.3 Character editing 214
10.6.4 Generalized editing 215
10.6.5 User-defined derived-type editing 216
10.7 Control edit descriptors 217
10.7.1 Position editing 217
10.7.2 Slash editing 218
10.7.3 Colon editing 218
10.7.4 SS, SP, and S editing. 218
10.7.5 P editing 218
10.7.6 BN and BZ editing 219
10.7.7 RU, RD, RZ, RN, RC, and RP editing 219
10.7.8 DC and DP editing. 220
10.8 Character string edit descriptors. 220
10.9 List-directed formatting 220
10.9.1 List-directed input 221
10.9.2 List-directed output. 223
10.10 Namelist formatting 224
10.10.1 Namelist input 224
10.10.2 Namelist output 228

11 Program units 231

11.1 Main program 231
11.1.1 Main program executable part 232
11.1.2 Main program internal subprograms 232
11.2 External subprograms 232
11.3 Modules 232
11.3.1 Module reference 233
11.3.2 The USE statement and use association 233
11.4 Block data program units 235

12 Procedures 237

12.1 Procedure classifications 237
12.1.1 Procedure classification by reference 237
12.1.2 Procedure classification by means of definition 237
12.2 Characteristics of procedures 238
12.2.1 Characteristics of dummy arguments 238
12.2.2 Characteristics of function results 238
12.3 Procedure interface 238
12.3.1 Implicit and explicit interfaces 239
12.3.2 Specification of the procedure interface 239
12.4 Procedure reference 247
12.4.1 Actual arguments, dummy arguments, and argument association 249
12.4.2 Function reference 257
12.4.3 Subroutine reference 257
12.5 Procedure definition 257
12.5.1 Intrinsic procedure definition 257
12.5.2 Procedures defined by subprograms 258
12.5.3 Definition and invocation of procedures by means other than Fortran 264
12.5.4 Statement function 265
12.6 Pure procedures 265
12.7 Elemental procedures 267
12.7.1 Elemental procedure declaration and interface 267
12.7.2 Elemental function actual arguments and results 268
12.7.3 Elemental subroutine actual arguments 268

13 Intrinsic procedures and modules 271

13.1 Intrinsic functions 271
13.2 Elemental intrinsic procedures 271
13.3 Arguments to intrinsic procedures 271
13.4 Bit manipulation and inquiry procedures 272
13.5 Numeric manipulation and inquiry functions 272
13.6 Array intrinsic functions 273
13.6.1 The shape of array arguments 273
13.6.2 Mask arguments 273
13.7 Standard intrinsic subroutines 274
13.8 Standard generic intrinsic functions 274
13.8.1 Argument presence inquiry function 274
13.8.2 Numeric functions 274
13.8.3 Mathematical functions 275
13.8.4 Character functions 275
13.8.5 Character inquiry function 275

13.8.6 Kind functions 275
13.8.7 Logical function 276
13.8.8 Numeric inquiry functions 276
13.8.9 Bit inquiry function 276
13.8.10 Bit manipulation functions 276
13.8.11 Transfer function 276
13.8.12 Floating-point manipulation functions 276
13.8.13 Vector and matrix multiply functions 277
13.8.14 Array reduction functions. 277
13.8.15 Array inquiry functions 277
13.8.16 Array construction functions 277
13.8.17 Array reshape function 277
13.8.18 Array manipulation functions 277
13.8.19 Array location functions 277
13.8.20 Allocation status inquiry functions 277
13.8.21 Pointer association status functions 278
13.8.22 Type inquiry functions 278
13.8.23 System environment inquiry functions. 278
13.9 Standard intrinsic subroutines. 278
13.10 Specific names for standard intrinsic functions. 278
13.11 Specifications of the standard intrinsic procedures 280
13.12 The ISO_FORTRAN_ENV intrinsic module 331
13.12.1 Standard input/output units 332
13.12.2 Input/output status. 332

14 Exceptions and IEEE arithmetic 333

14.1 Derived data types defined in the modules. 334
14.2 The exceptions 334
14.3 The rounding modes. 336
14.4 Halting 336
14.5 The floating point status. 336
14.6 Exceptional values. 337
14.7 IEEE arithmetic. 337
14.8 Tables of the procedures. 338
14.8.1 Inquiry functions 338
14.8.2 Elemental functions 338
14.8.3 Kind function 339
14.8.4 Elemental subroutines. 339
14.8.5 Nonelemental subroutines 339
14.9 Specifications of the procedures 339
14.10 Examples 352

15 Interoperability with C. 355

15.1 The ISO_C_BINDING intrinsic module 355
15.2 Interoperation between Fortran and C entities 356
15.2.1 Fortran scalar intrinsic entities and C entities 356
15.2.2 Interoperation with C pointer types 358
15.2.3 Interoperability inquiry functions 358
15.2.4 Interoperation with C struct types. 359
15.2.5 Interoperation with C array types 360
15.2.6 Interoperation with C functions. 361
15.2.7 Interoperation with C global variables 362

16 Scope, association, and definition 365

16.1 Scope of names 365
16.1.1 Global entities. 365
16.1.2 Local entities. 366
16.1.3 Statement and construct entities 372
16.2 Scope of labels 373
16.3 Scope of external input/output units 373
16.4 Scope of operators 373
16.5 Scope of the assignment symbol 373
16.6 Scope of derived-type input/output generic specifiers 373
16.7 Association 374
16.7.1 Name association 374
16.7.2 Pointer association 377
16.7.3 Storage association. 379
16.7.4 Inheritance association 381
16.7.5 Establishing associations. 381
16.8 Definition and undefinition of variables 382
16.8.1 Definition of objects and subobjects 382
16.8.2 Variables that are always defined 382
16.8.3 Variables that are initially defined 382
16.8.4 Variables that are initially undefined 383
16.8.5 Events that cause variables to become defined 383
16.8.6 Events that cause variables to become undefined 384
16.8.7 Variable definition context 386

A. Glossary of technical terms 387

B. Decremental features 399

B.1 Deleted features. 399
B.2 Obsolescent features 400
B.2.1 Alternate return 400
B.2.2 Computed GO TO statement 400
B.2.3 Statement functions 400
B.2.4 DATA statements among executables 400
B.2.5 Assumed character length functions 401
B.2.6 Fixed form source 401
B.2.7 CHARACTER* form of CHARACTER declaration 401

C. Extended notes 403

C.1 Section 4 notes 403
C.1.1 Intrinsic and derived data types (4.4, 4.5). 403
C.1.2 Selection of the approximation methods (4.4.2). 404
C.1.3 Extensible types (4.5.3) 404
C.1.4 Pointers (4.5.1) 405
C.1.5 Structure constructors and generic names 406
C.1.6 Final subroutines (4.5.1.9, 4.5.10, 4.5.11, 4.5.12) 407
C.2 Section 5 notes 408
C.2.1 The POINTER attribute (5.1.2.11). 408
C.2.2 The TARGET attribute (5.1.2.13) 409
C.2.3 The VOLATILE attribute (5.1.2.15) 410
C.3 Section 6 notes 410

C.3.1 Structure components (6.1.2) 410
C.3.2 Allocation with dynamic type (6.3.1) 411
C.3.3 Pointer allocation and association 412
C.4 Section 7 notes 413
C.4.1 Character assignment 413
C.4.2 Evaluation of function references 413
C.4.3 Pointers in expressions 413
C.4.4 Pointers on the left side of an assignment 413
C.4.5 An example of a FORALL construct containing a WHERE construct 414
C.4.6 Examples of FORALL statements 415
C.5 Section 8 notes 415
C.5.1 Loop control 415
C.5.2 The CASE construct 415
C.5.3 Additional examples of DO constructs 415
C.5.4 Examples of invalid DO constructs 417
C.6 Section 9 notes 418
C.6.1 External files (9.2). 418
C.6.2 Nonadvancing input/output (9.2.3.1). 419
C.6.3 Asynchronous input/output 420
C.6.4 OPEN statement (9.4.5) 421
C.6.5 Connection properties (9.4.3) 422
C.6.6 CLOSE statement (9.4.6) 423
C.7 Section 10 notes 423
C.7.1 Number of records (10.3, 10.4, 10.7.2) 423
C.7.2 List-directed input (10.9.1) 423
C.8 Section 11 notes 424
C.8.1 Main program and block data program unit (11.1, 11.4). 424
C.8.2 Dependent compilation (11.3) 424
C.8.3 Examples of the use of modules 426
C.9 Section 12 notes 432
C.9.1 Portability problems with external procedures (12.3.2.2) 432
C.9.2 Procedures defined by means other than Fortran (12.5.3) 432
C.9.3 Procedure interfaces (12.3) 432
C.9.4 Argument association and evaluation (12.4.1.2) 433
C.9.5 Pointers and targets as arguments (12.4.1.2) 434
C.9.6 Polymorphic Argument Association (12.4.1.3). 435
C.10 Section 16 notes 435
C.10.1 Examples of host association (16.7.1.3) 435
C.10.2 Generic resolution and dynamic dispatch (16.1.2.4.4). 436
C.11 Array feature notes 437
C.11.1 Summary of features 437
C.11.2 Examples. 438
C.11.3 FORmula TRANslation and array processing 442
C.11.4 Sum of squared residuals 443
C.11.5 Vector norms: infinity-norm and one-norm 443
C.11.6 Matrix norms: infinity-norm and one-norm. 443
C.11.7 Logical queries 443
C.11.8 Parallel computations 444
C.11.9 Example of element-by-element computation 444
C.11.10 Bit manipulation and inquiry procedures. 445
C.12 Section 16 notes 445
C.12.1 Runtime environments 445
C.12.2 Examples of Interoperation between Fortran and C Functions . . . 445
C.12.3 Example of C calling Fortran 447

D. Index 449