

CONVEX FORTRAN

Master Index

Document No. 720-003230-000

Second Edition, Rev. 1

March 1990

CONVEX Computer Corporation
Richardson, Texas USA

CONVEX FORTRAN Master Index
Order No. DSW-033
Second Edition, Rev. 1

© 1987, 1988, 1989, 1990 CONVEX Computer Corporation
All rights reserved.

This document is copyrighted. This document may not, in whole or part, be copied, duplicated, reproduced, translated, stored electronically, or reduced to machine-readable form without prior written consent from CONVEX Computer Corporation.

Although the material contained herein has been carefully reviewed, CONVEX Computer Corporation (CONVEX) does not warrant it to be free of errors or omissions. CONVEX reserves the right to make corrections, updates, revisions or changes to the information contained herein. CONVEX does not warrant the material described herein to be free of patent infringement.

UNLESS PROVIDED OTHERWISE IN WRITING WITH CONVEX COMPUTER CORPORATION (CONVEX), THE PROGRAM DESCRIBED HEREIN IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. SOME STATES DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES. THE ABOVE EXCLUSION MAY NOT BE APPLICABLE TO ALL PURCHASERS BECAUSE WARRANTY RIGHTS CAN VARY FROM STATE TO STATE. IN NO EVENT WILL CONVEX BE LIABLE TO ANYONE FOR SPECIAL, COLLATERAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES, INCLUDING ANY LOST PROFITS OR LOST SAVINGS, ARISING OUT OF THE USE OR INABILITY TO USE THIS PROGRAM. CONVEX WILL NOT BE LIABLE EVEN IF IT HAS BEEN NOTIFIED OF THE POSSIBILITY OF SUCH DAMAGE BY THE PURCHASER OR ANY THIRD PARTY.

CONVEX and the CONVEX logo ("C") are registered trademarks of CONVEX Computer Corporation.
ConvexOS is a trademark of CONVEX Computer Corporation.
COVUE is a registered trademark of CONVEX Computer Corporation. COVUE consists of the following products: COVUEbatch, COVUEbinary, COVUEedt, COVUElib, COVUenet, and COVUEshell.
Cray is a registered trademark of Cray Research, Inc.
Sun FORTRAN is a trademark of Sun Microsystems, Inc.
UNIX is a registered trademark of AT&T Bell Laboratories.
VAX and VMS are trademarks of Digital Equipment Corporation.

Printed in the United States of America

Index

\$ descriptor lrm-8-19
: descriptor lrm-8-20

-72 option ug-1-8

A

-a1 option ug-1-6
absolute value ug-C-14
ACCEPT statement lrm-7-1, lrm-7-13
access, direct ug-2-5
ACCESS keyword lrm-7-21
access modes ug-2-5
access, sequential ug-2-5
accessing files lrm-7-4
actual arguments lrm-9-1
adb debugger ug-4-3
adjustable arrays lrm-9-2
alternate return arguments lrm-9-5, lrm-9-9
analyzer, performance ug-1-6
AND, bitwise ug-C-11
ANSI standard formatting lrm-1-4
apostrophe descriptor lrm-8-5
arc cosine ug-C-6
arc sine ug-C-6
arc tangent ug-C-6
arc tangent (degree) ug-C-7
arc tangent, two arguments ug-C-6
arc tangent, two arguments (degree) ug-C-7
argument packets ug-5-1, ug-5-7
argument pointer ug-5-1
argument-passing mechanisms ug-5-2
arguments lrm-9-8, lrm-9-9, ug-C-1
arguments, alternate return lrm-9-9
arguments, character lrm-9-3
arguments, dummy lrm-9-9
arithmetic expression, data types lrm-2-9
arithmetic expressions lrm-2-9
arithmetic IF statement lrm-6-3
arithmetic operators lrm-2-9, ug-C-18
array arguments lrm-9-2
array declaration lrm-2-8
array storage lrm-2-8
array subscripts lrm-2-8
array table ug-1-12
arrays lrm-2-7
ASCII character set lrm-E-1
assembly-language debugger ug-4-3
ASSIGN LOCK directive lrm-C-3
ASSIGN statement lrm-5-3
assigned GOTO statement lrm-6-2
assignment statement lrm-5-1
ASSOCIATEVARIABLE keyword lrm-7-23
assumed-length character argument lrm-3-4
assumed-size arrays lrm-9-3
auxiliary input/output statements lrm-7-20
auxiliary I/O operations ug-C-21

B

B descriptors lrm-8-14
-B option ug-1-8
BACKSPACE statement lrm-7-32, lrm-7-33
BEGIN ORDER directive lrm-C-4

BEGIN SECTION directive lrm-C-4
BEGIN_TASKS directive lrm-C-5
Binary Datafile Format Conversions lrm-7-34
bitwise AND ug-C-11
bitwise circular shift ug-C-13
bitwise clear ug-C-13
bitwise complement ug-C-11
bitwise extract ug-C-12
bitwise OR ug-C-11
bitwise set ug-C-12
bitwise shift ug-C-11
bitwise test ug-C-12
bitwise XOR ug-C-11
blank common storage lrm-3-1
BLANK keyword lrm-7-23, lrm-F-2
BLOCK DATA statement lrm-10-1
BLOCK DATA statement in subprograms lrm-9-9
block IF statement lrm-6-4
blockdata subprogram lrm-10-1
block-level profiling ug-4-4
BLOCKSIZE keyword lrm-7-23
BN descriptor lrm-8-14
bprof profiler ug-1-6, ug-4-7
built-in functions lrm-9-6
BYTE lrm-2-1, ug-A-1
BZ descriptor lrm-8-14

C

C interface ug-1-13
-c option ug-1-4
CALL statement lrm-9-10
calling conventions ug-5-1, ug-C-1
calling utility routines ug-6-1
carriage-control characters lrm-8-29
CARRIAGECONTROL keyword lrm-7-24
-cfc option lrm-G-1, ug-1-2
CHAR function ug-3-5
character arguments lrm-9-3
character constants lrm-2-6, ug-3-1
character conversions lrm-5-1
character data ug-3-1
character descriptor A lrm-8-4
character equivalence lrm-3-6
character expressions lrm-2-11
CHARACTER FUNCTION statement lrm-9-9
character I/O ug-3-4
character library functions ug-3-4
character relationals ug-C-13
character representation ug-A-5
character set lrm-1-1
character strings, concatenating ug-3-3
character substrings lrm-2-12, ug-3-3
CHARACTER type-declaration statements lrm-3-4
character variables, declaring ug-3-2
character-per-column formatting lrm-1-3
characters, carriage-control lrm-8-29
character-valued function ug-5-1
circular shift, bitwise ug-C-13
clear, bitwise ug-C-13

- CLOSE statement lrm-7-29
 - code-generation options ug-1-4
 - colon descriptor lrm-8-20
 - comma field separator lrm-8-21
 - comment line lrm-1-2
 - common block storage lrm-3-1
 - common blocks lrm-3-7
 - common logarithm ug-C-4
 - COMMON statement lrm-2-7, lrm-3-1
 - comparison operators ug-C-18
 - compiler directives lrm-1-4, lrm-C-1
 - compiler features ug-1-1
 - compiler messages ug-1-9, ug-B-1
 - compiler options ug-1-2
 - compiling FORTRAN 66 programs lrm-F-1
 - compiling programs ug-1-2
 - complement, bitwise ug-C-11
 - COMPLEX lrm-2-1
 - complex conjugate ug-C-9
 - complex descriptor lrm-8-4
 - complex, imaginary part ug-C-9
 - complex programmed operators ug-C-17
 - complex, real part ug-C-9
 - complex representation ug-A-4
 - complex values ug-C-2
 - COMPLEX*16 lrm-2-1
 - COMPLEX*16 constants lrm-2-4
 - COMPLEX*8 lrm-2-1, lrm-2-10
 - COMPLEX*8 constants lrm-2-4
 - computed GOTO statement lrm-6-2
 - concatenation lrm-2-11
 - conjugate, complex ug-C-9
 - constant expressions lrm-2-13
 - constants lrm-2-3
 - consultant package ug-4-5
 - contact*, aborting the report J-3, J-6, ug-E-3, ug-E-6
 - contact*, editing the report J-6, ug-E-6
 - contact*, ending a response J-3, ug-E-3
 - contact*, ending the report J-6, ug-E-6
 - .contact* file, skipping first prompt by using J-3, ug-E-3
 - contact*, including files in the report J-6, ug-E-6
 - contact*, invoking J-1, J-4, ug-E-1, ug-E-4
 - contact*, prerequisites J-1, ug-E-1
 - contact*, prompts J-4, ug-E-4
 - contact*, reporting problems J-1, ug-E-1
 - contact*, restrictions on tilde-escape sequences J-6, ug-E-6
 - contact*, reviewing the report J-6, ug-E-6
 - contact*, skipping first prompt by using *.contact* file J-3, ug-E-3
 - contact*, step-by-step discussion of prompts J-4, ug-E-4
 - contact*, submitting *dead.report* file J-3, ug-E-3
 - contact*, submitting the report J-6, ug-E-6
 - contact*, suspending the report J-3, ug-E-3
 - contact*, tilde-escape sequences J-4, ug-E-4
 - contact*, tips on using J-2, ug-E-2
 - continuation indicator lrm-1-3
 - continuation line lrm-1-3
 - CONTINUE statement lrm-6-10
 - control statement lrm-6-1
 - conversion feature, when to use lrm-7-34
 - conversion, fix-to-float ug-C-8
 - conversion, float-to-fix ug-C-7, ug-C-8
 - conversion, integer ug-C-15
 - conversion of data types lrm-2-2
 - conversion, REAL*16 to REAL*4 ug-C-15
 - conversion, REAL*16 to REAL*8 ug-C-15
 - conversion, REAL*4 to REAL*16 ug-C-15
 - conversion, REAL*8 to REAL*16 ug-C-15
 - conversion restrictions lrm-7-35
 - conversion, user-defined lrm-7-36
 - Conversion Using a Shell Variable lrm-7-39
 - Conversion Using OPEN Statement lrm-7-35
 - conversions, IEEE/native ug-C-13
 - CONVEX consultant ug-4-5
 - CONVEX FORTRAN ug-1-1
 - CONVEX math library ug-C-1
 - CONVEX symbolic debugger ug-4-5
 - ConvexOs Utilities ug-6-1
 - cosine ug-C-5
 - cosine, hyperbolic ug-C-7
 - COVUEshell lrm-1-7, lrm-7-21
 - Cray compatibility lrm-G-1
 - cross-reference generator ug-1-7, ug-4-1
 - cs* option ug-1-6
 - csd* debugger ug-4-5
 - CXpa* ug-1-6, ug-4-3
- ## D
- D descriptor lrm-8-11
 - D indicator lrm-1-4
 - Data File Formats ug-2-6
 - data format lrm-7-3, lrm-7-4, lrm-7-40
 - data representation ug-A-1
 - data representations ug-5-6
 - DATA statement lrm-4-1
 - data types lrm-2-1
 - data types, arithmetic expression lrm-2-9
 - data types, conversion lrm-2-2
 - data types, equivalenced lrm-3-5
 - dataformat lrm-7-26
 - data-type length specifiers lrm-3-3
 - date* utility ug-6-4
 - db* option ug-1-6
 - dc* option lrm-1-4, ug-1-6
 - dead.report* file, submitting J-3, ug-E-3
 - dead.report* file, using -*r* option to submit J-3, ug-E-3
 - debug statements lrm-1-4
 - debugger, assembly-language ug-4-3
 - debugger, symbolic ug-4-5
 - debugging options ug-1-6
 - DECODE statement lrm-7-19
 - default descriptor values lrm-8-21
 - DEFAULTFILE keyword lrm-7-24
 - descriptors lrm-8-3
 - development tools, program ug-4-1

-dfc option lrm-7-34
-dfc option ug-1-2
 diagnostic messages ug-1-9
 difference, positive ug-C-10
 dimension declarator lrm-2-8
 DIMENSION statement lrm-3-4
 direct access lrm-7-4, ug-2-5
 direct-access file ug-2-6
 direct-access WRITE statements lrm-7-15
 directives, compiler lrm-1-4, lrm-C-1
 DISPOSE keyword lrm-7-25
 DO list, implied lrm-4-2
 DO loops, extended range lrm-6-8
 DO statement lrm-6-8, lrm-6-9
 DO statements lrm-6-6
 DO WHILE statement lrm-6-9
 dollar sign descriptor lrm-8-19
 DOUBLE PRECISION lrm-2-1
 double-precision constants lrm-2-3
-ds option ug-1-3
 dummy argument lrm-9-1
 dummy argument, NAMELIST statement
 lrm-3-9
 dummy arguments lrm-9-1, lrm-9-8, lrm-9-9
 dummy arguments, arrays lrm-9-2
 dump, postmortem ug-4-6

E

E descriptor lrm-8-11
 edit descriptors lrm-8-3
 ENCODE statement lrm-7-18
 END DO statement lrm-6-9
 END ORDER directive lrm-C-4
 END SECTION directive lrm-C-4
 END specifier ug-7-1
 END statement lrm-6-11
 END statement in subprograms lrm-9-9
 ENDFILE record lrm-7-2
 ENDFILE statement lrm-7-32
 end-of-file specifier lrm-7-9
 END_TASKS directive lrm-C-5
 entry points, intrinsics ug-C-2
 entry points, I/O list element transmission
 ug-C-21
 entry points, I/O list initialization ug-C-20
 ENTRY statement lrm-9-11
-ep option ug-1-3
 EQUIVALENCE statement lrm-2-7, lrm-3-5
 equivalencing arrays lrm-3-5
 ERR keyword lrm-7-25
 ERR specifier ug-7-1
 error messages ug-1-9
 error reporting J-1, ug-E-1
 error specifier lrm-7-8
 error utility ug-4-7
error utility ug-B-1
 error-processing utilities ug-7-4
 errors, runtime ug-7-1
errsns utility ug-6-5
errtrap utility ug-7-5
 examples ug-5-7

exception, runtime ug-7-1
 exceptions ug-7-3
 executable program lrm-1-1
 executing programs ug-1-9
exit utility ug-6-5
 exponential ug-C-4
 exponentiation programmed operators
 ug-C-16
 expressions lrm-2-9
 expressions, constant lrm-2-13
 external files ug-2-5
 external READ statements lrm-7-10
 EXTERNAL statement lrm-3-10
 extract, bitwise ug-C-12

F

F descriptor lrm-8-10
-F66 option ug-1-2
fc command line ug-1-2
-fi option ug-1-4
 field declaration lrm-H-5
 field descriptors lrm-8-3
 field separators, external lrm-8-21
 FILE keyword lrm-7-25
 file positioning lrm-8-3
 file type ug-2-5
 file-naming conventions ug-1-1
 file-positioning statements lrm-7-32
 files lrm-7-2
 files, accessing lrm-7-4
 files, FORTRAN source ug-1-1
 FIND statement lrm-7-20
 fix-to-float conversion ug-C-8
 floating-point data representation ug-A-2
 floating-point, IEEE ug-A-4
 floating-point, native ug-A-3
 floating-point representation, IEEE ug-1-1,
 ug-1-4
 float-to-fix conversion ug-C-7, ug-C-8
-fn option ug-1-4
for\$ prefix ug-C-2
 FORCE_PARALLEL directive lrm-C-6
 FORCE_PARALLEL_EXT directive lrm-C-5
 FORCE_VECTOR directive lrm-C-6
 FORM keyword lrm-7-26
 format code separators lrm-F-3
 FORMAT control lrm-8-2
 format conversions lrm-7-34
 format specifications lrm-8-1
 format specifier lrm-7-7
 FORMAT statement lrm-8-1
 formats, variable lrm-8-22
 formatted I/O ug-2-4
 formatted records lrm-7-2
 formatting, list-directed lrm-8-23
 FORTRAN 66 compatibility lrm-F-1
 FORTRAN 77 formatting lrm-1-4
 FORTRAN argument packets ug-5-1
 FORTRAN character set lrm-E-1
 FORTRAN intrinsic library ug-C-1
 FORTRAN I/O library ug-C-19

fpp lrm-1-7, lrm-B-1
-fpp ug-1-8
 FREE LOCK directive lrm-C-3
 FUNCTION statement lrm-9-9
 function subprograms lrm-9-9
 function-naming convention ug-C-2
 functions lrm-9-5
 functions, intrinsic ug-C-4
fxref program ug-1-7, ug-4-1

G

G descriptor lrm-8-12
 generic and intrinsic functions, table lrm-A-1
gerror utility ug-7-7
 global scalar optimization ug-1-14
 GOTO statement lrm-6-1
gprof profiler ug-1-6, ug-4-7

H

H descriptor lrm-8-5
 hexadecimal constants lrm-2-5
 Hollerith constants lrm-2-6
 Hollerith representation ug-A-5
 hyperbolic cosine ug-C-7
 hyperbolic sine ug-C-7
 hyperbolic tangent ug-C-7

I

I descriptor lrm-8-7
-i option ug-1-4
-i2 option ug-1-4
-i4 option ug-1-4
-i8 option ug-1-4
 ICHAR function ug-3-4
idate utility ug-6-4
 IEEE 754 standard ug-D-1
 IEEE compatibility ug-D-1
 IEEE floating-point ug-A-4
 IEEE floating-point representation ug-1-1,
 ug-1-4
 IEEE/native conversions ug-C-13
ierrno utility ug-7-7
 IF statements lrm-6-3
 IF THEN statement lrm-6-4
-il option ug-1-3, ug-1-15
 imaginary part of complex ug-C-9
 IMPLICIT NONE statement lrm-3-2
 IMPLICIT statement lrm-3-2
 implied-DO list lrm-4-2, lrm-7-6
 #include statement lrm-1-7
 INCLUDE statement lrm-1-7
 #include statement lrm-B-1
 INCLUDE statement lrm-H-1
 INDEX function ug-3-5
 index, string ug-C-13
 Inf operand lrm-8-10, lrm-8-12, lrm-8-13,
 ug-A-4
 initial line lrm-1-3
 inline substitution ug-1-15
 input, list-directed lrm-8-23
 input/output ug-2-1

input/output lists lrm-7-5
 input/output statements lrm-7-1
 input/output statements, auxiliary lrm-7-20
 INQUIRE statement lrm-7-30
 integer constants lrm-2-3
 integer conversion ug-C-15
 integer descriptor lrm-8-4
 integer, nearest ug-C-14
 integer part of real ug-C-8
 integer representation ug-A-1
 INTEGER*1 lrm-2-1
 INTEGER*2 lrm-2-1
 INTEGER*4 lrm-2-1
 INTEGER*8 lrm-2-1
 internal files lrm-7-3, ug-2-5, ug-2-6
 internal READ statements lrm-7-12
 internal WRITE statements lrm-7-16
 intrinsic functions lrm-9-5, ug-C-4
 intrinsic library ug-C-1
 INTRINSIC statement lrm-3-10
 invoking the compiler ug-1-2
 I/O error processing ug-7-1
 I/O forms ug-2-4
 I/O list element transmission ug-C-20
 I/O list initialization ug-C-20
 I/O list termination ug-C-21
 I/O operation ug-C-19
 I/O runtime naming convention ug-C-20
 I/O statement format lrm-7-5
 I/O statements lrm-7-1
 IOSTAT keywCONVEX Guide to Vector and
 Parallel Optimizationord lrm-7-27
 IOSTAT specifier ug-7-2
-is option ug-1-3, ug-1-15
-iw option ug-1-7

K

keywords, OPEN statement lrm-7-21

L

L descriptor lrm-8-6
 LEN function ug-3-5
 length, string ug-C-13
 lexical comparison functions ug-3-6
libF77 library ug-C-2
libI77.a library ug-C-19
 libraries, runtime ug-C-1
 library, intrinsic ug-C-1
 library, math ug-C-1
 limits, system lrm-D-1
-link ug-1-8
 list, implied-DO lrm-4-2
 list-directed character input lrm-8-24
 list-directed complex input lrm-8-24
 list-directed formatting lrm-8-23
 list-directed input lrm-8-23
 list-directed I/O ug-2-4
 list-directed, null value lrm-8-24
 list-directed output lrm-8-27
 list-directed, slashes lrm-8-24
 listing options ug-1-7

LNBLNK function ug-3-5
 loader ug-1-9
 loading programs ug-1-9
 %LOC function lrm-9-7, ug-5-4
 local scalar optimization ug-1-14
 logarithm, common ug-C-4
 logarithm, natural ug-C-4
 logical constants lrm-2-6
 logical descriptor lrm-8-4
 logical elements lrm-2-11
 logical entities lrm-2-10
 logical expressions lrm-2-10
 logical IF statement lrm-6-3
 logical names ug-2-1
 logical operator .XOR. lrm-2-11
 logical records ug-2-6
 logical representation ug-A-1
 LOGICAL*1 lrm-2-1
 LOGICAL*2 lrm-2-1, lrm-2-10
 LOGICAL*4 lrm-2-1
 LOGICAL*8 lrm-2-1
 longjmp utility ug-7-4
 loop replication ug-1-4, ug-1-15
 loop table ug-1-10
 loop-level profiling ug-4-4
 -LST option ug-1-7
 -LSTI option ug-1-7

M

machine-dependent optimization ug-1-15
 main program lrm-1-1, lrm-3-8
 MAP keyword lrm-H-5
 math library ug-C-1
 maximum ug-C-9
 MAXREC keyword lrm-7-27
 MAX_TRIPS directive lrm-C-6
 message options ug-1-7
 messages ug-1-9, ug-B-1
 minimum ug-C-9
 mth\$ prefix ug-C-2
 multiple statements lrm-1-4
 mvbits utility ug-6-5

N

-na option ug-1-7
 NAME keyword lrm-7-25
 namelist input lrm-8-25
 namelist specifier lrm-7-9
 NAMELIST statement lrm-3-9
 namelist-directed formatting lrm-8-25
 namelist-directed I/O ug-2-6
 namelist-directed output lrm-8-28
 names, logical ug-2-1
 names, runtime ug-C-2
 NaN operand lrm-8-10, lrm-8-12, lrm-8-13,
 ug-A-4
 native floating-point ug-A-3
 natural logarithm ug-C-4
 nearest integer ug-C-14
 nested block IF statement lrm-6-6
 nested DO loops lrm-6-7

NEXT_TASK directive lrm-C-5
 NML keyword lrm-7-9
 -no option ug-1-3
 non-FORTRAN-to-FORTRAN calling
 sequence ug-5-5
 nonrepeatable descriptors lrm-8-2
 NO_PARALLEL directive lrm-C-8
 NO_RECURRENCE directive lrm-C-7
 NO_SIDE_EFFECTS directive lrm-C-7
 NOSPANBLOCKS keyword lrm-7-27
 NO_VECTOR directive lrm-C-8
 numeric type-declaration statements lrm-3-3
 -nv option ug-1-7
 -nw option ug-1-7

O

O descriptor lrm-8-8
 -O option ug-1-3
 -o option ug-1-8
 -OO option ug-1-14
 -O1 option ug-1-3, ug-1-14
 -O2 option ug-1-3, ug-1-14
 -O3 optimization ug-1-14
 -O3 option ug-1-3
 octal constants lrm-2-4, lrm-2-5
 OPEN statement lrm-7-21, ug-2-2
 OPEN statement conversions lrm-7-35
 OPEN statement keywords lrm-F-2
 operator precedence lrm-2-9
 optimization ug-1-13
 optimization, global scalar ug-1-14
 optimization, local scalar ug-1-14
 optimization, machine-dependent ug-1-15
 optimization report ug-1-7, ug-1-9, ug-1-10
 options, compiler ug-1-2
 OPTIONS statement lrm-1-5
 OR, bitwise ug-C-11
 -or option ug-1-7

P

P descriptor lrm-8-15
 -p option ug-1-6, ug-4-7
 -p8 option ug-1-4
 -pa option ug-1-6
 -pab option ug-1-6
 packets, argument ug-5-1
 -par ug-1-6
 parallelization ug-1-14
 PARAMETER statement lrm-3-7
 PARAMETER statement, alternate lrm-3-8
 PAUSE statement lrm-6-10
 -pb option ug-1-6, ug-4-7
 -pd8 option ug-1-5
 performance analyzer ug-1-6, ug-4-3
 perror utility ug-7-7
 -pg option ug-1-6, ug-4-7
 pmd utility ug-4-6
 pointer, argument ug-5-1
 positive difference ug-C-10
 postmortem dump ug-4-6
 precedence, operator lrm-2-9

preconnection of units ug-2-1
 PREFER_PARALLEL directive lrm-C-8
 PREFER_PARALLEL_EXT directive
 lrm-C-8
 PREFER_VECTOR directive lrm-C-9
 preprocessor lrm-1-7, lrm-B-1
 PRINT statement lrm-7-1, lrm-7-17
 priority, data type lrm-2-10
 problem reporting ug--0
 procedure names ug-5-6
 procedures as dummy arguments lrm-9-4
prof profiler ug-1-6, ug-4-7
 profiler, *bprof* ug-1-6
 profiler, *CXpa* ug-1-6
 profiler, *gprof* ug-1-6
 profiler, *prof* ug-1-6
 profilers ug-4-7
 profiling options ug-1-6
 program development tools ug-4-1
 program, executable lrm-1-1
 program interfaces ug-1-13
 program, main lrm-1-1, lrm-3-8
 PROGRAM statement lrm-1-1, lrm-3-8
 PROGRAM statement in subprograms
 lrm-9-9
 program units lrm-1-1
 PSTRIIP directive lrm-C-9
 -*pw* option ug-1-7

Q

Q descriptor lrm-8-20

R

R descriptor lrm-8-17
 -*r* option ug-1-5
 -*r4* option ug-1-5
 -*r8* option ug-1-5
ran utility ug-6-5
 rank, data type lrm-2-10
 -*re* option ug-1-5
 READ statement lrm-7-1, lrm-7-10
 READ statement, direct lrm-7-12
 READ statement, external lrm-7-10, lrm-7-12
 READ statement, internal lrm-7-12
 READ statement, sequential lrm-7-10
 READONLY keyword lrm-7-27
 REAL lrm-2-1, lrm-5-2
 real constants lrm-2-3
 real data representation ug-A-2
 real descriptor lrm-8-4
 real, integer part ug-C-8
 real part of complex ug-C-9
 REAL*16 lrm-2-1, lrm-2-3, lrm-2-10,
 lrm-5-2, ug-C-2
 REAL*16 data representation ug-A-2
 REAL*16 programmed operators: ug-C-18
 REAL*16 to REAL*4 conversion ug-C-15
 REAL*16 to REAL*8 conversion ug-C-15
 REAL*4 lrm-2-1, lrm-2-3, lrm-5-2
 REAL*4 data representation ug-A-2
 REAL*4 to REAL*16 conversion ug-C-15

REAL*4 to REAL*8 conversion ug-C-15
 REAL*8 lrm-2-1, lrm-2-3, lrm-2-10, lrm-5-2
 REAL*8 data representation ug-A-2
 REAL*8 product of REAL*4 ug-C-9
 REAL*8 to REAL*16 conversion ug-C-15
 REAL*8 to REAL*4 conversion ug-C-15
 RECL keyword lrm-7-27
 RECORD keyword lrm-H-4
 record specifier lrm-7-8
 records lrm-7-1
 records, logical ug-2-6
 RECORDSIZE keyword lrm-7-27
 RECORDTYPE keyword lrm-7-28
 %REF function lrm-9-6, ug-5-4
 registers, scalar ug-C-1
 registers, vector ug-C-1
 relational expressions lrm-2-10
 relationals, character ug-C-13
 remainder ug-C-10
 repeatable descriptors lrm-8-2, lrm-8-3
 replication, loop ug-1-4
 report, optimization ug-1-7, ug-1-10
 reserved operand lrm-8-10, lrm-8-12,
 lrm-8-13, ug-A-3
 Restrictions on Conversions lrm-7-35
 RETURN, alternate lrm-9-10
 RETURN statement lrm-6-11, lrm-9-11,
 lrm-9-12
 RETURN statement in subprograms lrm-9-9
 return values ug-5-6
 REWIND statement lrm-7-32
 RINDEX function ug-3-5
 -*rl* option ug-1-4, ug-1-15
 Rop operand lrm-8-10, lrm-8-12, lrm-8-13,
 ug-A-3
 routine-level profiling ug-4-4
 ROW_WISE directive lrm-C-9
 runtime data items ug-C-19
 runtime error messages ug-1-12, ug-B-2
 runtime errors and exceptions ug-7-1
 runtime formats lrm-8-22
 runtime interface ug-1-13
 runtime libraries ug-C-1
 runtime messages ug-B-1
 runtime names ug-C-2
 runtime prefixes ug-C-2
 runtime stack ug-5-5
 runtime utilities ug-6-1

S

S descriptor lrm-8-16
 -*S* option ug-1-5
 -*sa* option ug-1-2
 SAVE statement lrm-3-10
 -*sc* option ug-1-6
 SCALAR directive lrm-C-10
 scalar registers ug-C-1
 scalar truncation ug-C-2
 scale factor lrm-8-15
secnds utility ug-6-5
 SELECT directive lrm-C-11

- semicolon separator lrm-1-4
 - sequential access lrm-7-4, ug-2-5
 - sequential READ statements lrm-7-10
 - sequential WRITE statement, unformatted lrm-7-15
 - sequential-access file ug-2-6
 - sequential-access WRITE statements lrm-7-14
 - set, bitwise ug-C-12
 - setjmp* utility ug-7-4
 - sfc* option lrm-I-1, ug-1-3
 - shell variable conversions lrm-7-39
 - shell variables lrm-7-3
 - shift, bitwise ug-C-11
 - shift, bitwise circular ug-C-13
 - sign, transfer of ug-C-11
 - signal handling examples ug-7-7
 - signal* utility ug-7-6
 - signals and exceptions ug-7-2
 - sine ug-C-5
 - sine, hyperbolic ug-C-7
 - sl* option ug-1-7
 - slash descriptor lrm-8-20
 - source files ug-1-1
 - SP descriptor lrm-8-16
 - specification statements lrm-3-1
 - specifiers lrm-7-7
 - square root ug-C-4
 - SS descriptor lrm-8-16
 - stack, runtime ug-5-5
 - statement field lrm-1-4
 - statement function reference lrm-9-8
 - statement functions lrm-9-8
 - statement label lrm-1-3
 - statement label assignment lrm-5-3
 - statements, executable lrm-1-2
 - statements, multiple lrm-1-4
 - statements, nonexecutable lrm-1-2
 - STATUS keyword lrm-7-28, lrm-F-3
 - status specifier lrm-7-8
 - STOP statement lrm-6-10
 - storage, array lrm-2-8
 - string index ug-C-13
 - string length ug-C-13
 - string-manipulation programmed operators ug-C-18
 - strip mining, parallel lrm-C-9
 - strip mining, vector lrm-C-13
 - structure declaration lrm-H-4
 - STRUCTURE keyword lrm-H-4
 - SU descriptor lrm-8-16
 - subprogram lrm-1-1
 - subprogram calling conventions ug-5-1
 - subprograms lrm-9-1
 - SUBROUTINE statement in subprograms lrm-9-9
 - subroutine subprograms lrm-9-10
 - Sun FORTRAN compatibility lrm-I-1
 - symbolic debugger ug-4-5
 - symbolic Names lrm-2-1
 - SYNCH_PARALLEL directive lrm-C-11
 - system errors ug-B-2
 - system limits lrm-D-1
 - system utilities ug-6-1
 - system* utility ug-6-3
- T**
- T descriptors lrm-8-18
 - tab character lrm-1-4
 - tab-key formatting lrm-1-4
 - table, array ug-1-12
 - TAC, Technical Assistance Center J-1, ug-E-1
 - tangent ug-C-5
 - tangent, hyperbolic ug-C-7
 - target machine ug-1-5
 - tasking directives lrm-C-5
 - Technical Assistance Center (TAC) J-1, ug-E-1
 - test, bitwise ug-C-12
 - tilde-escape sequences J-4, ug-E-4
 - tilde-escape sequences, restrictions on use J-6, ug-E-6
 - time* utility ug-6-5
 - TL descriptor lrm-8-18
 - tl* option ug-1-8
 - tm* option ug-1-5
 - tools, program development ug-4-1
 - TR descriptor lrm-8-18
 - traceback* utility ug-7-6
 - transfer of sign ug-C-11
 - trouble reports J-1, ug-E-1
 - TYPE keyword lrm-7-28
 - TYPE statement lrm-7-1, lrm-7-17
 - type-declaration statements lrm-3-3
- U**
- unconditional GOTO statement lrm-6-1
 - unformatted I/O ug-2-4
 - unformatted records lrm-7-2
 - UNION keyword lrm-H-5
 - UNIT keyword lrm-7-29
 - unit specifier lrm-7-7
 - units lrm-7-3
 - units, input/output ug-2-1
 - units, program lrm-1-1
 - UNIX-to-UNIX Communication Protocol J-1, ug-E-1
 - UNIX-to-UNIX copy command, *uucp* J-1, ug-E-1
 - UNROLL directive lrm-C-12
 - uo* option ug-1-4
 - ur* option ug-1-4
 - User-Defined Conversions lrm-7-36
 - utilities ug-6-1
 - utilities, ConvexOS ug-6-1
 - utility routines, how to call ug-6-1
 - UUCP, connection to TAC J-1, ug-E-1
 - uucp*, UNIX-to-UNIX copy command J-1, ug-E-1

V

%VAL function lrm-9-6, ug-5-3
values, complex ug-C-2
variable formats lrm-8-22
variables lrm-2-7
VAX FORTRAN compatibility lrm-H-1
VAX FORTRAN records lrm-H-4
VAX-11 utilities ug-6-4
vector mask programmed operators: ug-C-18
vector registers ug-C-1
vectorization ug-1-14
vers command, using to find program version
 number J-2, ug-E-2
-vfc option lrm-H-1, ug-1-3
VMS FORTRAN compatibility lrm-H-1
-vn option ug-1-8
VSTRIP directive lrm-C-12

W

whence command, using to find program path
 name J-2, ug-E-2
which command, using to find program path
 name J-2, ug-E-2
WRITE statement lrm-7-1, lrm-7-14,
 lrm-8-27
WRITE statement, list-directed output for-
 mats lrm-8-27
WRITE statements, direct access lrm-7-15
WRITE statements, internal lrm-7-16

X

X descriptor lrm-8-17, lrm-F-3
XOR, bitwise ug-C-11
-xr option ug-1-7
-xrl option ug-1-7

Z

Z descriptor lrm-8-9