

Appendix F

Subject Index

In the index the F90/95 intrinsic attributes, functions, subroutines, statements, etc. are shown in upper-case letters even though Fortran is not case sensitive. The page numbers are cited with the chapter (or appendix) number followed by a period, followed by the pages in that chapter separated by commas. Topics that occur frequently are only cited at their first few uses.

A edit descriptor 4.34,35
ABS intrinsic 4.24 B.1
abstract data type 2.5,7
access restriction 1.23 2.5 3.1
ACCESS specifier
accessor A.1
accuracy of real arithmetic
ACHAR intrinsic 4.33
ACTION specifier
actual argument
actual array argument
actual pointer argument
ADVANCE specifier 3.8 4.49,50
allocatable array 5.3
ALLOCATABLE attribute 5.3 B.12
ALLOCATE
 statement 5.3 B.13
 status 4.29,30 B.13
ALLOCATED intrinsic B.1
allocation statement B.4
allocation status
alphabetic sorting
alternate RETURN statement B.18
ampersand
analysis
.AND. 3.17 4.16,17,48
angle class
APOSTROPHE
apostrophe edit descriptor
argument
 actual
 association
 dummy
 function 4.23
 list
 presence
 subroutine 4.23
arguments 1.15
arithmetic expression
arithmetic IF statement
arithmetic operators 4.4
arithmetic unit
array
 allocatable
 allocation
 assumed-shape
 assumed-size 4.31
 automatic 5.2,3 A.1
 bounds
 conformable
 constant
 constructor 4.24 5.4,10,14 B.4
 deferred-shape
 dimensions B.4,7
 dummy argument 5.3
 element
 explicit-shape
 extent 5.1
 extraction 5.9
 flip 5.9
 initialization 5.2,4,5
 input
 inquiry B.4
 mask 5.10,11
 name in an I/O list
 output
 overflow A.1
 packing B.4
 pointer
 rank 4.31,53
 reduction B.5
 reshape 5.13,14
 shape 5.1
 shifts 5.14
 size 5.1
 subscripts 5.2
 unknown size 4.31
array element order
array of pointers
array processing intrinsics 5.6,7,11
array section
array specification
array variable
array-valued derived-type component
array-valued function 4.31
array-valued literal constant
ASCII
 character 2.1 4.33,35
 collating sequence 4.32
assembly language 1.18
ASSIGN statement B.16,17
assignment operator 1.11,12
assignment statement A.1 B.17
ASSOCIATED intrinsic 4.45
assumed-length dummy argument
assumed-shape array
assumed-size array
asterisk
 format specifier 2.3,4,5
 default input unit
 default output unit
attribute
 ALLOCATABLE B.16
 DIMENSION B.16
 EXTERNAL
 INTENT B.16
 INTRINSIC
 KIND B.16
 object 1.22, 2.4, A.1
 OPTIONAL B.16
 PARAMETER B.16
 POINTER B.16
 PRIVATE B.16
 PUBLIC B.16
 SAVE B.16
 SEQUENCE
 TARGET B.16
attributes 1.23
automatic array 4.48 5.3
automatic character length

B edit descriptor 4.36
back substitution
back-up B.5
BACKSPACE statement 4.29 B.17
base class 7.1 A.2
behaviour 1.23
binary digit 4.29
binary file 5.5
binary number 4.36
binary operator 4.29
binary tree
bit intrinsic functions 4.29 B.5
blank character 4.33
BLANK specifier
BLOCK DATA statement B.16
block IF construct
block WHERE construct
BN edit descriptor
Boolean 2.1
bottom-up design 1.4
bounds
 array B.5
 lower
 scalar B.5
 upper
bubble sort 4.52
bug 1.8
BZ edit descriptor

C language 1.1
C++ language 1.1,11,15,21 2.2
 4.4,5,8,9,10,14,17,19,22
 4.23,27,29,35,38,39,40,41
 4.45,47 5.2,4,9,24,25
call by reference 4.31
call by value 4.31
CALL statement
CASE construct
CASE DEFAULT statement 4.17,18
case expression
case selector 4.17,18
CASE statement 4.17,18
character
 argument
 array
 assignment
 constant 3.8
 control

- data I/O
- expression
- length
- pointer 4.45
- substring
- character edit descriptor 3.8
- character set
 - default 4.32
 - Fortran
- CHARACTER statement 4.3 B.13
- CHARACTER type 2.1 4.31
- characteristics
 - dummy argument
 - result variable
- chemical element 2.4,7
- Circle class 3.2,4,19
- class
 - base
 - defined
 - derived 4.38
 - hierarchies 3.2
- classes 1.18,23 2.8 3.1
- CLOSE statement 4.29
- closing a file
- CMPLX intrinsic 5.7
- collating sequence
- colon edit descriptor
- colon operator 4.7,25 5.8
- column extraction 5.9
- comment
 - fixed source
 - free source
 - ! statement
- comments 1.1,6 4.1
- COMMON block
- COMMON block name
- COMMON statement 4.27 B.16
- comparing character strings 4.32
- comparison of two real
- compiler 1.19 3.6
- COMPLEX statement 4.3 B.13,21
- COMPLEX type 2.1, B.6
- component
 - derived type 2.4
- composition
- computed GO TO B.16,21
- concatenation
 - operator
- condition
 - end-of-file 4.29
 - end-of-record 4.29
 - error
- conditionals 1.6,7,14 4.13
- conformable arrays
- connectivity 5.12
- constant
 - character
 - derived type
 - integer
 - literal
 - named
 - real
- constant expression
- constructor
 - default 1.23
 - intrinsic 2.5 3.2
 - manual 2.8 3.7
 - structure
- constructors 1.23 3.2
- containers 8.1
- CONTAINS statement 2.9 3.1 4.25,43
- continuation 1.11 B.18
- CONTINUE statement B.16
- control characters 4.32,35
- conversion constants
- copies B.6
- count-controlled DO 1.13 4.11
- counting B.6
- CPU time 4.28
- curve-fitting 4.49,50
- CYCLE
 - named 4.20
 - statement 4.9 B.13
- D edit descriptor

- data abstraction 1.23
- data hiding 3.1
- DATA statement B.16,19
- data member 2.9
- data structure
 - defining 4.39
 - initializing 4.39
 - interpretation 4.40
 - nested 4.38
- data types 2.1
- Date class 3.5
- DEALLOCATE
 - statement 5.3 B.13
 - status 4.29
- deallocation
- debugging 1.19,20
- decimal exponent range
- decimal precision
- default
 - accessibility
 - character set
 - constructor 1.23
 - input unit 5.5
 - kind 4.3
 - output unit 5.5
 - precision
 - private accessibility
 - public accessibility
- deferred-shape array
- defined operation 4.31
- DELIM specifier B.16
- delimiter
- dereferencing 4.8
- derived class 7.1
- derived type
 - argument
 - component
 - constant
 - definition in a module
- destructor 3.2
- dimension
 - attribute
- DIMENSION statement 4.25 B.19
- direct access
 - READ statement
 - WRITE statement
- DIRECT specifier B.16
- DO
 - abort 4.10,20
 - construct 4.10
 - cycle
 - forever 4.10
 - loop 2.9
 - named 4.9,20
 - nested 4.19
 - termination
 - until 4.10,20
 - variable B.16
- DO statement 2.9 4.9 B.13
- DO WHILE statement 4.9,16,20 B.19
- documentation 1.21
- DOT_PRODUCT intrinsic 1.14
- DOUBLE PRECISION attribute 2.1 4.3
- DOUBLE PRECISION statement B.16,17
- double precision 2.3
- doubly linked list 8.15
- Drill class 6.1
- dummy argument
- dummy array argument 5.3
- dummy pointer argument
- dynamic binding
- dynamic character 4.31
- dynamic data structure
- dynamically allocated array 5.3
- dynamically allocated memory
- E edit descriptor
- edit descriptor
 - A 4.34,35
 - B 4.36
 - BN
 - BZ
 - D B.20
 - E B.20
 - EN

ES 3.4 B.20
 F B.20
 G B.20
 I 4.36,38 B.20
 L
 O 4.36 B.20
 P
 S
 SP
 SS
 T B.20
 TL B.20
 TR B.20
 X B.20
 Z 4.36 B.20
 / B.20
 :
 ELEMENTAL prefix
 ELSE IF statement 4.9,16
 ELSE statement 3.16 4.9
 ELSE WHERE statement 4.9
 embedded format
 Employee class 7.5,9,12,15
 EN edit descriptor
 encapsulation 3.1
 END DO statement 2.9
 END FUNCTION statement
 END IF statement 3.16 4.9
 END INTERFACE statement 4.37
 END MODULE statement 2.3
 END PROGRAM statement 2.3 4.2
 END SELECT statement
 END statement
 END SUBROUTINE statement
 end-of-file condition 4.29
 end-of-record condition 4.29
 end-of-transmission 4.32
 END= 4.30
 ENDFILE statement 4.29 B.20
 ENTRY statement B.13,20
 EOSHIFT intrinsic 5.11,14
 .EQ., see ==
 equality of two reals
 EQUIVALENCE statement B.16,20
 .EQV. 4.17
 EM specifier
 error
 checking for
 compilation
 condition
 execution
 I/O
 logical
 semantic
 syntactic
 ES edit descriptor
 exception 4.29,30
 exception descriptor
 IOSTAT 4.13,29
 STAT 4.29 5.3
 executable statement
 execution error
 existence B.7
 EXISTS specifier 4.30
 EXIT
 named 4.20
 statement 4.9,25
 explicit
 interface 4.31
 loops 4.9
 explicit-shape array
 exponent range
 exponential fit 4.50
 exponential format
 expression
 arithmetic
 constant
 evaluation
 in an output list
 mixed-mode expression
 expressions 1.12 4.1
 extending an operator
 extent 5.1
 EXTERNAL attribute
 external file 4.13,37,47
 external procedure 4.47
 EXTERNAL statement B.13
 F edit descriptor B.20
 Fibonacci number
 ADT 2.7
 class 2.8
 file
 access
 connection
 creation
 disconnection
 existence
 external
 inquiry
 internal
 position
 FILE specifier
 fill in B.7
 fixed source form
 floating-point numbers
 flow control 1.13 4.1,9
 FMT specifier
 FORALL construct
 FORM specifier
 format
 embedded
 list-directed
 user input
 FORMAT statement 3.4 B.14
 formatted file
 formatted I/O statement
 formatted record
 FORMATTED specifier
 Fortran Character Set
 fraction 4.43,44
 free source form
 function
 elemental
 length
 name
 pure
 reference
 result
 type
 with no arguments
 function actual argument
 function dummy argument
 FUNCTION statement 2.9 4.22
 functions 1.6,15 4.22
 G edit descriptor
 Game of Life 1.4,9,16,20 4.23,25
 gather 5.12,14,15
 Gaussian elimination
 .GE., see >=
 generic
 defined operator
 function 3.2,4
 identifier
 interface 3.4,7,15
 interface block
 name 4.31
 operator
 procedure 4.31
 geometry module
 global variables 1.16 4.27
 Global Position class 6.7
 GO TO statement 4.9,16,19 B.14
 Great Arc class 6.7
 greatest common divisor 3.16 4.49
 .GT., see >
 hash table
 hexadecimal number 4.36
 host association
 host program unit
 host scoping unit
 I edit descriptor 4.36,38 B.20
 IF
 construct 3.16 4.14
 named 4.18
 nested 4.14
 IF statement 3.5 4.15 B.14

IF ELSE	3.16 4.14	DEALLOCATE	5.3
imaginary part	4.8	DIGITS	
implicit declaration	4.3	DOT_PRODUCT	1.14 5.6,7,11
implicit interface	4.25	EOSHIFT	5.11,14
implicit loop	4.13,24 5.5	EPSILON	5.7
IMPLICIT NONE statement	4.3,28 B.14	EXP	4.8 5.7
IMPLICIT statement	4.3	FLOAT	
INCLUDE line	3.6	FLOOR	4.8 5.8
INDEX intrinsic	4.33,36	HUGE	4.7
index array		IACHAR	4.33,35
index bounds		IAND	4.29
infinite loop	4.10	IBCLR	4.29
information hiding	3.1	IBITS	4.29
inheritance	3.1,10 4.27 7.1	IBSET	4.29
initial statement		ICCHAR	4.33
initial value		IEOR	4.29
initialization expression		IMAG	4.8
input device		INDEX	4.32
input editing		INT	5.7,8
input list		IOR	4.29
input record		ISHFT	4.29
input statement		ISHFTC	4.29
inquiry	B.7	KIND	2.3
list-directed	4.13	LBOUND	
input unit		LEN	4.33,36
I/O statement		LEN_TRIM	4.33
INQUIRE statement	4.30 B.14	LGE	4.33
inquire-by-file	4.30	LGTT	4.33
inquire-by-output-list		LLE	4.33
inquire-by-unit	4.30	LLT	4.33
instance		LOG	4.8 5.7
INT intrinsic	5.7,8	LOG10	4.8 5.7
integer		LOGICAL	2.1
argument		MATXUL	5.5,6,7,11
constant		MAXLOC	4.24 5.7,11
division		MAXVAL	4.24 5.7,11
expression		MERGE	5.11
kind		MINLOC	4.24 5.7,11
literal constant		MINVAL	4.24 5.7,11
numbers		MOD	4.6,8
pointer	4.45	MODULO	3.16 4.8
INTEGER type	2.1,9	MVBITS	4.29
INTENT attribute	3.4 4.9,23	NINT	4.8 5.7,8
INTENT statement	B.14	NOT	4.29
interface	1.2,32 3.2 4.30	NULL	
INTERFACE ASSIGNMENT state-		PACK	5.11
ment	3.16 4.44	PRESENT	
B.14		PRODUCT	5.7,11
interface block	3.16 6.4	RANDOM_NUMBER	5.7
interface body	4.31	RANDOM_SEED	5.7
INTERFACE OPERATOR statement	3.16 4.44	REAL	5.7
5.12,13 B.14		REPEAT	4.33 5.11
INTERFACE statement	3.4 4.37 B.14	RESHAPE	5.4,7,11
internal file	4.35	SCAN	4.33
internal procedure	4.24,28	SELECTED_INT_KIND	2.1
internal variable		SELECTED_REAL_KIND	2.1,3
INTRINSIC attribute	B.22	SHAPE	5.7
intrinsic constructor	4.43	SIGN	4.24 5.7
intrinsic data type	4.4	SIN	1.13 4.8 5.7
intrinsic procedures and calls		SINH	4.8 5.7
ABS	4.8,24 5.7	SIZE	4.23 5.7,13
ACHAR	4.33,35	SPREAD	5.11
ACOS	4.8 5.7	SQRT	4.6,8 5.7
ADJUSTL		SUM	1.14 4.23 5.6,11
AIMAG	5.7	TAN	5.7
AINT	4.8 5.7,8	TANH	4.8 5.8
ALL	5.7,11	TINY	5.8
ALLOCATE	5.3	TRANSFER	4.29 5.11
ALLOCATED	5.3	TRANSPOSE	5.6,8,11
ANINT	5.7,8	TRIM	4.33
ANY	5.7,11	UBOUND	
ASSOCIATED	4.45,46	UNPACK	5.11
ASIN	4.8 5.7	VERIFY	4.33
ATAN	4.8 5.7	WHERE	4.25
ATAN2	1.15 4.8 5.7	INTRINSIC statement	B.14,22
BIT_SIZE	4.29	inverse of a matrix	4.48
BTEST	4.29	IOLength specifier	B.14
CEILING	4.8 5.7,8	IOSTAT specifier	B.13
CHAR	4.32	iteration count	
CMPLX	5.7	iterative methods	
CONJG	4.8 5.7	iterator	
COS	4.8 5.7	keyword	
COSH	4.8 5.7	argument	
COUNT	5.7,11		
CSHIFT	5.11,14		

KIND intrinsic 2.2,3
 kind
 default
 inquiry
 selector
 type B.8,13
 kind type parameter
 of an expression

L edit descriptor
 label 4.16
 latitude 6.7
 LBOUND intrinsic
 .LE., see <=
 leading blanks
 least squares fit 4.49,52
 LEN intrinsic 4.33
 LEN_TRIM intrinsic 4.34
 length
 of a character argument
 of a character variable 4.35
 specification
 lexical comparison intrinsic 4.33
 LGE intrinsic 4.33
 LGT intrinsic 4.33
 library function 1.19
 line
 continuation 1.11 B.18
 maximum length of
 multiple statements on 4.48
 linked list
 circular
 double 8.15
 pointer 4.45,47
 single 8.10
 list-directed
 data value termination on input
 format specifier
 formatting
 input
 output
 PRINT statement
 READ statement
 literal constant
 array-valued
 LLE intrinsic 4.33
 LLT intrinsic 4.33
 local variable
 location in an array B.8
 logical
 expression 1.13
 function
 literal constant
 value
 variable
 logical IF statement
 LOGICAL intrinsic
 logical operator 4.17
 LOGICAL statement B.14
 LOGICAL type 2.1 4.17 B.8
 longitude 6.7
 loop
 abort 4.19
 cycle 4.19
 counter 4.10,11
 implied 4.12,24
 indexed 1.13 4.11,48
 infinite 4.10
 named 4.9
 nested 4.12,13
 post-test 4.10,20
 pre-test 4.10,20
 variable 1.13
 loops 1.6,7,13 4.12
 loss of precision
 lower bound
 lower case letters 4.36
 .LT., see <

main program
 maintainability
 Manager class 3.1 7.5,8,10,12,13,15
 mantissa
 many-one array section
 masked array assignment

masks 4.13 5.10,12,25 B.1,2,3,9
 massively parallel computer
 mathematical constants 2.3
 Matlab 4.4,5,7,8,9,10,12,14
 4.22,23,27 5.2,4,5,6,7
 5.9,9,24,25
 matrix
 addition 5.18,24
 column 5.11,16
 diagonal 5.17
 factorization 5.21
 inverse 5.12,20,24
 multiplication 5.12,19,24
 operations 5.12
 partition 5.17
 row 5.16
 shifts 5.14
 square 5.16
 symmetric 5.17
 transpose 5.12,17
 mean 4.23
 memory
 allocation
 deallocation
 leak 8.9
 message
 mixed kind expressions
 mixed-mode expression
 model number
 bit
 integer
 real
 modular design 1.5,6
 modular program development 1.2
 module 1.18 2.3 3.1 4.27,28
 module procedure
 MODULE PROCEDURE state-
 ment 3.4,16 5.12,13 7.1
 MODULE statement 2.9 7.1 B.14
 module variable 2.3 4.28
 multiple inheritance 7.1

name length
 NAME specifier
 named
 DO construct 4.9
 IF construct 4.18
 CASE construct
 SELECT construct 4.18
 named constant
 NAMED specifier
 NAMELIST statement B.16,22
 .NE., see /=
 negative iteration count
 negative subscript value
 .NEQV. 4.17
 nested
 data structures 4.38
 DO loops 4.13
 IF blocks 4.14
 implied loops 4.12
 scoping unit
 Newton-Raphson method 1.13,25
 NEXTREC specifier
 NML specifier B.23
 node
 non-advancing I/O 3.18
 non-advancing READ statement
 non-counting DO loop
 non-default
 character
 character set
 complex number
 integer
 kind
 logical
 real
 .NOT. 4.17,49
 NULL 4.46
 null character 4.33
 NULLIFY statement 4.46 B.14,22
 NUMBER specifier
 number B.9
 numerical sorting

O edit descriptor 4.36
 object 3.1
 object-oriented
 analysis 1.21,23
 design 1.21,24
 languages 1.21,24
 programming 1.1,21 3.1 6.1
 obsolescent statements 2.1,2 B.17
 octal number 4.36
 ONLY qualifier 4.27 7.1 B.25
 OPEN statement B.14,25
 OPENED specifier 4.30
 operator
 definition 4.43
 overloading 3.14 4.43
 operators
 arithmetic
 binary
 concatenation
 unary
 optional argument 2.9 3.7
 OPTIONAL attribute 2.9 3.7 4.30,31
 OPTIONAL statement B.14,22
 .OR. 3.7 4.49
 order of evaluation
 effect of parentheses
 output
 output device
 output editing
 output format
 output list
 expression in
 output statement
 list-directed
 overflow
 overloading 3.14,16 4.43

 P edit descriptor
 PACK intrinsic procedure
 PAD specifier B.16
 padding 5.14
 parallel computers 3.19
 PARAMETER attribute 2.3
 PARAMETER statement B.14,22
 parameterized constants
 parameterized data types
 parameterized real variables
 parameterized variables
 parentheses to set order
 pass by reference 4.7,31
 pass by value 4.8,31
 PAUSE statement B.16,22
 peripheral device
 Person class 3.6 4.47
 physical constants
 pointer
 allocation
 array
 assignment 4.45
 association status
 component of derived type
 deallocation
 dummy argument
 input and output
 inquiry B.10
 linked list
 nullification
 return value 4.31
 target 4.45
 variable
 POINTER attribute 4.31,45
 POINTER statement B.14
 pointer-valued function
 polymorphism 3.1 7.1
 POSITION specifier
 post-condition
 power law fit 4.50
 precedence orders 4.5
 precision 2.2
 pre-condition 3.7 4.30
 PRESENT intrinsic
 PRINT statement
 PRIVATE attribute
 private components 2.6
 private members 2.6

PRIVATE qualifier
 PRIVATE statement 2.9 B.14
 procedure
 argument
 interface
 intrinsic
 libraries
 size
 PRODUCT intrinsic
 Professor class 7.3
 program
 design 1.3,11
 errors
 name
 structure
 testing 1.18
 PROGRAM statement 2.3 4.2 B.14
 program unit
 external routine
 function
 internal routine
 main
 module
 subroutine
 projectile 4.49
 prototype 1.6,23 4.30
 pseudocode 1.5,8,9,14,16 4.10
 PUBLIC attribute
 public attributes 2.6
 public member
 PUBLIC statement 2.9 B.14
 PURE prefix

 quadratic equation 1.8

 random access
 random number
 range
 range of a DO loop
 rank
 rational number
 arithmetic 3.15
 class 3.14
 derived type 3.16
 READ specifier
 READ statement 2.4 4.12,29 B.14
 read-only file
 READWRITE specifier
 real
 argument
 arithmetic operation
 constant
 DO variable
 exponential literal constant
 expression
 literal constant
 number
 part (of a complex number)
 variable
 REAL intrinsic
 REAL statement 2.3 B.14
 REAL type 2.1
 REC specifier B.16
 RECL specifier B.16
 Rectangle class 3.2,3,4,19
 recursion
 recursive
 algorithms 4.49
 bisection method
 data structure
 function B.15,21,24
 procedure
 process
 reference
 subroutine B.15,24,25
 RECURSIVE qualifier 3.16 B.21,24
 reduction B.11
 relational expression
 relational operator 4.4
 rename modifier 7.1,4 B.25
 renaming of module entities
 repeat count
 repeatable edit descriptor
 repeated format
 RESHAPE intrinsic

reshaping an array B.11
 restricting access to module
 restrictions on a logical IF
 restrictions on a DO loop
 result length
 RESULT specification 2.9 3.4 B.13
 result variable
 return from a procedure
 RETURN statement 4.9 B.15,16
 returns 1.15
 reverse order B.11
 REWIND statement 4.29 B.15,24
 root
 roots of a quadratic equation
 round-off error
 row extraction 5.9

 S edit descriptor
 SAVE attribute
 SAVE statement B.15,24
 in a module 4.27
 scalar
 conformable with an array
 scalar product of two vectors
 scalar variable
 scale factor
 scatter 5.12,14
 scope
 scoping unit
 scratch file
 SELECT CASE statement 4.9,17,18 B.15
 SELECTED_INT_KIND intrinsic
 SELECTED_REAL_KIND intrinsic
 semantic error
 SEQUENCE attribute B.24
 sequential access
 sequential file
 sequential I/O statement
 SEQUENTIAL specifier
 shape
 side effects 8.9
 SHAPE intrinsic procedure
 shifts B.11
 simultaneous linear equations
 singly linked list 8.10
 size
 SIZE intrinsic 5.13
 SIZE specifier
 solution of linear equations
 sorting, bubble
 source form
 fixed form
 free form
 SP edit descriptor
 space character
 SPACING intrinsic B.17
 sparse matrix
 Sparse Vector class
 specification
 specification expression
 specification statement
 specifier
 ACCESS B.16
 ACTION
 ADVANCE B.16,23
 APPEND B.16
 ASIS B.16
 BLANK B.16
 DELIM B.16
 DIRECT B.16
 END B.16,22
 EOR B.16
 ERR B.13
 EXIST B.21
 FILE B.21
 FMT B.14,16,23
 FORM B.16
 FORMATTED B.16
 IOLENGTH B.14,21
 IOSTAT B.13,16,21
 NAME B.14,21
 NAMED
 NEXTREC
 NEW B.16
 NML B.23,25
 NONE B.16
 NULL B.16
 NUMBER
 OLD B.16
 OPENED B.21
 PAD B.16
 POSITION B.16
 QUOTE B.16
 READ B.16
 READWRITE B.16
 REC B.16,23,25
 RECL B.16
 REPLACE B.16
 REWIND B.16
 SEARCH B.16
 SEQUENTIAL B.16
 SIZE B.16
 STAT B.13
 STATUS B.16
 UNFORMATTED B.16
 UNIT B.14,16
 UNKNOWN B.16
 WRITE B.16
 ZERO B.16
 SS edit descriptor
 statement entity
 statement function B.16
 statement label 4.16
 statement order
 statements 1.1,2,11
 STATUS specifier
 STOP statement 3.7 4.9 5.3 B.15
 storage unit
 character
 numeric
 string 2.1
 strong typing 4.3,28
 structure
 components 4.38
 constructor 4.43
 structured programming 1.15
 Student class 3.7,11
 subprograms 4.21
 subprogram interface 3.16 4.31 6.4
 subroutine
 actual argument
 arguments
 call
 dummy argument
 interface
 library
 name
 result
 SUBROUTINE statement 2.9 4.22 B.15
 subscript 2.9 5.1
 subscript expression
 subscript triplet 4.7 5.2,8
 substring
 SUM intrinsic 1.14
 swap 9.1
 syntactic error 1.1,19
 SYSTEM_CLOCK call 4.28

 tabs 4.50
 T edit descriptor
 target B.11
 TARGET attribute 4.45
 TARGET statement B.15
 template 9.1
 TL edit descriptor
 top-down design
 TR edit descriptor
 TRANSPOSE intrinsic 5.4
 TRIM intrinsic 4.33
 trailing blanks
 tree-structured data 4.45 8.1
 truncation error
 type conversion intrinsic
 TYPE declaration statement 2.4,9
 type parameter
 TYPE statement 2.4,9 B.15

 UBOUND intrinsic
 unary operator
 undefined array

undefined pointer status
 underflow
 unformatted
 file
 I/O statements
 record
 UNFORMATTED specifier
 unit number
 unit specifier
 until construct
 upper bound
 upper case letters 4.36
 US Military Standard 4.29
 USE association
 USE statement 2.3 3.2,4,16 4.27
 7.1 B.15

variable
 character
 declaration
 initial value
 internal
 local
 name 4.2
 variables 1.12 4.1
 Vector class 5.25 B.12
 vector subscript 5.12

WHERE construct 5.10 B.25
 WHERE statement 3.4 4.9 5.9 B.14,25
 while loop 4.20,222
 WRITE specifier 3.16 B.14
 WRITE statement

X edit descriptor

Z edit descriptor
 zero-sized array

! comment
 continuation marker
 namelist data initiator

()
 implied loop bounds
 subscript bounds

(/ /) array constructor 5.2

** exponentiation 4.8
 + overloaded 3.16
 - overloaded
 /

 edit descriptor
 list-directed data terminator
 namelist data terminator
 value separator

// concatenation 4.32

/=
 not equal
 overloaded

:
 edit descriptor
 subscript triplet

:: attribute terminator 2.3
 ; statement terminator

<
 less than
 overloaded

<=
 less than or equal to
 overloaded

=
 assignment
 overloaded 3.16

==
 equal to
 overloaded 3.16

>=
 greater than or equal to
 overloaded

=> rename option

>
 greater than
 overloaded

- character in a name