I.P. Sharp Associates

SHARP APL Utility Library Catalogue

THE

SHARP APL

UTILITY LIBRARY

CATALOGUE

22 JANUARY 1985

THIS CATALOGUE CONSISTS OF THREE PARTS:

- AN INDEX OF THE CATEGORIES OF THE UTILITY LIBRARY,
- SUMMARIES OF THE UTILITIES FOUND IN EACH CATEGORY, AND
- DETAILED DESCRIPTIONS OF EACH UTILITY IN THE LIBRARY.

THE CATALOGUE REFLECTS THE STATE OF THE LIBRARY AT THE DATE PRINTED ABOVE.

UTILITIES DESCRIBED IN THIS CATALOGUE MAY BE DEFINED IN THE ACTIVE WORKSPACE BY <UDEFINE> FROM WORKSPACE 2 ULIBRARY. FOR EXAMPLE, TO DEFINE THE UTILITY SAUCE/SQUEEZE, EXECUTE THE FOLLOWING.

```
)COPY 2 ULIBRARY UDEFINE
UDEFINE 'SAUCE/SQUEEZE'
```

REVISION DATE: 1984-11-27

CATEGORY 1 MISCELLANEOUS — CATEGORY 1

DJK/DATATYPE.1	DETERMINES THE INTERNAL DATA TYPE OF AN ARRAY (OR PACKAGE).
DJK/PAUSE.1	<PAUSE> SUSPENDS. THE USER MAY RESUME EXECUTION BY ENTERING 'RESUME'.
MGF/FFIB.1	FAST FIBONACCI FUNCTION; FFIB (⍳7)-⎕IO ↔ 0 1 1 2 3 5 8
MJAB/BSTABLE.2	TABLE OF VALID OVERSTRIKES
MJAB/BSTRANSLATE.1	CHANGES CHARACTER STRING WITH IMBEDDED BACKSPACES USED TO REPRESENT OVERSTIKES INTO TRUE OVERSTRIKES
MJAB/HSPASK.5	PROMPTS USER FOR LOCATION (REMOTE OR TORONTO) TO PRINT. GETS DELIVERY INSTRUCTIONS. SUBMITS HSPREQ.
PCB/DEFAULT.1	LEFT AGRUMENT IS CHARACTER VECTOR CONTAINING NAME OR EXPRESSION.
ROHAN/QSAVE.1	SAVES THE ACTIVE WORKSPACE UNDER A GIVEN NAME.
SAUCE/DESCRIBE.1	AN OVERVIEW OF THE FUNCTIONS IN SAUCE/*.
TS/NODE.1	RETURNS NODE NUMBERS AND LOCATIONS OF (SPECIFIED) NODES OR LOCATIONS ON THE IPSA NETWORK.

CATEGORY 2 STRUCTURAL TRANSFORMATIONS — CATEGORY 2

DBA/RCAT.1	VERY FAST ROWWISE CATENATION OF CHARACTER VECTORS OR MATRICES
DJK/ENDSPOSE.1	MOVES SPECIFIED AXES TO THE END OF THE SHAPE VECTOR.
HUI/PASTE.1	GIVEN TWO VECTORS OF FORMATTED TEXT, WILL PUT THEM TOGETHER SIDE BY SIDE.
MJAB/ENC.1	BREAKS UP SIMPLE TEXT VECTOR INTO ENCLOSED VECTOR OF WORDS.
MJAB/ON.1	CREATES A MATRIX RESULT WITH ALL ROWS OF LEFT ARG BEFORE ALL ROWS OF RIGHT.
MRAB/∆BOX.1	SIMULATION OF APL.68000'S ⎕BOX; MAKE MATRIX FROM VECTOR OR VECTOR FROM MATRIX
MRAB/∆PVM.1	VECTOR TO MATRIX ACCORDING TO PARTITIONING BOOLEAN VECTOR
PCB/CH∆RAVEL.1	REPRESENTS AN ARRAY AS CHARACTER VECTOR, TRAILING BLANKS REMOVED.
PCB/MATRIX.1	ASSURES RESULT IS A MATRIX
PCB/ON.1	JOINS TWO ARRAYS ONE ABOVE THE OTHER, CONVERTING TYPE WHERE THEY ARE MIXED
ROHAN/MAKE∆NL.1	TAKES A CHARACTER-ARRAY ARGUMENT OF NAMES, OF ANY RANK, AND RETURNS A LEFT-JUSTIFIED MATRIX NAMELIST
SAUCE/CATENATEROWS.1	CATENATES ONE VECTOR OR MATRIX UNDER ANOTHER, USING OVERTAKE AS NECESSARY.
SAUCE/CYCLICMESH.1	E.G. 4 CYCLICMESH 2 12⍴'AEI','BFJ','CGK','DHL' ↔ 2 12⍴'ABCDEFGHIJKL'.
SAUCE/LEFTJUSTIFY.1	LEFT JUSTIFIES EACH ROW OF AN ARRAY.
SAUCE/MESH.1	MESHES CATENATED ARRAYS. E.G. 1 2 3 3 2 1 MESH 2 6 ⍴'AD','BE','CF' ↔ 2 6 ⍴'ABCDEF'.
SAUCE/PREVERSE.1	PARTITIONED ⌽ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/RIGHTJUSTIFY.1	RIGHT JUSTIFIES EACH ROW OF AN ARRAY.
SAUCE/VTOM.1	RETURNS A MAT WITH ONE STRING PER ROW, WHERE EACH STRING IN THE RIGHT ARG IS PRECEDED BY 1↑RIGHTARG.
YUDI/FILL.1	FILLS A CHARACTER STRING WITH A DELIMITER FOR A SPECIFIED NUMBER OF TIMES

CATEGORY 3 PARTITIONED ARRAY HANDLING — CATEGORY 3

MJAB/ENC.1	BREAKS UP SIMPLE TEXT VECTOR INTO ENCLOSED VECTOR OF WORDS.
MRAB/ΔPVM.1	VECTOR TO MATRIX ACCORDING TO PARTITIONING BOOLEAN VECTOR
ROHAN/PARTITION.1	PARTITIONS <ARRAY> ALONG ITS LAST AXIS, INTO A VECTOR OF ENCLOSURES.
ROHAN/PARTITIONΔVECTOR.1	FASTER THAN ROHAN/PARTITION, BUT ONLY WORKS ON VECTORS.
ROHAN/SUBTOTAL.1	INSERTS FIRST-AXIS SUBTOTALS INTO AN ARRAY.
SAUCE/EXTENDPARTS.1	FOR EXTENDING PARTS. E.G. 1 1 0 0 1 0 0 EXTENDPARTS 1 0 0 0 1 1 0 ↔ 1 0 0 0 1 0 0 1 0 0 0.
SAUCE/MASKPARTS.1	E.G. 1 0 1 0 MASKPARTS 1 0 0 1 0 1 0 1 0 ↔ 1 1 1 0 0 1 1 0 0.
SAUCE/PANDREDUCE.1	PARTITIONED ∧/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PANDSCAN.1	PARTITIONED ∧\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PEQSCAN.1	PARTITIONED =\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PGRADEDOWN.1	PARTITIONED ⍒ OVER THE LAST AXIS OF OF AN ARRAY.
SAUCE/PGRADEUP.1	PARTITIONED ⍋ OVER THE LAST AXIS OF AN ARRAY.
SAUCE/PLENGTHS.2	E.G. PLENGTHS 1 1 0 0 1 0 0 0 0 1 0 ↔ 1 3 5 2.
SAUCE/PLESCAN.1	PARTITIONED ≤\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PLTSCAN.1	PARTITIONED <\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PMAXREDUCE.1	PARTITIONED ⌈/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PMAXSCAN.1	PARTITIONED ⌈\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PMINREDUCE.1	PARTITIONED ⌊/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PMINSCAN.1	PARTITIONED ⌊\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PNESCAN.1	PARTITIONED ≠\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PORREDUCE.1	PARTITIONED ∨/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PORSCAN.1	PARTITIONED ∨\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PPLUSREDUCE.1	PARTITIONED +/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PPLUSSCAN.1	PARTITIONED +\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PREVERSE.1	PARTITIONED ⌽ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

- -

CATEGORY 4 ENCLOSED ARRAYS — CATEGORY 4

DBA/RCAT.1	VERY FAST ROWWISE CATENATION OF CHARACTER VECTORS OR MATRICES
DJK/EACH.2	E.G. 'FOO ⍵*2' EACH 5⊃6⊃<⍳7 ↔ (FOO 5*2)⊃(FOO 6*2)⊃<FOO (⍳7)*2.
MJAB/ENC.1	BREAKS UP SIMPLE TEXT VECTOR INTO ENCLOSED VECTOR OF WORDS.
PLA/ENCVECFRMAT.1	ENCLOSED VECTOR FROM SIMPLE MATRIX.
RCM/ENCLARRAYS/ENCLVECFROMMAT.1	ENCLOSED VECTOR FROM SIMPLE MATRIX.
RCM/ENCLARRAYS/VECFROMENCLVEC.1	SIMPLE VECTOR FROM ENCLOSED VECTOR
ROHAN/PARTITION.1	PARTITIONS <ARRAY> ALONG ITS LAST AXIS, INTO A VECTOR OF ENCLOSURES.
ROHAN/PARTITIONΔVECTOR.1	FASTER THAN ROHAN/PARTITION, BUT ONLY WORKS ON VECTORS.
ROHAN/SIMPLE.2	DETERMINES WHETHER AN ARRAY IS SIMPLE.

CATEGORY 5 NUMERIC CALCULATION

ANO/CHOLESKY.1	CHOLESKY FACTORIZATION (DECOMPOSITION) OF A MATRIX
ANO/KRONECKER.1	∇KRONECKER CALCULATES THE KRONECKER, OR DIRECT, PRODUCT OF TWO MATRICES.
ANO/PENROSE.1	∇PENROSE CALCULATES THE GENERALIZED INVERSE OF A MATRIX
ANO/POLYDIV.1	RETURNS THE QUOTIENT OF TWO (VECTOR) POLYNOMIALS
ANO/POLYMULT.1	RETURNS THE PRODUCT OF TWO (VECTOR) POLYNOMIALS
HUI/DEPRECIATE.1	STRAIGHT-LINE DEPRECIATION.
HUI/DIOPHANTINE.1	SOLVES C = A+.×X IN POSITIVE INTEGERS.
HUI/MAVG.1	COMPUTES THE K-STEP MOVING AVERAGE ON THE LAST AXIS OF AN ARRAY.
HUI/PERMINV.1	COMPUTES INVERSE PERMUTATIONS.
HUI/POLY.1	FINDS THE COEFFICIENTS OF A POLYNOMIAL HAVING SPECIFIED ROOTS.
MGF/FFIB.1	FAST FIBONACCI FUNCTION; FFIB (⍳7)-⎕IO ↔ 0 1 1 2 3 5 8
ROHAN/SUBTOTAL.1	INSERTS FIRST-AXIS SUBTOTALS INTO AN ARRAY.
SAUCE/ALLOCEQ.1	ALLOCATES A NUMBER EQUITABLY SUBJECT TO LIMITS. E.G. 29 ALLOCEQ 3 20 500 7 ↔ 3 9.5 9.5 7.
SAUCE/ALLOCFIFO.1	ALLOCATES A NUMBER ON A FIFO BASIS, SUBJECT TO LIMITS. E.G. 9 ALLOCFIFO 4 6 11 ↔ 4 5 0.
SAUCE/DISTROUND.1	DISTRIBUTIVE ROUNDING. E.G. 0.01 DISTROUND 2 6 ⍴÷ 1 3 ↔ 2 6 ⍴ 1 0.34 1 0.33 1 0.33.
SAUCE/PMAXREDUCE.1	PARTITIONED ⌈/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PMAXSCAN.1	PARTITIONED ⌈\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PMINREDUCE.1	PARTITIONED ⌊/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PMINSCAN.1	PARTITIONED ⌊\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PPLUSREDUCE.1	PARTITIONED +/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PPLUSSCAN.1	PARTITIONED +\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/RECURRENCE.1	RESULT[I]←LEFTARGUMENT[I]+RIGHTARGUMENT[I]×RESULT[I-1]; RESULT[⎕IO]←LEFTARGUMENT[⎕IO].

CATEGORY 6 SORTING AND GRADING

DONW/NAME∆SORT.2	SORTS MATRIX OF COMPANY NAMES, EXCLUDING LEADING DEFINITE ARTICLES
HUI/CLASSIFY.1	INPUT: RANGES (INTERVALS) AND SOME NUMBERS. OUTPUT: WHICH INTERVAL EACH NUMBER IS IN.
HUI/PERMINV.1	COMPUTES INVERSE PERMUTATIONS.
HUI/SORTLOCAL.1	SORTS THE LOCAL VARIABLES IN A FUNCTION HEADER
MGF/TOPOSORT.1	TOPOLOGICAL SORT OF PRECEDENCE MATRIX.
MJAB/ALPHABETIZE.1	ALPHABETIZES MATRIX NAMELIST PUTTING 'B̲AT' BETWEEN 'BAT' AND 'CAT'.
MRAB/NAME∆SORT.1	RETURNS VECTOR OF ROW INDICES TO SORT A MATRIX OF NAMES, TEL.DIRECTORY-STYLE
SAUCE/FREQDIST.1	RETURNS THE FREQUENCY DISTRIBUTION, IN SPECIFIED CLASSES, OF NUMERIC DATA.
SAUCE/GEROWS.1	DETERMINES WHICH ROWS OF A MATRIX ARE ≥ (IN THE SENSE OF ⍋) A VECTOR.
SAUCE/GRADEDOWN.1	APPLIES ⍒ OVER THE DESIGNATED AXIS OF AN ARRAY.
SAUCE/GRADEUP.1	APPLIES ⍋ OVER THE DESIGNATED AXIS OF AN ARRAY.
SAUCE/GTROWS.1	DETERMINES WHICH ROWS OF A MATRIX ARE > (IN THE SENSE OF ⍋) A VECTOR.
SAUCE/PGRADEDOWN.1	PARTITIONED ⍒ OVER THE LAST AXIS OF OF AN ARRAY.
SAUCE/PGRADEUP.1	PARTITIONED ⍋ OVER THE LAST AXIS OF AN ARRAY.

CATEGORY 7 SEARCHING

DJK/INROWS.1	FINDS WHICH ROWS OF A MATRIX CONTAIN A STRING ANYWHERE WITHIN THEM.
HUI/CLASSIFY.1	INPUT: RANGES (INTERVALS) AND SOME NUMBERS. OUTPUT: WHICH INTERVAL EACH NUMBER IS IN.
LHG/SS.1	GENERAL STRING SEARCH PRIMITIVE
LHG/∆RPLC.1	GENERAL REPLACE OF ONE STRING BY ANOTHER.
MK/LOCATE.1	MATCHES A WORD WITH A STRING OF WORDS, RETURNING ITS LOCATION IF FOUND
MTH/UNIQUEROWS.1	FINDS ALL UNIQUE ROWS IN A TWO-DIMENSIONAL CHARACTER MATRIX
PCB/INDEX.1	RETURN ROWS OF MATRIX α WHERE ROWS OF MATRIX ω FOUND, WITH SIMILARITY CHECKING.
SAUCE/GEROWS.1	DETERMINES WHICH ROWS OF A MATRIX ARE ≥ (IN THE SENSE OF ⍋) A VECTOR.
SAUCE/GTROWS.1	DETERMINES WHICH ROWS OF A MATRIX ARE > (IN THE SENSE OF ⍋) A VECTOR.
SAUCE/INDEXOFROWS.2	RETURNS THE 'INDEX OF' EACH ROW OF ONE MATRIX IN ANOTHER MATRIX.
SAUCE/MEMBERROWS.1	RETURNS A BOOLEAN ARRAY INDICATING WHETHER EACH ROW OF ONE ARG IS A ROW OF THE OTHER ARG.
SAUCE/NOTMEMBERROWS.1	RETURNS A BOOLEAN ARRAY INDICATING WHETHER EACH ROW OF ONE ARG IS A NOT A ROW OF THE OTHER ARG.
SAUCE/PMAXREDUCE.1	PARTITIONED ⌈/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PMAXSCAN.1	PARTITIONED ⌈\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PMINREDUCE.1	PARTITIONED ⌊/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PMINSCAN.1	PARTITIONED ⌊\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PROINDEXOF.1	PROGRESSIVE DYADIC IOTA, E.G. 'ABA' PROINDEXOF 'ACAABA' ↔ ⎕IO+ 0 3 2 3 1 3.
SAUCE/RUNLENGTHS.1	E.G. RUNLENGTHS 1 9 9 4 4 4 9 ↔ 1 2 3 1.
SAUCE/STRINGFIND.2	FINDS ALL OCCURRENCES OF ONE VECTOR IN ANOTHER.

CATEGORY 8 PACKAGES

DJK/COVERFNS.1	A DESCRIPTION OF A SET OF UTILITIES FOR MAINTAINING FUNCTIONS ON FILE.
DJK/COVERFNS/CFCHANGE.1	APPLIES <CH> TO FUNCTIONS IN A PACKAGE ON FILE.
DJK/COVERFNS/CFFIND.1	RETURNS THE NAMES OF ALL FUNCTIONS IN A PACKAGE ON FILE THAT LOCALIZE ONE OR MORE SPECIFIED NAMES.
DJK/COVERFNS/CFREPLACE.1	FOR FUNCTIONS IN A PACKAGE ON FILE: REPLACES NAMES IN A FUNCTION HEADER NAMELIST.
DJK/COVERFNS/CFRETRIEVE.1	RETRIEVES THE COMPONENT THAT IS READ BY A FN (SEE THE DESCRIPTION OF DJK/COVERFNS FOR DETAILS).
DJK/COVERFNS/CFSTORE.1	PACKAGES AND STORES ON FILE APPROPRIATE OBJECTS IN THE ACTIVE WS.
DJK/PEXECUTE.1	EXECUTES AN EXPRESSION 'WITHIN' A PACKAGE.
MGF/PKGMATCH.1	AS R←α≡ω, BUT ACCEPTS ALSO PACKAGES AS ARGUMENTS.
MRAB/∆PCR.1	EXTRACT MATRIX REPRESENTATION OF A FUNCTION FROM A PACKAGE (SIMILAR TO ⎕PVAL)
PCB/FNEXTRACT.1	EXTRACT ⎕CR OF A FUNCTION IN A PACKAGE.
PCB/FNPACK.1	RETURNS PACKAGE CONTAINING FUNCTION FROM ⎕CR ARGUMENT
RMILL/SPINWSDOC.3	SUBMIT A WSDOC OF THE OBJECTS IN A PACKAGE
WGR/PACKSHOW.1	FUNCTION TO DISPLAY CONTENTS OF PACKAGE(INCLUDING FUNCTIONS AND OTHER PACKAGES) IN NEAT FORM

CDB/FILECONTENTS.2	DISPLAYS CONTENTS OF A FILE
CDE/FILEACCESS.4	THE DEFINITION OF THIS UTILITY CONTAINS THE "CDE/FILEACCESS" MANUAL.
CDE/FILEACCESS/ACCESS.2	CREATES A FILE ACCESS MATRIX
CDE/FILEACCESS/CLEARACCESS.2	CLEARS ACCESS MATRICES OF ALL REFERENCED FILES
CDE/FILEACCESS/DACCESS.1	DECODE PERMISSION NUMBERS TO FILE FUNCTION NAMES
CDE/FILEACCESS/GIVEACCESS.2	SETS THE ACCESS MATRICES OF THE REFERENCED FILE; OR'S IT IF AN ENTRY FOR AN ACCOUNT ALREADY EXISTS
CDE/FILEACCESS/LISTACCESS.2	LISTS THE FILE ACCESS MATRICES IN A GRAPHICAL WAY
CDE/FILEACCESS/PASSNO.2	APPLY PASSNUMBER; CATENATES ENCLOSURE OF ⍺ AND ⍵
CDE/FILEACCESS/REMOVEACCESS.2	REMOVES ALL OR SPECIFIC FILE ACCESS FOR SELECTED ACCOUNTS
CDE/FILEACCESS/SETACCESS.2	SETS THE ACCESS MATRICES OF SELECTED FILES, REPLACES ENTRIES OF EXITSTING REFERENCED ACCOUNTS
DJK/COVERFNS.1	A DESCRIPTION OF A SET OF UTILITIES FOR MAINTAINING FUNCTIONS ON FILE.
DJK/COVERFNS/CFCHANGE.1	APPLIES <CH> TO FUNCTIONS IN A PACKAGE ON FILE.
DJK/COVERFNS/CFFIND.1	RETURNS THE NAMES OF ALL FUNCTIONS IN A PACKAGE ON FILE THAT LOCALIZE ONE OR MORE SPECIFIED NAMES.
DJK/COVERFNS/CFREPLACE.1	FOR FUNCTIONS IN A PACKAGE ON FILE: REPLACES NAMES IN A FUNCTION HEADER NAMELIST.
DJK/COVERFNS/CFRETRIEVE.1	RETRIEVES THE COMPONENT THAT IS READ BY A FN (SEE THE DESCRIPTION OF DJK/COVERFNS FOR DETAILS).
DJK/COVERFNS/CFSTORE.1	PACKAGES AND STORES ON FILE APPROPRIATE OBJECTS IN THE ACTIVE WS.
DJK/FDELETE.1	DELETES SPECIFIED COMPONENTS FROM A FILE.
DJK/FIRSTLINES.2	DISPLAYS THE HEADER AND FIRST LINE OF TEXT OF FILED MAILBOX MESSAGES.
DJK/FIRSTLINES.3	DISPLAYS THE HEADER AND FIRST LINE OF TEXT OF FILED MAILBOX MESSAGES.
MJAB/FDELETE.1	TIES ALL FILES ON OWNERS ACCOUNT AND LOOPS THROUGH ASKING WHETHER TO DELETE THEM.
MJAB/HSPASK.5	PROMPTS USER FOR LOCATION (REMOTE OR TORONTO) TO PRINT. GETS DELIVERY INSTRUCTIONS. SUBMITS HSPREQ.
PCB/RDCIDECODE.2	RETURNS CHARACTER ARRAY OF FORMATTED DATES AND TIMES FOR ⎕RDCI TIMESTAMPS
PCB/RDCIENCODE.1	RETURNS ⎕RDCI ENDCODINGS FOR ARRAY OF DATES AND TIMES IN ⎕TS FORMAT
PESCH/FILE.1	ALLOWS EXPRESSIONS WITH FILE PRIMITIVES REFERRING TO FILES BY NAME
PESCH/TIE.1	CREATES OR FINDS TIE NUMBER GIVEN FILE NAME; AVOIDS EXHAUSTING FILE TIE QUOTA
RCM/FILES/COMPS∆REPORT.2	PREPARES A REPORT ON SPECIFIED COMPONENTS OF A FILE
RCM/FILES/DECODE∆TIME.1	DECODES SYSTEM WRITE DATE/TIME
RCM/FILES/FAPPEND.1	COVER FUNCTION FOR ⎕APPEND- RESIZES FILE IF NECESSARY
RCM/FILES/FAPPENDR.1	COVER FUNCTION FOR ⎕APPENDR- RESIZES FILE IF NECESSARY
RCM/FILES/FCOMPARE.2	COMPARES 2 FILES THAT HAVE THE SAME NUMBER OF COMPONENTS
RCM/FILES/FCOMPRESS.1	ANALOG TO PRIMITIVE / FOR FILES
RCM/FILES/FCOPY.1	COPIES PART OR ALL OF FILE AS SPECIFIED
RCM/FILES/FDROP.1	ANALOG TO PRIMITIVE ↓ FOR FILES
RCM/FILES/FERASE.1	ERASES A FILE SPECIFIED BY ARGUMENT
RCM/FILES/FEXPAND.1	ANALOG TO PRIMITIVE \ FOR FILES
RCM/FILES/FEXTENT.3	RETURNS TYPE, SPACE, RANK, AND SHAPE OF SPECIFIED COMPONENTS.
RCM/FILES/FILES∆REPORT.1	PRODUCES A REPORT ON THE FILES BELONGING TO ONE OR MORE ACCOUNTS
RCM/FILES/FINDEX.1	ANALOG TO PRIMITIVE [] FOR FILES
RCM/FILES/FINSERT.1	INSERTS DATA ITEM INTO COPY OF INPUT FILE AT SPECIFIED LOCATION
RCM/FILES/FLIB.1	COVER FUNCTION FOR ⎕LIB- RETURNS ALPHABETIZED FILE LIBRARY LIST
RCM/FILES/FMATCH.1	MATCHES COMPS OF 2 FILES WHERE ONE FILE HAS HAD COMPS INSERTED OR THE OTHER HAS HAD COMPS DELETED
RCM/FILES/FMERGE.1	MERGES SEVERAL FILES INTO A SINGLE FILE
RCM/FILES/FOPEN.2	SHARE TIES A FILE, AND CREATES IT IF NECESSARY
RCM/FILES/FPRINT.1	PRINTS CHARACTERISTICS AND/OR VALUES OF COMPONENTS IN A FILE
RCM/FILES/FREPLACE.1	COVER FUNCTION FOR ⎕REPLACE- RESIZES FILE IF NECESSARY
RCM/FILES/FRESIZE.1	COVER FUNCTION FOR ⎕RESIZE- IF ASKING TOO MUCH, GETS AS MUCH AS FILE RES ALLOWS
RCM/FILES/FREVERSE.1	ANALOG TO PRIMITIVE ⌽ FOR FILES
RCM/FILES/FROTATE.1	ANALOG TO PRIMITIVE ⊖ FOR FILES
RCM/FILES/FSPLIT.1	SPLITS THE COMPONENTS OF A FILE INTO SEVERAL FILES
RCM/FILES/FTAKE.1	ANALOG TO PRIMITIVE ↑ FOR FILES
RCM/FILES/FTS.1	RETURNS TINE STAMPS FOR DATES WHEN SPECIFIED COMPONENTS WERE WRITTEN
RCM/FILES/FWRITE.1	WRITES TO SPECIFIED COMPONENT. IF COMP DOESN'T EXIST, CREATES IT AND ANY NECESSARY PRECEDING COMPS.
RMILL/FCOMPCOPY.1	NEARLY WSFULL PROOF COPY ONE FILE COMPONENT TO ANOTHER
RMILL/ROLLAPPENDR.1	⎕APPENDR FOR A ROLLING FILE

RWILL/TIED.1	*CHECKS TO SEE WHETHER A TIED FILE IS ⎕TIE'D OR ⎕STIE'D.*
ROHAN/EASYTIE.2	*TIES A FILE AND RETURNS THE TIE NUMBER, WITH PROVISION FOR LATER UNTYING A NEWLY-TIED FILE.*
ROHAN/EASYUNTIE.1	*UNTIES A FILE TIED BY ROHAN/EASYTIE.*
ROHAN/FILES.2	*RETURNS A FORMATTED LIST OF THE FILES IN THE SPECIFIED LIBRARY.*
ROHAN/FILESPACE.1	*RETURNS NAMES AND ⎕SIZES OF ALL FILES IN THE SPECIFIED LIBRARY, IN A CHARACTER MATRIX WITH TOTAL.*
ROHAN/MAKE∆STD∆FILENAME.1	*PUTS A FILENAME INTO THE STANDARD 22-ELEMENT FORM RETURNED BY ⎕LIB AND ⎕NAMES.*
ROHAN/TIED.1	*SHOWS WHICH FILES ARE TIED TO WHAT NUMBERS, IN A READABLE FORMAT.*
WGR/FTT.2	*FORMS FORMATTED REPRESENTATION OF TIMESTAMPS FROM ⎕RDCI FORM.ALLOWS VECTOR INPUT*
WGR/REPORT.1	*RETURNS A SUMMARY REPORT OF FILES IN A LIBRARY OR SET OF FILENAMES.*
WGR/TTF.1	*FORMS ⎕RDCI[3] FORMAT ENCODED TIMESTAMP VECTOR FROM MATRIX OF ⎕TS FORM TIMESTAMPS*

CATEGORY 10 MAILBOX

DJK/FIRSTLINES.2	DISPLAYS THE HEADER AND FIRST LINE OF TEXT OF FILED MAILBOX MESSAGES.
DJK/FIRSTLINES.3	DISPLAYS THE HEADER AND FIRST LINE OF TEXT OF FILED MAILBOX MESSAGES.

- -

CATEGORY 11 CONCURRENT USE OF FILES

CDE/FILEACCESS.4	THE DEFINITION OF THIS UTILITY CONTAINS THE "CDE/FILEACCESS" MANUAL.
CDE/FILEACCESS/ACCESS.2	CREATES A FILE ACCESS MATRIX
CDE/FILEACCESS/CLEARACCESS.2	CLEARS ACCESS MATRICES OF ALL REFERENCED FILES
CDE/FILEACCESS/DACCESS.1	DECODE PERMISSION NUMBERS TO FILE FUNCTION NAMES
CDE/FILEACCESS/GIVEACCESS.2	SETS THE ACCESS MATRICES OF THE REFERENCED FILE; OR'S IT IF AN ENTRY FOR AN ACCOUNT ALREADY EXISTS
CDE/FILEACCESS/LISTACCESS.2	LISTS THE FILE ACCESS MATRICES IN A GRAPHICAL WAY
CDE/FILEACCESS/PASSNO.2	APPLY PASSNUMBER; CATENATES ENCLOSURE OF α AND ω
CDE/FILEACCESS/REMOVEACCESS.2	REMOVES ALL OR SPECIFIC FILE ACCESS FOR SELECTED ACCOUNTS
CDE/FILEACCESS/SETACCESS.2	SETS THE ACCESS MATRICES OF SELECTED FILES, REPLACES ENTRIES OF EXITSTING REFERENCED ACCOUNTS
RMILL/TIED.1	CHECKS TO SEE WHETHER A TIED FILE IS ⎕TIE'D OR ⎕STIE'D.

- -

CATEGORY 12 FILE ORGANIZATION

DJK/FDELETE.1	DELETES SPECIFIED COMPONENTS FROM A FILE.
RCM/FILES/FMERGE.1	MERGES SEVERAL FILES INTO A SINGLE FILE
WGR/REPORT.1	RETURNS A SUMMARY REPORT OF FILES IN A LIBRARY OR SET OF FILENAMES.

CATEGORY 13 FILE TOOLS

CATEGORY 13

DJK/COVERFNS.1	A DESCRIPTION OF A SET OF UTILITIES FOR MAINTAINING FUNCTIONS ON FILE.
DJK/COVERFNS/CFCHANGE.1	APPLIES <CH> TO FUNCTIONS IN A PACKAGE ON FILE.
DJK/COVERFNS/CFFIND.1	RETURNS THE NAMES OF ALL FUNCTIONS IN A PACKAGE ON FILE THAT LOCALIZE ONE OR MORE SPECIFIED NAMES.
DJK/COVERFNS/CFREPLACE.1	FOR FUNCTIONS IN A PACKAGE ON FILE: REPLACES NAMES IN A FUNCTION HEADER NAMELIST.
DJK/COVERFNS/CFRETRIEVE.1	RETRIEVES THE COMPONENT THAT IS READ BY A FN (SEE THE DESCRIPTION OF DJK/COVERFNS FOR DETAILS).
DJK/COVERFNS/CFSTORE.1	PACKAGES AND STORES ON FILE APPROPRIATE OBJECTS IN THE ACTIVE WS.
DJK/FDELETE.1	DELETES SPECIFIED COMPONENTS FROM A FILE.
MJAB/FDELETE.1	TIES ALL FILES ON OWNERS ACCOUNT AND LOOPS THROUGH ASKING WHETHER TO DELETE THEM.
PCB/RDCIDECODE.2	RETURNS CHARACTER ARRAY OF FORMATTED DATES AND TIMES FOR ⎕RDCI TIMESTAMPS
PCB/RDCIENCODE.1	RETURNS ⎕RDCI ENDCODINGS FOR ARRAY OF DATES AND TIMES IN ⎕TS FORMAT
RCM/FILES/COMPS∆REPORT.2	PREPARES A REPORT ON SPECIFIED COMPONENTS OF A FILE
RCM/FILES/FCOMPARE.2	COMPARES 2 FILES THAT HAVE THE SAME NUMBER OF COMPONENTS
RCM/FILES/FCOPY.1	COPIES PART OR ALL OF FILE AS SPECIFIED
RCM/FILES/FDROP.1	ANALOG TO PRIMITIVE ↓ FOR FILES
RCM/FILES/FEXTENT.3	RETURNS TYPE, SPACE, RANK, AND SHAPE OF SPECIFIED COMPONENTS.
RCM/FILES/FILES∆REPORT.1	PRODUCES A REPORT ON THE FILES BELONGING TO ONE OR MORE ACCOUNTS
RCM/FILES/FMATCH.1	MATCHES COMPS OF 2 FILES WHERE ONE FILE HAS HAD COMPS INSERTED OR THE OTHER HAS HAD COMPS DELETED
RCM/FILES/FMERGE.1	MERGES SEVERAL FILES INTO A SINGLE FILE
RMILL/FCOMPCOPY.1	NEARLY WSFULL PROOF COPY ONE FILE COMPONENT TO ANOTHER
RMILL/ROLLAPPENDR.1	⎕APPENDR FOR A ROLLING FILE
RMILL/TIED.1	CHECKS TO SEE WHETHER A TIED FILE IS ⎕TIE'D OR ⎕STIE'D.
WGR/FTT.2	FORMS FORMATTED REPRESENTATION OF TIMESTAMPS FROM ⎕RDCI FORM.ALLOWS VECTOR INPUT
WGR/REPORT.1	RETURNS A SUMMARY REPORT OF FILES IN A LIBRARY OR SET OF FILENAMES.
WGR/TTF.1	FORMS ⎕RDCI[3] FORMAT ENCODED TIMESTAMP VECTOR FROM MATRIX OF ⎕TS FORM TIMESTAMPS

- -

CATEGORY 14 COMMUNICATION BETWEEN TASKS

CATEGORY 14

DJK/FIRSTLINES.2	DISPLAYS THE HEADER AND FIRST LINE OF TEXT OF FILED MAILBOX MESSAGES.
DJK/FIRSTLINES.3	DISPLAYS THE HEADER AND FIRST LINE OF TEXT OF FILED MAILBOX MESSAGES.
MIKE/NTASK/RAN.1	EXECUTES FUNCTION 'NTASKWORK' AS A RESTARTABLE NTASK. TRANSFERS NAMED ITEMS TO AND FROM NTASK.
MIKE/STASK/EXEC.1	EXECUTES RIGHT ARGUMENT AS AN STASK ON ACCOUNT SPECIFIED BY LEFT ARGUMENT.
RCM/SHAREDVARS/MONITOR.1	NTASK MONITOR WHICH PROCESSES REQUESTS FROM OTHER TASKS
RCM/SHAREDVARS/NEWOFFERS.1	DETECTS OFFERS BY NEW PROCESSORS, AND SETS UP SHARES WITH THEM
RCM/SHAREDVARS/PORT.1	ALLOWS)PORT TO BE DONE UNDER PROGRAM CONTROL
RCM/SHAREDVARS/SETUP∆INTERFACE.1	SETS UP SPECIFIED INTERFACE BETWEEN TWO PROCESSORS RUNNING
RCM/SHAREDVARS/TRANSMIT1.1	TRANSMITS DATA TO PARTNER USING SIMPLEX MECHANISM
RCM/SHAREDVARS/TRANSMIT2.1	TRANSMITS DATA TO PARTNER USING HALF-DUPLEX MECHANISM
ROHAN/SVHOLD.1	HOLDS THE SPECIFIED SHARED VARIABLES, SOMEWHAT LIKE ⎕HOLD HOLDS FILES.
ROHAN/SVRELEASE.1	TO BE USED WITH ROHAN/SVHOLD.

CATEGORY 15 SHARED VARIABLES | CATEGORY 15

RCM/SHAREDVARS/NEWOFFERS.1	DETECTS OFFERS BY NEW PROCESSORS, AND SETS UP SHARES WITH THEM
RCM/SHAREDVARS/SETUP∆INTERFACE.1	SETS UP SPECIFIED INTERFACE BETWEEN TWO PROCESSORS RUNNING
RCM/SHAREDVARS/TRANSMIT1.1	TRANSMITS DATA TO PARTNER USING SIMPLEX MECHANISM
RCM/SHAREDVARS/TRANSMIT2.1	TRANSMITS DATA TO PARTNER USING HALF-DUPLEX MECHANISM
ROHAN/SVHOLD.1	HOLDS THE SPECIFIED SHARED VARIABLES, SOMEWHAT LIKE ⎕HOLD HOLDS FILES.
ROHAN/SVRELEASE.1	TO BE USED WITH ROHAN/SVHOLD.

- -

CATEGORY 16 B-TASKS | CATEGORY 16

(THERE ARE NO UTILITIES IN CATEGORY 16.)

- -

CATEGORY 17 S-TASKS | CATEGORY 17

CDB/TALK.3	ALLOWS INTERACTIVE COMMUNICATION WITH AN STASK
MIKE/STASK/EXEC.1	EXECUTES RIGHT ARGUMENT AS AN STASK ON ACCOUNT SPECIFIED BY LEFT ARGUMENT.
RCM/SHAREDVARS/STASK∆NUMS.1	CONVERTS THE CHARACTER OUTPUT OF AN STASK TO A NUMERIC ARRAY.
RCM/SHAREDVARS/STASK∆OUTPUT.1	RETURNS AN ARRAY OF OUTPUT FROM STASK

- -

CATEGORY 18 N-TASKS | CATEGORY 18

MIKE/NTASK/RAN.1	EXECUTES FUNCTION 'NTASKWORK' AS A RESTARTABLE NTASK. TRANSFERS NAMED ITEMS TO AND FROM NTASK.
RCM/SHAREDVARS/MONITOR.1	NTASK MONITOR WHICH PROCESSES REQUESTS FROM OTHER TASKS
ROHAN/N∆EXECUTE.2	EXECUTES A GIVEN STATEMENT IN AN N-TASK.

- -

CATEGORY 19 EXECUTION MONITORING | CATEGORY 19

DJK/CPU.1	MEASURES THE NUMBER OF CPU UNITS REQUIRED TO EXECUTE EXPRESSIONS.
DJK/SPACE.1	CALCULATES AN UPPER BOUND (WITHIN 1K) ON THE BYTES REQUIRED TO EXECUTE AN EXPRESSION.
HUI/RESET.1	RESETS STOP AND TRACE VECTORS OF FUNCTIONS.

BAP/PRINTTABLE.1	FORMATS A CHARACTER STRING TO FIT ON PAGE LIKE)FNS
DJK/PAUSE.1	<PAUSE> SUSPENDS. THE USER MAY RESUME EXECUTION BY ENTERING 'RESUME'.
HUI/APLTOCOURIER.1	CONVERTS TEXT IN APL FONT TO COURIER FONT.
HUI/COURIERTOAPL.1	CONVERTS TEXT IN COURIER FONT TO APL FONT.
MJAB/BSTABLE.2	TABLE OF VALID OVERSTRIKES
MJAB/BSTRANSLATE.1	CHANGES CHARACTER STRING WITH IMBEDDED BACKSPACES USED TO REPRESENT OVERSTIKES INTO TRUE OVERSTRIKES
MJAB/HDS/CHARCHANGE.1	CHANGES MESSAGE CHARACTER ON HDS108
MJAB/HDS/FINDPOS.1	REPORTS POSITION OF CURSOR ON HDS108 SCREEN.
MJAB/HDS/KEYCODE.1	TRANSLATES HDS108 KEY NUMBER INTO ⎕ARBOUT CODE FOR THAT KEY.
MJAB/HDS/KEYSET.1	SETS A KEY ON THE HDS 108.
MJAB/HDS/MOVECURSOR.1	RETURNS ⎕ARBOUT SEQUENCE NEEDED TO MOVE CURSOR ON HDS108 TO POSITION IN ARGUMENT.
MJAB/HDS/SAVESCREEN.2	SAVES THE CONTENTS OF THE HDS108 SCREEN AS AN APL CHARACTER VECTOR.
MJAB/HDS/SENDSCREEN.2	SENDS CHARACTERS FROM HDS SCREEN TO APL.
MJAB/HDS/SENDUPTO.2	SENDS CHARACTER FROM CURRENT CURSOR POSITION TO ARGUMENT POSITION.
MJAB/TEXTLOOP.1	ACCEPTS MULTIPLE LINES OF TEXT.
PKI/ADMPRINT.1	USES ADMPRINT TO QUEUE A PRINT REQUEST, FOR PROCESSING ON A 3279.
PKI/ARBIO/OVERSTRIKES.3	AN INTEGER MATRIX SPECIFYING WHICH PAIRS OF APL CHARACTERS ARE VALID OVERSTRUCK CHARACTERS
PKI/FSCLS.1	CANCELS OR ENQUEUES A FILE FOR PRINTING BY ADMPRINT.
PKI/FSLOG.1	ADDS TEXT TO QUEUED PRINTER FILE OPENED USING FSOPEN.
PKI/FSOPEN.1	OPEN QUEUED PRINTING DEVICE FOR OUTPUT (USUALLY IBM 3287) USING AP126.
PKI/GDDM.1	UTILITY TO PERFORM AP126 CALLS, WITH ERROR CHECKING.
RCM/ARBIO/ARBINPUT.1	SUBSTITUTE FOR ⍞; GETS INPUT FROM A TERMINAL.
RCM/ARBIO/ARBIN∆OVERSTRIKE.1	MAPS A SEQUENCE OF OVERSTRUCK APL CHARS INTO A SINGLE CHAR.
RCM/ARBIO/ARBOUTPUT.1	PRINTS TEXT AT A TERMINAL; A SUBSTITUTE FOR <⍞←>.
RCM/ARBIO/ARBOUT∆CARRIAGE.1	INSERTS CARRIAGE CONTROL CHARACTERS INTO TEXT; LIKELY USED WITH ⎕ARBOUT.
RCM/ARBIO/ARBOUT∆OVERSTRIKE.1	MAPS A SINGLE OVERSTRUCK APL CHARACTER INTO A SEQUENCE OF NON-OVERSTRUCK APL CHARACTERS.
RCM/ARBIO/OVERSTRIKES.1	AN INTEGER MATRIX SPECIFYING WHICH PAIRS OF APL CHARACTERS ARE VALID OVERSTRUCK CHARACTERS.
RCM/ARBIO/TRANSLATE.1	A TABLE OF THE CHARS PRINTED BY CERTAIN TERMINALS UPON RECEIPT OF 7 BIT ASCII TRANSMISSION CODES.
RCM/TERMFNS/AJ510/CHARSET.1	CHANGES AJ510 CHARACTER SET. ARGUMENT- 1→ASCII, 2→APL, 3→GRAPHICS
RCM/TERMFNS/AJ510/FEATURE.1	FOR AJ510. ARGUMENT- 0→END FEATURE, 1→UNDERLINE, 2→BLINK, 3→INVERSE VIDEO, 4→LOW INTENSITY
RCM/TERMFNS/AJ510/MOVE.1	FOR AJ510. MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)
RCM/TERMFNS/AJ510/RESET∆TERM.1	FOR AJ510. RESETS TERMINAL; SAME AS ESC) ρ
RCM/TERMFNS/AJ510/SET∆HTABS.1	FOR AJ510. SETS HORIZONTAL TAB STOPS
RCM/TERMFNS/AJ510/STATUS.1	FOR AJ510. ARGUMENT- 1 OR 0, SPECIFYING STATUS LINE ON OR OFF
RCM/TERMFNS/AJ510/UCALPHA.1	FOR AJ510. ARGUMENT- 0 OR 1, MEANING TURN UPPER CASE ALPHA OFF OR ON
RCM/TERMFNS/AJ832/AUTO∆LINEFEED.1	FOR AJ832. SETS VERTICAL PITCH TO 3 LINES/INCH
RCM/TERMFNS/AJ832/BOLD∆FACE.1	FOR AJ832. TURNS ON OR OFF PRINT ENHANCEMENT MODE
RCM/TERMFNS/AJ832/CLEAR∆HTABS.1	FOR AJ832. CLEARS ALL HORIZONTAL TAB SETTINGS
RCM/TERMFNS/AJ832/CLEAR∆VTABS.1	FOR AJ832. CLEARS ALL VERTICAL TAB SETTINGS
RCM/TERMFNS/AJ832/CONTROL∆ENABLE.1	FOR AJ832. ENABLES OR DISABLES CONTROL CODE INTERPRETATION
RCM/TERMFNS/AJ832/GOTO.1	FOR AJ832. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/AJ832/LINE∆FEED.1	FOR AJ832. CAUSES VERTICAL MOVEMENT AS SPECIFIED
RCM/TERMFNS/AJ832/MOVE.1	FOR AJ832. MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)
RCM/TERMFNS/AJ832/PLOT∆MODE.1	FOR AJ832. TURNS ON OR OFF STANDARD PLOT MODE
RCM/TERMFNS/AJ832/PLOT∆SPACING.1	FOR AJ832. SETS HORIZONTAL OR VERTICAL SPACING FOR PLOT MODE.
RCM/TERMFNS/AJ832/PRINT.1	FOR AJ832. PRINTS TEXT. USES ⎕ARBOUT.
RCM/TERMFNS/AJ832/PRINT∆ENABLE.1	FOR AJ832. ENABLE OR DISABLES TERMINAL PRINTING
RCM/TERMFNS/AJ832/RELEASE∆MARGIN.1	FOR AJ832. RELEASE MARGINS

RCM/TERMFNS/AJ832/RESET∆MARGINS.1	FOR AJ832. LEFT MARGIN RESTORED TO 0, RIGHT MARGIN TO 131
RCM/TERMFNS/AJ832/RESET∆PITCH.1	FOR AJ832. RESTORES HORIZONTAL PITCH TO SWITCH SETTING AND VERTICAL PITCH TO 6 LPI
RCM/TERMFNS/AJ832/RESET∆TERM.1	FOR AJ832. RESTORES SETTING AS WHEN POWERED UP
RCM/TERMFNS/AJ832/SELECT∆PITCH.1	FOR AJ832. SELECTS 1 OF 4 POSSIBLE PITCH SETTINGS: 1) 10H/6V, 2) 12H/6V, 3) 10H/8V, 4) 12H/8V
RCM/TERMFNS/AJ832/SET∆HTABS.1	FOR AJ832. SETS HORIZONTAL TAB STOPS
RCM/TERMFNS/AJ832/SET∆MARGIN.1	FOR AJ832. SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RCM/TERMFNS/AJ832/SET∆PAGINATION.1	FOR AJ832. SETS NUMBER OF LINES TO BE SKIPPED AT BOTTOM OF PAGE
RCM/TERMFNS/AJ832/SET∆PITCH.1	FOR AJ832. SETS INCREMENT FOR HORIZONTAL OR VERTICAL MOVEMENT
RCM/TERMFNS/AJ832/SET∆TOF.1	FOR AJ832. SETS TOP-OF-FORM
RCM/TERMFNS/AJ832/SET∆TOF∆LENGTH.1	FOR AJ832. SETS TOP-OF-FORM AND FORM LENGTH
RCM/TERMFNS/AJ832/SET∆VTABS.1	FOR AJ832. SETS VERTICAL TAB STOPS
RCM/TERMFNS/AJ832/SUBSCRIPT.1	FOR AJ832. PRINTS TEXT AS A SUBSCRIPT
RCM/TERMFNS/AJ832/SUPERSCRIPT.1	FOR AJ832. PRINTS TEXT AS SUPERSCRIPT
RCM/TERMFNS/AJ832/TWELVE∆PITCH.1	FOR AJ832. SETS HORIZONTAL PITCH TO 12 CHARS/INCH
RCM/TERMFNS/AJ832/ULTRAPLOT∆MODE.1	FOR AJ832. TURNS ULTRAPLOT MODE ON OR OFF.
RCM/TERMFNS/AJ832/TRANSLATE.1	FOR AJ832. A TABLE OF THE CHARACTERS PRINTED BY THE AJ832 UPON RECEIPT OF 7 BIT ASCII CODES.
RCM/TERMFNS/AJ860/CLEAR∆HTABS.1	FOR AJ860. CLEARS ALL HORIZONTAL TAB SETTINGS
RCM/TERMFNS/AJ860/CLEAR∆VTABS.1	FOR AJ860. CLEARS ALL VERTICAL TAB SETTINGS
RCM/TERMFNS/AJ860/EXPAND∆CHARS.1	FOR AJ860. ENABLE OR DISABLE EXPANDED CHAR SET
RCM/TERMFNS/AJ860/GOTO.1	FOR AJ860. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/AJ860/LINE∆FEED.1	FOR AJ860. GENERATES FULL AND HALF LINEFEEDS, UP OR DOWN.
RCM/TERMFNS/AJ860/PRINT.1	FOR AJ860. PRINTS TEXT. USES ⎕ARBOUT.
RCM/TERMFNS/AJ860/PRINT∆ENABLE.1	FOR AJ860. ENABLE OR DISABLES TERMINAL PRINTING
RCM/TERMFNS/AJ860/RESET∆MARGINS.1	FOR AJ860. LEFT MARGIN RESTORED TO 0, RIGHT MARGIN TO 131
RCM/TERMFNS/AJ860/RESET∆TERM.1	FOR AJ860. RESTORES TERMINAL SETTINGS TO THE STATE THEY WERE IN WHEN THE TERMINAL WAS POWERED UP.
RCM/TERMFNS/AJ860/SET∆HTABS.1	FOR AJ860. SETS HORIZONTAL TABS
RCM/TERMFNS/AJ860/SET∆MARGIN.1	FOR AJ860. SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RCM/TERMFNS/AJ860/SET∆PAGINATION.1	FOR AJ860. SETS NUMBER OF LINES TO BE SKIPPED AT BOTTOM OF PAGE
RCM/TERMFNS/AJ860/SET∆TOF.1	FOR AJ860. SETS TOP-OF-FORM
RCM/TERMFNS/AJ860/SET∆TOF∆LENGTH.1	FOR AJ860. SETS TOP-OF-FORM AND FORM LENGTH
RCM/TERMFNS/AJ860/SET∆VTABS.1	FOR AJ860. SETS VERTICAL TAB STOPS
RCM/TERMFNS/AJ860/SUBSCRIPT.1	FOR AJ860. PRINTS TEXT AS A SUBSCRIPT
RCM/TERMFNS/AJ860/SUPERSCRIPT.1	FOR AJ860. PRINTS TEXT AS SUPERSCRIPT
RCM/TERMFNS/AJ860/TRANSLATE.1	FOR AJ860. A TABLE OF THE CHARACTERS PRINTED BY THE AJ860 UPON RECEIPT OF 7 BIT ASCII CODES.
RCM/TERMFNS/DIABLO1620/AUTO∆LINEFEED.1	FOR DIABLO 1620. CAUSES TERMINAL TO PRINT WITH LINES DOUBLE-SPACED
RCM/TERMFNS/DIABLO1620/CLEAR∆HTABS.1	FOR DIABLO 1620. CLEARS ALL HORIZONTAL TABS
RCM/TERMFNS/DIABLO1620/GOTO.1	FOR DIABLO 1620. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/DIABLO1620/LINE∆FEED.1	FOR DIABLO 1620. CAUSES PLATEN TO MOVE AS SPECIFIED
RCM/TERMFNS/DIABLO1620/PLOT∆MODE.1	FOR DIABLO 1620. ARGUMENT- BOOLEAN SCALAR OR 1-ELEMENT VECTOR. 1→GRAPHICS, 0→GRAPHICS OFF
RCM/TERMFNS/DIABLO1620/PRINT.1	FOR DIABLO 1620. PRINTS TEXT. USES ⎕ARBOUT.
RCM/TERMFNS/DIABLO1620/PRINT∆COLOR.1	FOR DIABLO 1620. SELECTS COLOR OF RIBBON
RCM/TERMFNS/DIABLO1620/PRINT∆DIRECTION.1	FOR DIABLO 1620; SETS PRINT DIRECTION (FORWARD OR BACKWARD).
RCM/TERMFNS/DIABLO1620/SET∆HTABS.1	FOR DIABLO 1620. SETS HORIZONTAL TABS
RCM/TERMFNS/DIABLO1620/SET∆MARGIN.1	FOR DIABLO 1620. SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RCM/TERMFNS/DIABLO1620/SET∆PITCH.1	FOR DIABLO 1620. SET INCREMENT FOR HORIZONTAL OR VERTICAL MOVEMENT
RCM/TERMFNS/DIABLO1620/SET∆TOF.1	FOR DIABLO 1620. ALLOWS USER TO SET TOP OF FORM
RCM/TERMFNS/DIABLO1620/SUBSCRIPT.1	FOR DIABLO 1620. PRINTS TEXT AS A SUBSCRIPT
RCM/TERMFNS/DIABLO1620/SUPERSCRIPT.1	FOR DIABLO 1620. PRINTS TEXT AS SUPERSCRIPT
RCM/TERMFNS/DIABLO1620/TWELVE∆PITCH.1	FOR DIABLO 1620. CAUSES TERMINAL TO PRINT 12 CHARS/INCH
RCM/TERMFNS/DIABLO1620/TRANSLATE.1	FOR DIABLO 1620. A TABLE OF THE CHARS PRINTED BY THE DIABLO1620 UPON RECEIPT OF 7 BIT ASCII CODES.
RCM/TERMFNS/HP2641/CLEAR∆HTABS.1	FOR HP2641. CLEARS ALL HORIZONTAL TABS.
RCM/TERMFNS/HP2641/CONTROL∆CODES.1	FOR HP2641. ALLOWS CONTROL CODES TO DISPLAYED INSTEAD OF EXECUTED
RCM/TERMFNS/HP2641/CURSOR.1	FOR HP2641. MOVES CURSOR UP,DOWN,RIGHT OR LEFT N POSITIONS

RCM/TERMFNS/HP2641/DISPLAY.1	FOR HP2641. DISPLAYS APL FUNCTIONS USING SPECIAL FEATURES OF THE HP2641.
RCM/TERMFNS/HP2641/GOTO.1	FOR HP2641. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/HP2641/MENU.1	FOR HP2641. PRESENTS A MENU OF CHOICES TO THE USER AND RETURNS THE USER'S CHOICE
RCM/TERMFNS/HP2641/PAGE.1	FOR HP2641. DISPLAY THE NTH PREVIOUS OR NEXT PAGE
RCM/TERMFNS/HP2641/PRINT.1	FOR HP2641. PRINTS TEXT. USES ⎕ARBOUT.
RCM/TERMFNS/HP2641/RESET∆TERM.1	FOR HP2641. RESETS TERMINAL TO POWER-ON STATE
RCM/TERMFNS/HP2641/SCROLL.1	FOR HP2641. SCROLL DISPLAY UP OR DOWN N LINES
RCM/TERMFNS/HP2641/SET∆HTABS.1	FOR HP2641. SETS HORIZONTAL TABS
RCM/TERMFNS/HP2641/SET∆MARGIN.1	FOR HP2641. SET RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RCM/TERMFNS/HP2641/TRANSLATE.1	FOR HP2641. A TABLE OF THE CHARACTERS PRINTED BY THE HP2641 UPON RECEIPT OF 7 BIT ASCII CODES.
RCM/TERMFNS/TRENDATA4000A/CLEAR∆HTABS.1	FOR TRNDATA 4000A. CLEARS ALL HORIZONTAL TAB SETTINGS
RCM/TERMFNS/TRENDATA4000A/CLEAR∆VTABS.1	FOR TRENDATA 4000A. CLEARS ALL VERTICAL TAB SETTINGS
RCM/TERMFNS/TRENDATA4000A/GOTO.1	FOR TRENDATA 4000A. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/TRENDATA4000A/LINE∆FEED.1	FOR TRENDATA 4000A. GENERATES FULL AND HALF LINEFEEDS, UP OR DOWN.
RCM/TERMFNS/TRENDATA4000A/MOVE.1	FOR TRENDATA 4000A. MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)
RCM/TERMFNS/TRENDATA4000A/PLOT.1	FOR TRENDATA 4000A. MOVE OR DRAW IN PLOT MODE
RCM/TERMFNS/TRENDATA4000A/PLOT∆MODE.1	FOR TRENDATA 4000A. TURNS ON OR OFF STANDARD PLOT MODE
RCM/TERMFNS/TRENDATA4000A/PRINT.1	FOR TRENDATA 4000A. PRINTS TEXT. USES ⎕ARBOUT.
RCM/TERMFNS/TRENDATA4000A/PRINT∆COLOR.1	FOR TRENDATA 4000A. SELECTS COLOR OF RIBBON
RCM/TERMFNS/TRENDATA4000A/PRINT∆ENABLE.1	FOR TRENDATA 4000A. ENABLE OR DISABLES TERMINAL PRINTING
RCM/TERMFNS/TRENDATA4000A/RESET∆MARGINS.1	FOR TRENDATA 4000A. LEFT MARGIN IS RESTORED TO 0, RIGHT MARGIN RESTORED TO 131
RCM/TERMFNS/TRENDATA4000A/SELECT∆PITCH.1	FOR TRENDATA 4000A. SELECTS 1 OF 4 POSSIBLE PITCH SETTINGS: 10V/6H, 12V/6H, 10V/8H, 12V/8H
RCM/TERMFNS/TRENDATA4000A/SET∆HTABS.1	FOR TRENDATA 4000A. SETS HORIZONTAL TAB STOPS
RCM/TERMFNS/TRENDATA4000A/SET∆MARGIN.1	FOR TRENDATA 4000A. SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RCM/TERMFNS/TRENDATA4000A/SET∆PAGINATION.1	FOR TRENDATA 4000A. SETS NUMBER OF LINES TO BE SKIPPED AT BOTTOM OF PAGE
RCM/TERMFNS/TRENDATA4000A/SET∆TOF∆LENGTH.1	FOR TRENDATA 4000A. SETS TOP-OF-FORM AND FORM LENGTH
RCM/TERMFNS/TRENDATA4000A/SET∆VTABS.1	FOR TRENDATA 4000A. SETS VERTICAL TAB STOPS
RCM/TERMFNS/TRENDATA4000A/SUBSCRIPT.1	FOR TRENDATA 4000A. PRINTS TEXT AS A SUBSCRIPT
RCM/TERMFNS/TRENDATA4000A/SUPERSCRIPT.1	FOR TRENDATA 4000A. PRINTS TEXT AS SUPERSCRIPT
RCM/TERMFNS/TRENDATA4000A/TRANSLATE.1	FOR TRENDATA 4000A. A TABLE OF THE CHARS PRINTED BY THE 4000A UPON RECEIPT OF 7 BIT ASCII CODES.
ROHAN/GETINPUT.1	GETS A LINE OF CHAR INPUT, WITH AN EASY WAY TO TRY AGAIN IF THE INPUT IS LATER FOUND TO BE INVALID.
ROHAN/TABSET.2	SETS TABS, GIVEN THE TYPE OF TERMINAL AND THE DESIRED TAB SETTINGS.
WHAM/TOKEN∆DECIMALS.1	RETURN NO. DECIMALS IN 'NUMBERS' ALONG ROWS OF TEXT MATRIX
WHAM/TOKEN∆LEADZ.1	RETURNS WHETHER SINGLE NUMBER ON EACH ROW OF TEXT ARRAY CONTAINS LEAD ZEROES
WHAM/TOKEN∆SINGLE.1	RETURN WHETHER EACH ROW OF AN ARRAY CONTAINS A SINGLE TOKEN

DJK/PAUSE.1	*<PAUSE> SUSPENDS. THE USER MAY RESUME EXECUTION BY ENTERING 'RESUME'.*
MJAB/TEXTLOOP.1	*ACCEPTS MULTIPLE LINES OF TEXT.*
PKI/ARBIO/OVERSTRIKES.3	*AN INTEGER MATRIX SPECIFYING WHICH PAIRS OF APL CHARACTERS ARE VALID OVERSTRUCK CHARACTERS*
RCM/ARBIO/ARBINPUT.1	*SUBSTITUTE FOR ⍞; GETS INPUT FROM A TERMINAL.*
RCM/ARBIO/OVERSTRIKES.1	*AN INTEGER MATRIX SPECIFYING WHICH PAIRS OF APL CHARACTERS ARE VALID OVERSTRUCK CHARACTERS.*
ROHAN/GETINPUT.1	*GETS A LINE OF CHAR INPUT, WITH AN EASY WAY TO TRY AGAIN IF THE INPUT IS LATER FOUND TO BE INVALID.*
WHAM/TOKEN∆DECIMALS.1	*RETURN NO. DECIMALS IN 'NUMBERS' ALONG ROWS OF TEXT MATRIX*
WHAM/TOKEN∆LEADZ.1	*RETURNS WHETHER SINGLE NUMBER ON EACH ROW OF TEXT ARRAY CONTAINS LEAD ZEROES*
WHAM/TOKEN∆SINGLE.1	*RETURN WHETHER EACH ROW OF AN ARRAY CONTAINS A SINGLE TOKEN*

RCM/TERMFNS/AJ510/CHARSET.1	CHANGES AJ510 CHARACTER SET. ARGUMENT- 1→ASCII, 2→APL, 3→GRAPHICS
RCM/TERMFNS/AJ510/FEATURE.1	FOR AJ510. ARGUMENT- 0→END FEATURE, 1→UNDERLINE, 2→BLINK, 3→INVERSE VIDEO, 4→LOW INTENSITY
RCM/TERMFNS/AJ510/MOVE.1	FOR AJ510. MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)
RCM/TERMFNS/AJ510/RESET∆TERM.1	FOR AJ510. RESETS TERMINAL; SAME AS ESC) ρ
RCM/TERMFNS/AJ510/SET∆HTABS.1	FOR AJ510. SETS HORIZONTAL TAB STOPS
RCM/TERMFNS/AJ510/STATUS.1	FOR AJ510. ARGUMENT- 1 OR 0, SPECIFYING STATUS LINE ON OR OFF
RCM/TERMFNS/AJ510/UCALPHA.1	FOR AJ510. ARGUMENT- 0 OR 1, MEANING TURN UPPER CASE ALPHA OFF OR ON
RCM/TERMFNS/AJ832/AUTO∆LINEFEED.1	FOR AJ832. SETS VERTICAL PITCH TO 3 LINES/INCH
RCM/TERMFNS/AJ832/BOLD∆FACE.1	FOR AJ832. TURNS ON OR OFF PRINT ENHANCEMENT MODE
RCM/TERMFNS/AJ832/CLEAR∆HTABS.1	FOR AJ832. CLEARS ALL HORIZONTAL TAB SETTINGS
RCM/TERMFNS/AJ832/CLEAR∆VTABS.1	FOR AJ832. CLEARS ALL VERTICAL TAB SETTINGS
RCM/TERMFNS/AJ832/CONTROL∆ENABLE.1	FOR AJ832. ENABLES OR DISABLES CONTROL CODE INTERPRETATION
RCM/TERMFNS/AJ832/GOTO.1	FOR AJ832. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/AJ832/LINE∆FEED.1	FOR AJ832. CAUSES VERTICAL MOVEMENT AS SPECIFIED
RCM/TERMFNS/AJ832/MOVE.1	FOR AJ832. MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)
RCM/TERMFNS/AJ832/PLOT∆MODE.1	FOR AJ832. TURNS ON OR OFF STANDARD PLOT MODE
RCM/TERMFNS/AJ832/PLOT∆SPACING.1	FOR AJ832. SETS HORIZONTAL OR VERTICAL SPACING FOR PLOT MODE.
RCM/TERMFNS/AJ832/PRINT.1	FOR AJ832. PRINTS TEXT. USES □ARBOUT.
RCM/TERMFNS/AJ832/PRINT∆ENABLE.1	FOR AJ832. ENABLE OR DISABLES TERMINAL PRINTING
RCM/TERMFNS/AJ832/RELEASE∆MARGIN.1	FOR AJ832. RELEASE MARGINS
RCM/TERMFNS/AJ832/RESET∆MARGINS.1	FOR AJ832. LEFT MARGIN RESTORED TO 0, RIGHT MARGIN TO 131
RCM/TERMFNS/AJ832/RESET∆PITCH.1	FOR AJ832. RESTORES HORIZONTAL PITCH TO SWITCH SETTING AND VERTICAL PITCH TO 6 LPI
RCM/TERMFNS/AJ832/RESET∆TERM.1	FOR AJ832. RESTORES SETTING AS WHEN POWERED UP
RCM/TERMFNS/AJ832/SELECT∆PITCH.1	FOR AJ832. SELECTS 1 OF 4 POSSIBLE PITCH SETTINGS: 1) 10H/6V, 2) 12H/6V, 3) 10H/8V, 4) 12H/8V
RCM/TERMFNS/AJ832/SET∆HTABS.1	FOR AJ832. SETS HORIZONTAL TAB STOPS
RCM/TERMFNS/AJ832/SET∆MARGIN.1	FOR AJ832. SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RCM/TERMFNS/AJ832/SET∆PAGINATION.1	FOR AJ832. SETS NUMBER OF LINES TO BE SKIPPED AT BOTTOM OF PAGE
RCM/TERMFNS/AJ832/SET∆PITCH.1	FOR AJ832. SETS INCREMENT FOR HORIZONTAL OR VERTICAL MOVEMENT
RCM/TERMFNS/AJ832/SET∆TOF.1	FOR AJ832. SETS TOP-OF-FORM
RCM/TERMFNS/AJ832/SET∆TOF∆LENGTH.1	FOR AJ832. SETS TOP-OF-FORM AND FORM LENGTH
RCM/TERMFNS/AJ832/SET∆VTABS.1	FOR AJ832. SETS VERTICAL TAB STOPS
RCM/TERMFNS/AJ832/SUBSCRIPT.1	FOR AJ832. PRINTS TEXT AS A SUBSCRIPT
RCM/TERMFNS/AJ832/SUPERSCRIPT.1	FOR AJ832. PRINTS TEXT AS SUPERSCRIPT
RCM/TERMFNS/AJ832/TWELVE∆PITCH.1	FOR AJ832. SETS HORIZONTAL PITCH TO 12 CHARS/INCH
RCM/TERMFNS/AJ832/ULTRAPLOT∆MODE.1	FOR AJ832. TURNS ULTRAPLOT MODE ON OR OFF.
RCM/TERMFNS/AJ832/TRANSLATE.1	FOR AJ832. A TABLE OF THE CHARACTERS PRINTED BY THE AJ832 UPON RECEIPT OF 7 BIT ASCII CODES.
RCM/TERMFNS/AJ860/CLEAR∆HTABS.1	FOR AJ860. CLEARS ALL HORIZONTAL TAB SETTINGS
RCM/TERMFNS/AJ860/CLEAR∆VTABS.1	FOR AJ860. CLEARS ALL VERTICAL TAB SETTINGS
RCM/TERMFNS/AJ860/EXPAND∆CHARS.1	FOR AJ860. ENABLE OR DISABLE EXPANDED CHAR SET
RCM/TERMFNS/AJ860/GOTO.1	FOR AJ860. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/AJ860/LINE∆FEED.1	FOR AJ860. GENERATES FULL AND HALF LINEFEEDS, UP OR DOWN.
RCM/TERMFNS/AJ860/PRINT.1	FOR AJ860. PRINTS TEXT. USES □ARBOUT.
RCM/TERMFNS/AJ860/PRINT∆ENABLE.1	FOR AJ860. ENABLE OR DISABLES TERMINAL PRINTING
RCM/TERMFNS/AJ860/RESET∆MARGINS.1	FOR AJ860. LEFT MARGIN RESTORED TO 0, RIGHT MARGIN TO 131
RCM/TERMFNS/AJ860/RESET∆TERM.1	FOR AJ860. RESTORES TERMINAL SETTINGS TO THE STATE THEY WERE IN WHEN THE TERMINAL WAS POWERED UP.
RCM/TERMFNS/AJ860/SET∆HTABS.1	FOR AJ860. SETS HORIZONTAL TABS
RCM/TERMFNS/AJ860/SET∆MARGIN.1	FOR AJ860. SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RCM/TERMFNS/AJ860/SET∆PAGINATION.1	FOR AJ860. SETS NUMBER OF LINES TO BE SKIPPED AT BOTTOM OF PAGE
RCM/TERMFNS/AJ860/SET∆TOF.1	FOR AJ860. SETS TOP-OF-FORM
RCM/TERMFNS/AJ860/SET∆TOF∆LENGTH.1	FOR AJ860. SETS TOP-OF-FORM AND FORM LENGTH

```
RCM/TERMFNS/AJ860/SET∆VTABS.1          FOR AJ860.  SETS VERTICAL TAB STOPS
RCM/TERMFNS/AJ860/SUBSCRIPT.1          FOR AJ860.  PRINTS TEXT AS A SUBSCRIPT
RCM/TERMFNS/AJ860/SUPERSCRIPT.1        FOR AJ860.  PRINTS TEXT AS SUPERSCRIPT
RCM/TERMFNS/AJ860/TRANSLATE.1          FOR AJ860.  A TABLE OF THE CHARACTERS PRINTED BY THE AJ860 UPON RECEIPT OF 7 BIT
                                       ASCII CODES.

RCM/TERMFNS/DIABLO1620/AUTO∆LINEFEED.1 FOR DIABLO 1620.  CAUSES TERMINAL TO PRINT WITH LINES DOUBLE-SPACED
RCM/TERMFNS/DIABLO1620/CLEAR∆HTABS.1   FOR DIABLO 1620.  CLEARS ALL HORIZONTAL TABS
RCM/TERMFNS/DIABLO1620/GOTO.1          FOR DIABLO 1620.  ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/DIABLO1620/LINE∆FEED.1     FOR DIABLO 1620.  CAUSES PLATEN TO MOVE AS SPECIFIED
RCM/TERMFNS/DIABLO1620/PLOT∆MODE.1     FOR DIABLO 1620.  ARGUMENT- BOOLEAN SCALAR OR 1-ELEMENT VECTOR. 1→GRAPHICS, 0→
                                       GRAPHICS OFF
RCM/TERMFNS/DIABLO1620/PRINT.1         FOR DIABLO 1620.  PRINTS TEXT.  USES □ARBOUT.
RCM/TERMFNS/DIABLO1620/PRINT∆COLOR.1   FOR DIABLO 1620.  SELECTS COLOR OF RIBBON
RCM/TERMFNS/DIABLO1620/PRINT∆DIRECTION.1 FOR DIABLO 1620; SETS PRINT DIRECTION (FORWARD OR BACKWARD).
RCM/TERMFNS/DIABLO1620/SET∆HTABS.1     FOR DIABLO 1620.  SETS HORIZONTAL TABS
RCM/TERMFNS/DIABLO1620/SET∆MARGIN.1    FOR DIABLO 1620.  SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RCM/TERMFNS/DIABLO1620/SET∆PITCH.1     FOR DIABLO 1620.  SET INCREMENT FOR HORIZONTAL OR VERTICAL MOVEMENT
RCM/TERMFNS/DIABLO1620/SET∆TOF.1       FOR DIABLO 1620.  ALLOWS USER TO SET TOP OF FORM
RCM/TERMFNS/DIABLO1620/SUBSCRIPT.1     FOR DIABLO 1620.  PRINTS TEXT AS A SUBSCRIPT
RCM/TERMFNS/DIABLO1620/SUPERSCRIPT.1   FOR DIABLO 1620.  PRINTS TEXT AS SUPERSCRIPT
RCM/TERMFNS/DIABLO1620/TWELVE∆PITCH.1  FOR DIABLO 1620.  CAUSES TERMINAL TO PRINT 12 CHARS/INCH
RCM/TERMFNS/DIABLO1620/TRANSLATE.1     FOR DIABLO 1620.  A TABLE OF THE CHARS PRINTED BY THE DIABLO1620 UPON RECEIPT OF 7
                                       BIT ASCII CODES.

RCM/TERMFNS/HP2641/CLEAR∆HTABS.1       FOR HP2641.  CLEARS ALL HORIZONTAL TABS.
RCM/TERMFNS/HP2641/CONTROL∆CODES.1     FOR HP2641.  ALLOWS CONTROL CODES TO DISPLAYED INSTEAD OF EXECUTED
RCM/TERMFNS/HP2641/CURSOR.1            FOR HP2641.  MOVES CURSOR UP,DOWN,RIGHT OR LEFT N POSITIONS
RCM/TERMFNS/HP2641/DISPLAY.1           FOR HP2641.  DISPLAYS APL FUNCTIONS USING SPECIAL FEATURES OF THE HP2641.
RCM/TERMFNS/HP2641/GOTO.1              FOR HP2641.  ABSOLUTE TAB  TO SPECIFIED LOCATION
RCM/TERMFNS/HP2641/MENU.1              FOR HP2641.  PRESENTS A MENU OF CHOICES TO THE USER AND RETURNS THE USER'S CHOICE
RCM/TERMFNS/HP2641/PAGE.1              FOR HP2641.  DISPLAY THE NTH PREVIOUS OR NEXT PAGE
RCM/TERMFNS/HP2641/PRINT.1             FOR HP2641.  PRINTS TEXT.  USES □ARBOUT.
RCM/TERMFNS/HP2641/RESET∆TERM.1        FOR HP2641.  RESETS TERMINAL TO POWER-ON STATE
RCM/TERMFNS/HP2641/SCROLL.1            FOR HP2641.  SCROLL DISPLAY UP OR DOWN N LINES
RCM/TERMFNS/HP2641/SET∆HTABS.1         FOR HP2641.  SETS HORIZONTAL TABS
RCM/TERMFNS/HP2641/SET∆MARGIN.1        FOR HP2641.  SET RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RCM/TERMFNS/HP2641/TRANSLATE.1         FOR HP2641.  A TABLE OF THE CHARACTERS PRINTED BY THE HP2641 UPON RECEIPT OF 7 BIT
                                       ASCII CODES.

RCM/TERMFNS/TRENDATA4000A/CLEAR∆HTABS.1 FOR TRNDATA 4000A.  CLEARS ALL HORIZONTAL TAB SETTINGS
RCM/TERMFNS/TRENDATA4000A/CLEAR∆VTABS.1 FOR TRENDATA 4000A.  CLEARS ALL VERTICAL TAB SETTINGS
RCM/TERMFNS/TRENDATA4000A/GOTO.1       FOR TRENDATA 4000A.  ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/TRENDATA4000A/LINE∆FEED.1  FOR TRENDATA 4000A.  GENERATES FULL AND HALF LINEFEEDS, UP OR DOWN.
RCM/TERMFNS/TRENDATA4000A/MOVE.1       FOR TRENDATA 4000A.  MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION
                                       (RELATIVE TAB)
RCM/TERMFNS/TRENDATA4000A/PLOT.1       FOR TRENDATA 4000A.  MOVE OR DRAW IN PLOT MODE
RCM/TERMFNS/TRENDATA4000A/PLOT∆MODE.1  FOR TRENDATA 4000A.  TURNS ON OR OFF STANDARD PLOT MODE
RCM/TERMFNS/TRENDATA4000A/PRINT.1      FOR TRENDATA 4000A.  PRINTS TEXT.  USES □ARBOUT.
RCM/TERMFNS/TRENDATA4000A/PRINT∆COLOR.1 FOR TRENDATA 4000A.  SELECTS COLOR OF RIBBON
RCM/TERMFNS/TRENDATA4000A/PRINT∆ENABLE.1 FOR TRENDATA 4000A.  ENABLE OR DISABLES TERMINAL PRINTING
RCM/TERMFNS/TRENDATA4000A/RESET∆MARGINS.1 FOR TRENDATA 4000A.  LEFT MARGIN IS RESTORED TO 0, RIGHT MARGIN RESTORED TO 131
RCM/TERMFNS/TRENDATA4000A/SELECT∆PITCH.1 FOR TRENDATA 4000A.  SELECTS 1 OF 4 POSSIBLE PITCH SETTINGS: 10V/6H, 12V/6H,
                                       10V/8H, 12V/8H
RCM/TERMFNS/TRENDATA4000A/SET∆HTABS.1  FOR TRENDATA 4000A.  SETS HORIZONTAL TAB STOPS
RCM/TERMFNS/TRENDATA4000A/SET∆MARGIN.1 FOR TRENDATA 4000A.  SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RCM/TERMFNS/TRENDATA4000A/SET∆PAGINATION.1 FOR TRENDATA 4000A.  SETS NUMBER OF LINES TO BE SKIPPED AT BOTTOM OF PAGE
RCM/TERMFNS/TRENDATA4000A/SET∆TOF∆LENGTH.1 FOR TRENDATA 4000A.  SETS TOP-OF-FORM AND FORM LENGTH
RCM/TERMFNS/TRENDATA4000A/SET∆VTABS.1  FOR TRENDATA 4000A.  SETS VERTICAL TAB STOPS
RCM/TERMFNS/TRENDATA4000A/SUBSCRIPT.1  FOR TRENDATA 4000A.  PRINTS TEXT AS A SUBSCRIPT
RCM/TERMFNS/TRENDATA4000A/SUPERSCRIPT.1 FOR TRENDATA 4000A.  PRINTS TEXT AS SUPERSCRIPT
RCM/TERMFNS/TRENDATA4000A/TRANSLATE.1  FOR TRENDATA 4000A.  A TABLE OF THE CHARS PRINTED BY THE 4000A UPON RECEIPT OF 7
```

ROHAN/TABSET.2

BIT ASCII CODES.
SETS TABS, GIVEN THE TYPE OF TERMINAL AND THE DESIRED TAB SETTINGS.

RCM/TERMFNS/AJ510/MOVE.1	FOR AJ510. MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)
RCM/TERMFNS/AJ510/SET∆HTABS.1	FOR AJ510. SETS HORIZONTAL TAB STOPS
RCM/TERMFNS/AJ832/CLEAR∆HTABS.1	FOR AJ832. CLEARS ALL HORIZONTAL TAB SETTINGS
RCM/TERMFNS/AJ832/CLEAR∆VTABS.1	FOR AJ832. CLEARS ALL VERTICAL TAB SETTINGS
RCM/TERMFNS/AJ832/GOTO.1	FOR AJ832. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/AJ832/MOVE.1	FOR AJ832. MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)
RCM/TERMFNS/AJ832/SET∆HTABS.1	FOR AJ832. SETS HORIZONTAL TAB STOPS
RCM/TERMFNS/AJ832/SET∆VTABS.1	FOR AJ832. SETS VERTICAL TAB STOPS
RCM/TERMFNS/AJ860/CLEAR∆HTABS.1	FOR AJ860. CLEARS ALL HORIZONTAL TAB SETTINGS
RCM/TERMFNS/AJ860/CLEAR∆VTABS.1	FOR AJ860. CLEARS ALL VERTICAL TAB SETTINGS
RCM/TERMFNS/AJ860/GOTO.1	FOR AJ860. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/AJ860/SET∆HTABS.1	FOR AJ860. SETS HORIZONTAL TABS
RCM/TERMFNS/AJ860/SET∆VTABS.1	FOR AJ860. SETS VERTICAL TAB STOPS
RCM/TERMFNS/DIABLO1620/CLEAR∆HTABS.1	FOR DIABLO 1620. CLEARS ALL HORIZONTAL TABS
RCM/TERMFNS/DIABLO1620/GOTO.1	FOR DIABLO 1620. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/DIABLO1620/SET∆HTABS.1	FOR DIABLO 1620. SETS HORIZONTAL TABS
RCM/TERMFNS/HP2641/CLEAR∆HTABS.1	FOR HP2641. CLEARS ALL HORIZONTAL TABS.
RCM/TERMFNS/HP2641/GOTO.1	FOR HP2641. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/HP2641/SET∆HTABS.1	FOR HP2641. SETS HORIZONTAL TABS
RCM/TERMFNS/TRENDATA4000A/CLEAR∆HTABS.1	FOR TRNDATA 4000A. CLEARS ALL HORIZONTAL TAB SETTINGS
RCM/TERMFNS/TRENDATA4000A/CLEAR∆VTABS.1	FOR TRENDATA 4000A. CLEARS ALL VERTICAL TAB SETTINGS
RCM/TERMFNS/TRENDATA4000A/GOTO.1	FOR TRENDATA 4000A. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/TRENDATA4000A/MOVE.1	FOR TRENDATA 4000A. MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)
RCM/TERMFNS/TRENDATA4000A/SET∆HTABS.1	FOR TRENDATA 4000A. SETS HORIZONTAL TAB STOPS
RCM/TERMFNS/TRENDATA4000A/SET∆VTABS.1	FOR TRENDATA 4000A. SETS VERTICAL TAB STOPS
ROHAN/TABSET.2	SETS TABS, GIVEN THE TYPE OF TERMINAL AND THE DESIRED TAB SETTINGS.

DJK/COVERFNS/CFCHANGE.1	APPLIES <CH> TO FUNCTIONS IN A PACKAGE ON FILE.
DJK/INROWS.1	FINDS WHICH ROWS OF A MATRIX CONTAIN A STRING ANYWHERE WITHIN THEM.
DLF/DFN.1	TAKE ⎕CR OF FUNCTION; RETURN ⎕CR WITH ASSIGNED VARIABLES LOCALIZED
HUI/APLTOCOURIER.1	CONVERTS TEXT IN APL FONT TO COURIER FONT.
HUI/COURIERTOAPL.1	CONVERTS TEXT IN COURIER FONT TO APL FONT.
LHG/∆RPLC.1	GENERAL REPLACE OF ONE STRING BY ANOTHER.
MJAB/BSTRANSLATE.1	CHANGES CHARACTER STRING WITH IMBEDDED BACKSPACES USED TO REPRESENT OVERSTIKES INTO TRUE OVERSTRIKES
MJAB/ENC.1	BREAKS UP SIMPLE TEXT VECTOR INTO ENCLOSED VECTOR OF WORDS.
RCM/ARBIO/ARBIN∆EDIT.1	REMOVES 'DELETED' STRINGS IN A VECTOR CONTAINING BACKSPACES AND LINEFEEDS.
RCM/ARBIO/ARBIN∆OVERSTRIKE.1	MAPS A SEQUENCE OF OVERSTRUCK APL CHARS INTO A SINGLE CHAR.
ROHAN/EDITMAT.1	CHARACTER-MATRIX EDITOR WHICH BEHAVES LIKE 'CH' IN MAILBOX.
ROHAN/EDITVEC.1	CHARACTER-VECTOR EDITOR LIKE THAT IN MAILBOX. HANDLES NEWLINES, BACKSPACES, ETC.
SAUCE/ELEMREPLACE.1	REPLACES, IN A VECTOR CONTAINING NO ENCLOSURES, ALL OCCURRENCES OF A SCALAR BY A (DIFFERENT) VECTOR.

BAP/PRINTTABLE.1	FORMATS A CHARACTER STRING TO FIT ON PAGE LIKE)FNS
DJK/FORMATTS.2	E.G. FORMATTS 1982 9 26 5 52 4 37 ↔ '1982-09-26 05:52:04.037'.
GLO/PARA.1	FORMATS A VECTOR OF TEXT TO WITHIN A SPECIFIED WIDTH.
HUI/INWORDS.1	CONVERTS AN INTEGER INTO ENGLISH WORDS
HUI/PASTE.1	GIVEN TWO VECTORS OF FORMATTED TEXT, WILL PUT THEM TOGETHER SIDE BY SIDE.
MRAB/FORMATΔTS.1	CONVERTS ⎕TS-STYLE TIMES INTO ISO-COMPATIBLE CHARACTER REPRESENTATIONS.
MRAB/ΔBOX.1	SIMULATION OF APL.68000'S ⎕BOX; MAKE MATRIX FROM VECTOR OR VECTOR FROM MATRIX
MRAB/ΔPVM.1	VECTOR TO MATRIX ACCORDING TO PARTITIONING BOOLEAN VECTOR
ROHAN/CENTRE.1	CENTRES A VECTOR OF TEXT (SUCH AS A HEADING FOR A REPORT).
ROHAN/COLUMNIZE.2	GIVEN A NAMELIST (OR SIMILAR MATRIX), PUTS THE NAMES IN COLUMNS ACROSS THE PAGE.
ROHAN/FILES.2	RETURNS A FORMATTED LIST OF THE FILES IN THE SPECIFIED LIBRARY.
ROHAN/FORMATΔNL.1	FORMATS A MATRIX NAMELIST JUST LIKE)FNS,)VARS, AND)GRPS DO. RESULT: A MATRIX OF SPECIFIED WIDTH.
ROHAN/TIED.1	SHOWS WHICH FILES ARE TIED TO WHAT NUMBERS, IN A READABLE FORMAT.
SAUCE/LEFTJUSTIFY.1	LEFT JUSTIFIES EACH ROW OF AN ARRAY.
SAUCE/REMOVETRAIL.1	REMOVES TRAILING COLUMNS OR ELEMENTS FROM AN ARRAY.
SAUCE/RIGHTJUSTIFY.1	RIGHT JUSTIFIES EACH ROW OF AN ARRAY.
SAUCE/SQUEEZE.1	REMOVES ALL LEADING, TRAILING, AND REDUNDANT ELEMENTS (SPEC'D IN THE LEFT ARGUMENT) FROM A VECTOR.
SAUCE/VTOM.1	RETURNS A MAT WITH ONE STRING PER ROW, WHERE EACH STRING IN THE RIGHT ARG IS PRECEDED BY 1↑RIGHTARG.

- -

CATEGORY 26 EXECUTION CONTROL CATEGORY 26

HUI/RESET.1	RESETS STOP AND TRACE VECTORS OF FUNCTIONS.
ROHAN/IF.1	USAGE: →IF CONDITION ⋄ ACTIONΔIFΔCONDITIONΔTRUE
ROHAN/UNLESS.1	USAGE: →UNLESS CONDITION ⋄ ACTIONΔUNLESSΔCONDITIONΔTRUE

- -

CATEGORY 27 TEXT PROCESSING CATEGORY 27

DLF/DFN.1	TAKE ⎕CR OF FUNCTION; RETURN ⎕CR WITH ASSIGNED VARIABLES LOCALIZED
GLO/PARA.1	FORMATS A VECTOR OF TEXT TO WITHIN A SPECIFIED WIDTH.
HUI/APLTOCOURIER.1	CONVERTS TEXT IN APL FONT TO COURIER FONT.
HUI/COURIERTOAPL.1	CONVERTS TEXT IN COURIER FONT TO APL FONT.
HUI/INWORDS.1	CONVERTS AN INTEGER INTO ENGLISH WORDS
HUI/PASTE.1	GIVEN TWO VECTORS OF FORMATTED TEXT, WILL PUT THEM TOGETHER SIDE BY SIDE.
LHG/ΔRPLC.1	GENERAL REPLACE OF ONE STRING BY ANOTHER.
MJAB/ALPHABETIZE.1	ALPHABETIZES MATRIX NAMELIST PUTTING 'B̲AT' BETWEEN 'BAT' AND 'CAT'.
MJAB/FDEFINE.1	TAKES VECTOR OF ASCII CHARACTERS REPRESENTING AN APL FUNCTION AND DEFINES FUNCTION.
PKI/ARBIO/O̲V̲E̲R̲S̲T̲R̲I̲K̲E̲S̲.3	AN INTEGER MATRIX SPECIFYING WHICH PAIRS OF APL CHARACTERS ARE VALID OVERSTRUCK CHARACTERS
RCM/ARBIO/ARBINΔEDIT.1	REMOVES 'DELETED' STRINGS IN A VECTOR CONTAINING BACKSPACES AND LINEFEEDS.
RCM/ARBIO/ARBINΔOVERSTRIKE.1	MAPS A SEQUENCE OF OVERSTRUCK APL CHARS INTO A SINGLE CHAR.
RCM/ARBIO/O̲V̲E̲R̲S̲T̲R̲I̲K̲E̲S̲.1	AN INTEGER MATRIX SPECIFYING WHICH PAIRS OF APL CHARACTERS ARE VALID OVERSTRUCK CHARACTERS.
ROHAN/CENTRE.1	CENTRES A VECTOR OF TEXT (SUCH AS A HEADING FOR A REPORT).
SAUCE/ELEMREPLACE.1	REPLACES, IN A VECTOR CONTAINING NO ENCLOSURES, ALL OCCURRENCES OF A SCALAR BY A (DIFFERENT) VECTOR.
SAUCE/VTOM.1	RETURNS A MAT WITH ONE STRING PER ROW, WHERE EACH STRING IN THE RIGHT ARG IS PRECEDED BY 1↑RIGHTARG.
YUDI/FILL.1	FILLS A CHARACTER STRING WITH A DELIMITER FOR A SPECIFIED NUMBER OF TIMES

CDB/FNCOMPARE.1	COMPARE TWO FUNCTIONS
DJK/COVERFNS.1	A DESCRIPTION OF A SET OF UTILITIES FOR MAINTAINING FUNCTIONS ON FILE.
DJK/COVERFNS/CFCHANGE.1	APPLIES <CH> TO FUNCTIONS IN A PACKAGE ON FILE.
DJK/COVERFNS/CFFIND.1	RETURNS THE NAMES OF ALL FUNCTIONS IN A PACKAGE ON FILE THAT LOCALIZE ONE OR MORE SPECIFIED NAMES.
DJK/COVERFNS/CFREPLACE.1	FOR FUNCTIONS IN A PACKAGE ON FILE: REPLACES NAMES IN A FUNCTION HEADER NAMELIST.
DJK/COVERFNS/CFRETRIEVE.1	RETRIEVES THE COMPONENT THAT IS READ BY A FN (SEE THE DESCRIPTION OF DJK/COVERFNS FOR DETAILS).
DJK/COVERFNS/CFSTORE.1	PACKAGES AND STORES ON FILE APPROPRIATE OBJECTS IN THE ACTIVE WS.
DJK/CPU.1	MEASURES THE NUMBER OF CPU UNITS REQUIRED TO EXECUTE EXPRESSIONS.
DJK/DECOMMENT.1	REMOVES COMMENTS FROM SPECIFIED FUNCTIONS.
DJK/FNSUMMARY.1	DISPLAYS SYNTAX AND LEADING FULL-LINE COMMENTS OF SPECIFIED FUNCTIONS.
DJK/SPACE.1	CALCULATES AN UPPER BOUND (WITHIN 1K) ON THE BYTES REQUIRED TO EXECUTE AN EXPRESSION.
DLF/DFN.1	TAKE ⎕CR OF FUNCTION; RETURN ⎕CR WITH ASSIGNED VARIABLES LOCALIZED
HUI/GLOBAL.1	RETURNS A MATRIX OF GLOBAL IDENTIFIERS REFERENCED BY A GIVEN FUNCTION
HUI/RESET.1	RESETS STOP AND TRACE VECTORS OF FUNCTIONS.
HUI/SORTLOCAL.1	SORTS THE LOCAL VARIABLES IN A FUNCTION HEADER
HUI/XREF.1	RETURNS A LISTING OF THE XREF OF A FUNCTION
JEW/WSCOMPARE.1	COMPARES TWO WORKSPACES.
LLF/LISTFNS.1	FORMATTED LISTING OF FUNCTIONS NAMED IN RIGHT ARGUMENT
LLF/LISTFNSPREFIX.1	LIST ALL FUNCTIONS IN THE WS WHOSE NAMES BEGIN WITH PREFIX IN RIGHT ARGUMENT
LLF/LISTGRP.1	FORMATTED LISTING OF OBJECTS (NOT PACKAGES) IN GROUP NAMED IN RIGHT ARGUMENT
MJAB/FDEFINE.1	TAKES VECTOR OF ASCII CHARACTERS REPRESENTING AN APL FUNCTION AND DEFINES FUNCTION.
MRAB/ΔPCR.1	EXTRACT MATRIX REPRESENTATION OF A FUNCTION FROM A PACKAGE (SIMILAR TO ⎕PVAL)
PCB/DISF.3	FORMATTED DISPLAY OF FUNCTIONS IN PACKAGE ARGUMENT
PCB/FNEXTRACT.1	EXTRACT ⎕CR OF A FUNCTION IN A PACKAGE.
PCB/FNNAME.1	EXTRACTS FUNCTIONS NAME FROM ITS CANONICAL REPRESENTATION
PCB/FNPACK.1	RETURNS PACKAGE CONTAINING FUNCTION FROM ⎕CR ARGUMENT
PLEB/SC.1	SHIFTS TRAILING COMMENTS OF FUNCTION TO SPECIFIED COLUMN
ROHAN/IF.1	USAGE: →IF CONDITION ⋄ ACTIONΔIFΔCONDITIONΔTRUE
ROHAN/UNLESS.1	USAGE: →UNLESS CONDITION ⋄ ACTIONΔUNLESSΔCONDITIONΔTRUE

- -

CATEGORY 29 DEBUGGING *CATEGORY 29*

HUI/RESET.1	RESETS STOP AND TRACE VECTORS OF FUNCTIONS.

- -

CATEGORY 30 REPORT FORMATTING *CATEGORY 30*

BAP/PRINTTABLE.1	FORMATS A CHARACTER STRING TO FIT ON PAGE LIKE)FNS
ROHAN/CENTRE.1	CENTRES A VECTOR OF TEXT (SUCH AS A HEADING FOR A REPORT).
ROHAN/COLUMNIZE.2	GIVEN A NAMELIST (OR SIMILAR MATRIX), PUTS THE NAMES IN COLUMNS ACROSS THE PAGE.
SAUCE/LEFTJUSTIFY.1	LEFT JUSTIFIES EACH ROW OF AN ARRAY.
SAUCE/RIGHTJUSTIFY.1	RIGHT JUSTIFIES EACH ROW OF AN ARRAY.
SAUCE/VTOM.1	RETURNS A MAT WITH ONE STRING PER ROW, WHERE EACH STRING IN THE RIGHT ARG IS PRECEDED BY 1↑RIGHTARG.

MJAB/BSTRANSLATE.1	CHANGES CHARACTER STRING WITH IMBEDDED BACKSPACES USED TO REPRESENT OVERSTIKES INTO TRUE OVERSTRIKES
MJAB/HDS/CHARCHANGE.1	CHANGES MESSAGE CHARACTER ON HDS108
MJAB/HDS/FINDPOS.1	REPORTS POSITION OF CURSOR ON HDS108 SCREEN.
MJAB/HDS/KEYCODE.1	TRANSLATES HDS108 KEY NUMBER INTO ⎕ARBOUT CODE FOR THAT KEY.
MJAB/HDS/KEYSET.1	SETS A KEY ON THE HDS 108.
MJAB/HDS/MOVECURSOR.1	RETURNS ⎕ARBOUT SEQUENCE NEEDED TO MOVE CURSOR ON HDS108 TO POSITION IN ARGUMENT.
MJAB/HDS/SAVESCREEN.2	SAVES THE CONTENTS OF THE HDS108 SCREEN AS AN APL CHARACTER VECTOR.
MJAB/HDS/SENDSCREEN.2	SENDS CHARACTERS FROM HDS SCREEN TO APL.
MJAB/HDS/SENDUPTO.2	SENDS CHARACTER FROM CURRENT CURSOR POSITION TO ARGUMENT POSITION.
PKI/ADMPRINT.1	USES ADMPRINT TO QUEUE A PRINT REQUEST, FOR PROCESSING ON A 3279.
PKI/FSCLS.1	CANCELS OR ENQUEUES A FILE FOR PRINTING BY ADMPRINT.
PKI/FSLOG.1	ADDS TEXT TO QUEUED PRINTER FILE OPENED USING FSOPEN.
PKI/FSOPEN.1	OPEN QUEUED PRINTING DEVICE FOR OUTPUT (USUALLY IBM 3287) USING AP126.
PKI/GDDM.1	UTILITY TO PERFORM AP126 CALLS, WITH ERROR CHECKING.
RCM/ARBIO/ARBIN∆OVERSTRIKE.1	MAPS A SEQUENCE OF OVERSTRUCK APL CHARS INTO A SINGLE CHAR.
RCM/ARBIO/ARBOUT∆OVERSTRIKE.1	MAPS A SINGLE OVERSTRUCK APL CHARACTER INTO A SEQUENCE OF NON-OVERSTRUCK APL CHARACTERS.
RCM/TERMFNS/HP2641/CLEAR∆HTABS.1	FOR HP2641. CLEARS ALL HORIZONTAL TABS.
RCM/TERMFNS/HP2641/CONTROL∆CODES.1	FOR HP2641. ALLOWS CONTROL CODES TO DISPLAYED INSTEAD OF EXECUTED
RCM/TERMFNS/HP2641/CURSOR.1	FOR HP2641. MOVES CURSOR UP,DOWN,RIGHT OR LEFT N POSITIONS
RCM/TERMFNS/HP2641/DISPLAY.1	FOR HP2641. DISPLAYS APL FUNCTIONS USING SPECIAL FEATURES OF THE HP2641.
RCM/TERMFNS/HP2641/GOTO.1	FOR HP2641. ABSOLUTE TAB TO SPECIFIED LOCATION
RCM/TERMFNS/HP2641/MENU.1	FOR HP2641. PRESENTS A MENU OF CHOICES TO THE USER AND RETURNS THE USER'S CHOICE
RCM/TERMFNS/HP2641/PAGE.1	FOR HP2641. DISPLAY THE NTH PREVIOUS OR NEXT PAGE
RCM/TERMFNS/HP2641/PRINT.1	FOR HP2641. PRINTS TEXT. USES ⎕ARBOUT.
RCM/TERMFNS/HP2641/RESET∆TERM.1	FOR HP2641. RESETS TERMINAL TO POWER-ON STATE
RCM/TERMFNS/HP2641/SCROLL.1	FOR HP2641. SCROLL DISPLAY UP OR DOWN N LINES
RCM/TERMFNS/HP2641/SET∆HTABS.1	FOR HP2641. SETS HORIZONTAL TABS
RCM/TERMFNS/HP2641/SET∆MARGIN.1	FOR HP2641. SET RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RCM/TERMFNS/HP2641/TRANSLATE.1	FOR HP2641. A TABLE OF THE CHARACTERS PRINTED BY THE HP2641 UPON RECEIPT OF 7 BIT ASCII CODES.

- -

CATEGORY 32 FILE PERMISSION CONTROL CATEGORY 32

CDE/FILEACCESS.4	THE DEFINITION OF THIS UTILITY CONTAINS THE "CDE/FILEACCESS" MANUAL.
CDE/FILEACCESS/ACCESS.2	CREATES A FILE ACCESS MATRIX
CDE/FILEACCESS/CLEARACCESS.2	CLEARS ACCESS MATRICES OF ALL REFERENCED FILES
CDE/FILEACCESS/DACCESS.1	DECODE PERMISSION NUMBERS TO FILE FUNCTION NAMES
CDE/FILEACCESS/GIVEACCESS.2	SETS THE ACCESS MATRICES OF THE REFERENCED FILE; OR'S IT IF AN ENTRY FOR AN ACCOUNT ALREADY EXISTS
CDE/FILEACCESS/LISTACCESS.2	LISTS THE FILE ACCESS MATRICES IN A GRAPHICAL WAY
CDE/FILEACCESS/PASSNO.2	APPLY PASSNUMBER; CATENATES ENCLOSURE OF α AND ω
CDE/FILEACCESS/REMOVEACCESS.2	REMOVES ALL OR SPECIFIC FILE ACCESS FOR SELECTED ACCOUNTS
CDE/FILEACCESS/SETACCESS.2	SETS THE ACCESS MATRICES OF SELECTED FILES, REPLACES ENTRIES OF EXITSTING REFERENCED ACCOUNTS

CATEGORY 34 MEASURING TIME AND SPACE REQUIREMENTS

DJK/CPU.1	MEASURES THE NUMBER OF CPU UNITS REQUIRED TO EXECUTE EXPRESSIONS.
DJK/DATATYPE.1	DETERMINES THE INTERNAL DATA TYPE OF AN ARRAY (OR PACKAGE).
DJK/SPACE.1	CALCULATES AN UPPER BOUND (WITHIN 1K) ON THE BYTES REQUIRED TO EXECUTE AN EXPRESSION.
ROHAN/SIZES.2	RETURNS A FORMATTED TABLE OF THE NAMES AND SIZES OF THE OBJECTS SPECIFIED IN THE ARGUMENT.

- -

CATEGORY 35 WORKSPACE TOOLS

CDB/DOWSDOC.2	COVER FUNCTION FOR <WSDOC> FROM WORKSPACE <7 WSDOC>
DJK/FNSUMMARY.1	DISPLAYS SYNTAX AND LEADING FULL-LINE COMMENTS OF SPECIFIED FUNCTIONS.
HUI/RESET.1	RESETS STOP AND TRACE VECTORS OF FUNCTIONS.
JEW/WSCOMPARE.1	COMPARES TWO WORKSPACES.
LLF/LISTFNS.1	FORMATTED LISTING OF FUNCTIONS NAMED IN RIGHT ARGUMENT
LLF/LISTFNSPREFIX.1	LIST ALL FUNCTIONS IN THE WS WHOSE NAMES BEGIN WITH PREFIX IN RIGHT ARGUMENT
RMILL/SPINWSDOC.3	SUBMIT A WSDOC OF THE OBJECTS IN A PACKAGE
WGR/PACKSHOW.1	FUNCTION TO DISPLAY CONTENTS OF PACKAGE(INCLUDING FUNCTIONS AND OTHER PACKAGES) IN NEAT FORM

- -

CATEGORY 36 MEASURING USAGE AND CHARGES

CDB/QUOTA.1	RETURNS DISPLAY OF CURRENT USER'S QUOTAS.
DJK/CPU.1	MEASURES THE NUMBER OF CPU UNITS REQUIRED TO EXECUTE EXPRESSIONS.
ROHAN/COST.2	RETURNS A TABLE SIMILAR TO THAT DISPLAYED AT SIGNOFF, WITH THE ACTUAL DOLLAR COST ALSO GIVEN.

- -

CATEGORY 37 TIMES AND DATES

CDB/CALENDAR.2	RETURNS CALENDAR FOR YEAR AS 12 ELEMENT ENCLOSED VECTOR
DJK/FORMATTS.2	E.G. FORMATTS 1982 9 26 5 52 4 37 ↔ '1982-09-26 05:52:04.037'.
MRAB/FORMATΔTS.1	CONVERTS ⎕TS-STYLE TIMES INTO ISO-COMPATIBLE CHARACTER REPRESENTATIONS.
PCB/DATEDECODE.1	RETURNS CHARACTER ARRAY OF DATES IN STANDARD FORMAT, FROM ARRAY OF DAYNUMBERS.
PCB/DATEENCODE.1	RETURNS NUMERIC ARRAY OF JULIAN DAYNUMBERS, FROM CHARACTER ARRAY OF FORMATTED DATES.
PCB/DATEREP.1	RETURNS 3-ELEMENT REPRESENTATION OF EACH OF ARRAY OF JULIAN DAYNUMBERS
PCB/DNO.1	RETURNS JULIAN DAYNUMBERS FOR ARRAY OF DATES. LEFT ARG SPECIFIES DATE FORMAT.
PCB/RDCIDECODE.2	RETURNS CHARACTER ARRAY OF FORMATTED DATES AND TIMES FOR ⎕RDCI TIMESTAMPS
PCB/RDCIENCODE.1	RETURNS ⎕RDCI ENDCODINGS FOR ARRAY OF DATES AND TIMES IN ⎕TS FORMAT
ROHAN/DAY.2	RETURNS THE DAY OF THE WEEK ('MONDAY', 'TUESDAY', ETC.) THAT A DATE FALLS ON.
ROHAN/FORMATΔTIMESTAMP.1	PUTS THE GIVEN ⎕TS-STYLE TIMESTAMP INTO THE FORMAT <HH.MM.SS WWW DD MMM YYYY>.
TS/GETOFFSET.1	RETURNS AN OFFSET FROM UTC, IN SECONDS, FOR THE NODE SPECIFIED IN THE ARGUMENT.
TS/LTS.2	RETIRNS THE LOCAL TIMESTAMP WHEN GIVEN THE TIMESTAMP IN UTC.
TS/UTC.2	RETURNS A UTC TIMESTAMP FROM AN ARGUMENT IN LOCAL TIMESTAMP.
WGR/FTT.2	FORMS FORMATTED REPRESENTATION OF TIMESTAMPS FROM ⎕RDCI FORM.ALLOWS VECTOR INPUT
WGR/TTF.1	FORMS ⎕RDCI[3] FORMAT ENCODED TIMESTAMP VECTOR FROM MATRIX OF ⎕TS FORM TIMESTAMPS

CATEGORY 38 BUSINESS AND FINANCE — CATEGORY 38

CDB/ACC∆ADVANCE.1	ACCUMULATE PERIODIC PAYMENTS IN ADVANCE AT GIVEN INTEREST RATES
CDB/ACC∆ARREARS.1	ACCUMULATE PERIODIC PAYMENTS IN ARREARS AT GIVEN INTEREST RATES
CDB/AMORT.1	GENERATES A LEVEL PAYMENT AMORTIZATION TABLE
CDB/INTEREST.1	ALLOWS EASY SPECIFICATION OF VARYING INTEREST RATES
HUI/DEPRECIATE.1	STRAIGHT-LINE DEPRECIATION.
HUI/MAVG.1	COMPUTES THE K-STEP MOVING AVERAGE ON THE LAST AXIS OF AN ARRAY.
ROHAN/SUBTOTAL.1	INSERTS FIRST-AXIS SUBTOTALS INTO AN ARRAY.
SAUCE/ALLOCEQ.1	ALLOCATES A NUMBER EQUITABLY SUBJECT TO LIMITS. E.G. 29 ALLOCEQ 3 20 500 7 ↔ 3 9.5 9.5 7.
SAUCE/ALLOCFIFO.1	ALLOCATES A NUMBER ON A FIFO BASIS, SUBJECT TO LIMITS. E.G. 9 ALLOCFIFO 4 6 11 ↔ 4 5 0.
SAUCE/DISTROUND.1	DISTRIBUTIVE ROUNDING. E.G. 0.01 DISTROUND 2 6 ρ÷ 1 3 ↔ 2 6 ρ 1 0.34 1 0.33 1 0.33.

- -

CATEGORY 39 STATISTICS AND PROBABILITY — CATEGORY 39

ANO/CHOLESKY.1	CHOLESKY FACTORIZATION (DECOMPOSITION) OF A MATRIX
ANO/KRONECKER.1	∇KRONECKER CALCULATES THE KRONECKER, OR DIRECT, PRODUCT OF TWO MATRICES.
ANO/PENROSE.1	∇PENROSE CALCULATES THE GENERALIZED INVERSE OF A MATRIX
ANO/POLYDIV.1	RETURNS THE QUOTIENT OF TWO (VECTOR) POLYNOMIALS
ANO/POLYMULT.1	RETURNS THE PRODUCT OF TWO (VECTOR) POLYNOMIALS
DJK/RANDOMIZE.1	FINDS A 'RANDOM' VALUE FOR ⎕RL, USING 2↓⎕TS.
HUI/CLASSIFY.1	INPUT: RANGES (INTERVALS) AND SOME NUMBERS. OUTPUT: WHICH INTERVAL EACH NUMBER IS IN.
HUI/MAVG.1	COMPUTES THE K-STEP MOVING AVERAGE ON THE LAST AXIS OF AN ARRAY.
SAUCE/DISTROUND.1	DISTRIBUTIVE ROUNDING. E.G. 0.01 DISTROUND 2 6 ρ÷ 1 3 ↔ 2 6 ρ 1 0.34 1 0.33 1 0.33.
SAUCE/FREQDIST.1	RETURNS THE FREQUENCY DISTRIBUTION, IN SPECIFIED CLASSES, OF NUMERIC DATA.
SAUCE/RUNLENGTHS.1	E.G. RUNLENGTHS 1 9 9 4 4 4 9 ↔ 1 2 3 1.

DJK/INROWS.1	FINDS WHICH ROWS OF A MATRIX CONTAIN A STRING ANYWHERE WITHIN THEM.
DJK/RANDOMIZE.1	FINDS A 'RANDOM' VALUE FOR ⎕RL, USING 2↓⎕TS.
HUI/CLASSIFY.1	INPUT: RANGES (INTERVALS) AND SOME NUMBERS. OUTPUT: WHICH INTERVAL EACH NUMBER IS IN.
LHG/SS.1	GENERAL STRING SEARCH PRIMITIVE
MTH/UNIQUEROWS.1	FINDS ALL UNIQUE ROWS IN A TWO-DIMENSIONAL CHARACTER MATRIX
PCB/NOTEMPTY.1	BOOLEAN WITH 1 WHERE FIRST AXIS CONTAINS NON-ZERO OR NON-BLANK
PCB/NUB.1	ELIMINATE DUPLICATES
SAUCE/DISTINCT.1	REMOVES DUPLICATE ELEMENTS FROM A VECTOR. E.G. DISTINCT 3 5 3 4 ↔ 3 5 4.
SAUCE/DISTINCTROWS.1	REMOVES DUPLICATE ROWS FROM MATRIX, E.G. DISTINCTROWS 4 2⍴'AB','CD','AB','AD' ↔ 3 2⍴'AB','CD','AD'.
SAUCE/EXTENDPARTS.1	FOR EXTENDING PARTS. E.G. 1 1 0 0 1 0 0 EXTENDPARTS 1 0 0 0 1 1 0 ↔ 1 0 0 0 1 0 0 1 0 0 0.
SAUCE/FIRSTONES.1	SETS TO 0 ALL BUT THE FIRST 1 IN EACH SEQUENCE OF 1'S IN A BOOLEAN VECTOR.
SAUCE/FIRSTZEROES.1	SETS TO 1 ALL BUT THE FIRST 0 IN EACH SEQUENCE OF 0'S IN A BOOLEAN VECTOR.
SAUCE/FROM.1	A VARIANT OF INDEXING. EACH ROW OF THE LEFT ARGUMENT SELECTS ONE ELEMENT FROM THE RIGHT ARGUMENT.
SAUCE/FROMTO.1	E.G. 1 157 ¯12 FROMTO 2 161 ¯10 ↔ 1 2 157 158 159 160 161 ¯12 ¯11 ¯10.
SAUCE/GEROWS.1	DETERMINES WHICH ROWS OF A MATRIX ARE ≥ (IN THE SENSE OF ⍋) A VECTOR.
SAUCE/GTROWS.1	DETERMINES WHICH ROWS OF A MATRIX ARE > (IN THE SENSE OF ⍋) A VECTOR.
SAUCE/INDEXGEN.1	THE ARG SHOULD BE A VECTOR. RETURNS (⍳1↑VECTOR),(⍳1↑1↓VECTOR),(⍳1↑2↓VECTOR),...,⍳¯1↑VECTOR.
SAUCE/INDEXOFROWS.2	RETURNS THE 'INDEX OF' EACH ROW OF ONE MATRIX IN ANOTHER MATRIX.
SAUCE/LASTONES.1	SETS TO 0 ALL BUT THE LAST 1 IN EACH SEQUENCE OF 1'S IN A BOOLEAN VECTOR.
SAUCE/LASTZEROES.1	SETS TO 1 ALL BUT THE LAST 0 IN EACH SEQUENCE OF 0'S IN A BOOLEAN VECTOR.
SAUCE/MASKPARTS.1	E.G. 1 0 1 0 MASKPARTS 1 0 0 1 0 1 0 1 0 ↔ 1 1 1 0 0 1 1 0 0.
SAUCE/MEMBERROWS.1	RETURNS A BOOLEAN ARRAY INDICATING WHETHER EACH ROW OF ONE ARG IS A ROW OF THE OTHER ARG.
SAUCE/MESH.1	MESHES CATENATED ARRAYS. E.G. 1 2 3 3 2 1 MESH 2 6 ⍴'AD','BE','CF' ↔ 2 6 ⍴'ABCDEF'.
SAUCE/NOTMEMBERROWS.1	RETURNS A BOOLEAN ARRAY INDICATING WHETHER EACH ROW OF ONE ARG IS A NOT A ROW OF THE OTHER ARG.
SAUCE/PROINDEXOF.1	PROGRESSIVE DYADIC IOTA, E.G. 'ABA' PROINDEXOF 'ACAABA' ↔ ⎕IO+ 0 3 2 3 1 3.
SAUCE/REMOVETRAIL.1	REMOVES TRAILING COLUMNS OR ELEMENTS FROM AN ARRAY.
SAUCE/SQUEEZE.1	REMOVES ALL LEADING, TRAILING, AND REDUNDANT ELEMENTS (SPEC'D IN THE LEFT ARGUMENT) FROM A VECTOR.
SAUCE/STRINGFIND.2	FINDS ALL OCCURRENCES OF ONE VECTOR IN ANOTHER.
YUDI/FILL.1	FILLS A CHARACTER STRING WITH A DELIMITER FOR A SPECIFIED NUMBER OF TIMES

CATEGORY 41 BOOLEAN ARRAYS

CATEGORY 41

MGF/TOPOSORT.1	TOPOLOGICAL SORT OF PRECEDENCE MATRIX.
PCB/NOTEMPTY.1	BOOLEAN WITH 1 WHERE FIRST AXIS CONTAINS NON-ZERO OR NON-BLANK
SAUCE/EXTENDPARTS.1	FOR EXTENDING PARTS. E.G. 1 1 0 0 1 0 0 EXTENDPARTS 1 0 0 0 1 1 0 ↔ 1 0 0 0 1 0 0 1 0 0 0.
SAUCE/FIRSTONES.1	SETS TO 0 ALL BUT THE FIRST 1 IN EACH SEQUENCE OF 1'S IN A BOOLEAN VECTOR.
SAUCE/FIRSTZEROES.1	SETS TO 1 ALL BUT THE FIRST 0 IN EACH SEQUENCE OF 0'S IN A BOOLEAN VECTOR.
SAUCE/LASTONES.1	SETS TO 0 ALL BUT THE LAST 1 IN EACH SEQUENCE OF 1'S IN A BOOLEAN VECTOR.
SAUCE/LASTZEROES.1	SETS TO 1 ALL BUT THE LAST 0 IN EACH SEQUENCE OF 0'S IN A BOOLEAN VECTOR.
SAUCE/MASKPARTS.1	E.G. 1 0 1 0 MASKPARTS 1 0 0 1 0 1 0 1 0 ↔ 1 1 1 0 0 1 1 0 0.
SAUCE/MEMBERROWS.1	RETURNS A BOOLEAN ARRAY INDICATING WHETHER EACH ROW OF ONE ARG IS A ROW OF THE OTHER ARG.
SAUCE/NOTMEMBERROWS.1	RETURNS A BOOLEAN ARRAY INDICATING WHETHER EACH ROW OF ONE ARG IS A NOT A ROW OF THE OTHER ARG.
SAUCE/PANDREDUCE.1	PARTITIONED ∧/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PANDSCAN.1	PARTITIONED ∧\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PEQSCAN.1	PARTITIONED =\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PLESCAN.1	PARTITIONED ≤\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PLTSCAN.1	PARTITIONED <\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PNESCAN.1	PARTITIONED ≠\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PORREDUCE.1	PARTITIONED ∨/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.
SAUCE/PORSCAN.1	PARTITIONED ∨\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

(THERE ARE NO UTILITIES IN CATEGORY 42.)

- -

CATEGORY 43 FILE PRIMITIVE SIMULATION *CATEGORY 43*

DJK/FDELETE.1	DELETES SPECIFIED COMPONENTS FROM A FILE.
PESCH/FILE.1	ALLOWS EXPRESSIONS WITH FILE PRIMITIVES REFERRING TO FILES BY NAME
PESCH/TIE.1	CREATES OR FINDS TIE NUMBER GIVEN FILE NAME; AVOIDS EXHAUSTING FILE TIE QUOTA
RCM/FILES/FAPPEND.1	COVER FUNCTION FOR ⎕APPEND- RESIZES FILE IF NECESSARY
RCM/FILES/FAPPENDR.1	COVER FUNCTION FOR ⎕APPENDR- RESIZES FILE IF NECESSARY
RCM/FILES/FOPEN.2	SHARE TIES A FILE, AND CREATES IT IF NECESSARY
RCM/FILES/FREPLACE.1	COVER FUNCTION FOR ⎕REPLACE- RESIZES FILE IF NECESSARY
RCM/FILES/FRESIZE.1	COVER FUNCTION FOR ⎕RESIZE- IF ASKING TOO MUCH, GETS AS MUCH AS FILE RES ALLOWS
RCM/FILES/FWRITE.1	WRITES TO SPECIFIED COMPONENT. IF COMP DOESN'T EXIST, CREATES IT AND ANY NECESSARY PRECEDING COMPS.
RMILL/FCOMPCOPY.1	NEARLY WSFULL PROOF COPY ONE FILE COMPONENT TO ANOTHER
ROHAN/EASYTIE.2	TIES A FILE AND RETURNS THE TIE NUMBER, WITH PROVISION FOR LATER UNTYING A NEWLY-TIED FILE.
ROHAN/EASYUNTIE.1	UNTIES A FILE TIED BY ROHAN/EASYTIE.

- -

CATEGORY 44 EVENT TRAPPING *CATEGORY 44*

RCM/EVENTTRAP/ERRORLOG.1	AUTOMATICALLY LOGS IN A FILE ALL ERRORS NOT OTHERWISE TRAPPED
RCM/EVENTTRAP/EVENTΔREPORT.1	ANALYZES ⎕ER AND RETURNS (EVENT CODE),(INDEX OF BAD CHAR IN STMT),(⎕AV INDICES OF CHARS IN STMT).
RCM/EVENTTRAP/INTERPRET.1	ARGUMENT IS ⎕ER. EXPLICIT RESULT IS AN ERROR MESSAGE SUITABLE FOR A NAIVE (NON-PROGRAMMER) USER.
RCM/EVENTTRAP/SETUPΔERRORLOG.1	SETS UP WS AND FILE SO THAT ALL UNTRAPPED EVENTS MAY BE AUTOMATICALLY TRAPPED AND LOGGED IN A FILE

- -

CATEGORY 45 GRAPHICS AND PLOTTING *CATEGORY 45*

PKI/ADMPRINT.1	USES ADMPRINT TO QUEUE A PRINT REQUEST, FOR PROCESSING ON A 3279.
PKI/FSCLS.1	CANCELS OR ENQUEUES A FILE FOR PRINTING BY ADMPRINT.
PKI/FSLOG.1	ADDS TEXT TO QUEUED PRINTER FILE OPENED USING FSOPEN.
PKI/FSOPEN.1	OPEN QUEUED PRINTING DEVICE FOR OUTPUT (USUALLY IBM 3287) USING AP126.
PKI/GDDM.1	UTILITY TO PERFORM AP126 CALLS, WITH ERROR CHECKING.
PKI/SHADEAT.1	SHADES LOCUS OF POINTS AT SPECIFIED X LOCATIONS.

- -

CATEGORY 46 MODIFYING ARRAYS *CATEGORY 46*

HUI/APLTOCOURIER.1	CONVERTS TEXT IN APL FONT TO COURIER FONT.
HUI/COURIERTOAPL.1	CONVERTS TEXT IN COURIER FONT TO APL FONT.
LHG/ΔRPLC.1	GENERAL REPLACE OF ONE STRING BY ANOTHER.
SAUCE/ELEMREPLACE.1	REPLACES, IN A VECTOR CONTAINING NO ENCLOSURES, ALL OCCURRENCES OF A SCALAR BY A (DIFFERENT) VECTOR.

PKI/ADMPRINT.1	USES ADMPRINT TO QUEUE A PRINT REQUEST, FOR PROCESSING ON A 3279.
PKI/FSCLS.1	CANCELS OR ENQUEUES A FILE FOR PRINTING BY ADMPRINT.
PKI/FSLOG.1	ADDS TEXT TO QUEUED PRINTER FILE OPENED USING FSOPEN.
PKI/FSOPEN.1	OPEN QUEUED PRINTING DEVICE FOR OUTPUT (USUALLY IBM 3287) USING AP126.
PKI/GDDM.1	UTILITY TO PERFORM AP126 CALLS, WITH ERROR CHECKING.

TITLE: ANO/CHOLESKY.1

TYPE: FUNCTION

SUMMARY: CHOLESKY FACTORIZATION (DECOMPOSITION) OF A MATRIX

TIMESTAMP: 1983-11-28 19:39:20

CATEGORIES:
5 NUMERIC CALCULATION
39 STATISTICS AND PROBABILITY

DESCRIPTION:

SYNTAX: P←CHOLESKY OMEGA

ARGUMENT: A POSITIVE DEFINITE MATRIX OMEGA (SUCH AS, FOR EXAMPLE, A VARIANCE-COVARIANCE MATRIX OR A CORRELATION MATRIX)

RESULT: A NONSINGULAR MATRIX P SUCH THAT OMEGA←P+.×⍉P

∇CHOLESKY PERFORMS THE CHOLESKY FACTORIZATION, OR CHOLESKY DECOMPOSITION, OF A MATRIX. A COMMON MATRIX ALGEBRA CALCULATION, MATRIX FACTORIZATION HAS NUMEROUS USES IN ECONOMETRICS, ESPECIALLY IN THE APPLICATION OF GENERALIZED LEAST SQUARES AND ITS VARIOUS SPECIAL CASES (SUCH AS WEIGHTED LEAST SQUARES AND AUTOCORRELATION ADJUSTMENTS).

THE FUNCTION ORIGINATED IN IBM'S OLD STATPACK, AND WAS WRITTEN BY JACK PRINS, WHO SUPPORTED STATPACK. CHOLESKY HAS BEEN USED IN THE WORKSPACE 32 REGRESSION SINCE 1975, AND IN EASY SINCE 1979.

RESTARTABLE. ⎕IO LOCALIZED. NO VALIDATION IS PERFORMED. (FOR EXAMPLE, THE ARGUMENT OMEGA IS ASSUMED TO BE POSITIVE DEFINITE.)

TITLE: ANO/KRONECKER.1

TYPE: FUNCTION

SUMMARY: ∇KRONECKER CALCULATES THE KRONECKER, OR DIRECT, PRODUCT OF TWO MATRICES.

TIMESTAMP: 1984-03-02 19:17:32

CATEGORIES:
5 NUMERIC CALCULATION
39 STATISTICS AND PROBABILITY

DESCRIPTION:
∇KRONECKER CALCULATES THE KRONECKER PRODUCT, OR DIRECT PRODUCT, OF 2 MATRICES.
THE KRONECKER PRODUCT OF A MATRIX A (OF SHAPE M×N) AND A MATRIX B (OF SHAPE P×Q) IS A MATRIX C OF SHAPE MP×NQ SUCH THAT C[I;J] IS THE SCALAR PRODUCT OF A[I;J] AND THE ENTIRE MATRIX B (FOR I=1,...,M AND J=1,...,N).
SYNTAX: R←A KRONECKER B
ARGUMENTS: ANY NUMERIC ARRAYS A AND B OF RANK TWO OR LESS. NORMALLY BOTH ARGUMENT ARRAYS ARE OF RANK 2.
RESULT: A RANK-TWO ARRAY, WITH (1↑ρA)×(1↑ρB) ROWS AND (1↓ρA)×(1↓ρB) COLUMNS (SEE NOTE 2 BELOW), CONTAINING THE DIRECT PRODUCT OF THE ARRAYS A AND B.
EXAMPLE:

```
      ρZ←1 KRONECKER 2 2ρ⍳4
2 2
      Z
1 2
3 4
      ρZ←(2 2ρ⍳4) KRONECKER 3 3ρ⍳9
6 6
      Z
 1  2  3  2  4  6
 4  5  6  8 10 12
 7  8  9 14 16 18
 3  6  9  4  8 12
12 15 18 16 20 24
21 24 27 28 32 36
```

NOTES:
1) ⎕IO LOCALIZED.
2) SCALAR AND VECTOR ARGUMENTS ARE RESHAPED INTO MATRICES WITH ONE ROW.

TITLE: ANO/PENROSE.1

TYPE: FUNCTION

SUMMARY: ∇PENROSE CALCULATES THE GENERALIZED INVERSE OF A MATRIX

TIMESTAMP: 1984-02-25 15:45:50

CATEGORIES: 5 NUMERIC CALCULATION
39 STATISTICS AND PROBABILITY

DESCRIPTION:
∇PENROSE CALCULATES THE GENERALIZED INVERSE (ALSO CALLED THE MOORE-PENROSE INVERSE) OF A MATRIX. IF A IS ANY M×N MATRIX OF RANK R (NOT APL RANK!) THEN ITS UNIQUE GENERALIZED INVERSE G, ALSO OF RANK R, IS OF SHAPE N×M. G SATISFIES THESE FOUR CONDITIONS:

```
A+.×G+.×A ↔ A
G+.×A+.×G ↔ G
⍉A+.×G ↔ A+.×G
⍉G+.×A ↔ G+.×A
```

SYNTAX: G←PENROSE A
ARGUMENT: ANY (POSSIBLY SINGULAR) ARRAY A SUCH THAT 2=⍴⍴A. A NEED NOT BE SQUARE. NOR MUST A HAVE MORE ROWS THAN COLUMNS.
RESULT: THE GENERALIZED INVERSE MATRIX G OF A, SUCH THAT (⍴G)≡⌽⍴A, AND G SATISFIES THE 4 CONDITIONS NOTED ABOVE.
EXAMPLE:

```
      M
 1  87  25 100
 1  54  46 100
 1  75  91 100
 1  31  30 100
 1   7  57 100
      -.×M
0
      G←PENROSE M
      ⍴M
5 4
      ⍴G
4 5
      ∧/,1E¯12>|M-M+.×G+.×M
1
      ∧/,1E¯12>|G-G+.×M+.×G
1
      ∧/,1E¯12>|(M+.×G)-⍉M+.×G
1
      ∧/,1E¯12>|(G+.×M)-⍉G+.×M
1
```

SOURCE: APL QUOTE QUAD, VOLUME 10 NUMBER 2, DECEMBER 1979, P. 30.
NOTES:
1) ⎕IO LOCALIZED. NO VALIDATION OF ARGUMENT A.
2) ∇PENROSE CALLS A MONADIC, EXPLICIT-RESULT SUBFUNCTION ∇BASIS, WHICH CALCULATES A COLUMN BASIS FOR A. A CONVERGENCE TOLERANCE EMPLOYED BY ∇BASIS IS CONTAINED IN A GLOBAL VARIABLE EPS. EPS IS ASSIGNED (LOCALLY) THE VALUE 1E¯15 IF NON-EXISTENT.

TITLE: ANO/POLYDIV.1

TYPE: FUNCTION

SUMMARY: RETURNS THE QUOTIENT OF TWO (VECTOR) POLYNOMIALS

TIMESTAMP: 1984-01-25 17:53:35

CATEGORIES:
5 NUMERIC CALCULATION
39 STATISTICS AND PROBABILITY

DESCRIPTION:
∇POLYDIV CALCULATES THE QUOTIENT OF 2 POLYNOMIALS. IT IS THE INVERSE OF THE UTILITY ANO/POLYMULT.1.
SYNTAX: R←A POLYDIV B
ARGUMENTS: TWO VECTORS, EACH REPRESENTING THE COEFFICIENTS OF A POLYNOMIAL OF ANY DEGREE. A POLYNOMIAL OF DEGREE D IS REPRESENTED BY A VECTOR OF LENGTH D+1, THE ELEMENTS OF THE VECTOR BEING THE COEFFICIENTS ASSOCIATED WITH EXPONENTS 0 THROUGH D RESPECTIVELY. ZERO COEFFICIENTS MUST BE INCLUDED. FOR EXAMPLE, ¯1+X*4 WOULD BE REPRESENTED AS THE VECTOR ¯1 0 0 0 1.
RESULT: A VECTOR, OF THE SAME FORMAT AS THAT OF THE TWO ARGUMENT VECTORS, REPRESENTING THE QUOTIENT POLYNOMIAL. ITS LENGTH IS DETERMINED BY THE VALUE OF THE GLOBAL VARIABLE <NUMBEROFTERMS>. SINCE THE RESULT OF A POLYNOMIAL DIVIDE CAN HAVE AN INFINITE NUMBER OF TERMS, A VALUE FOR <NUMBEROFTERMS> IS NEEDED TO TRUNCATE THE DIVISION. IF A VARIABLE <NUMBEROFTERMS> DOES NOT EXIST, ∇POLYDIV RETURNS A RESULT OF LENGTH 50.
EXAMPLES:

```
      NUMBEROFTERMS←10
      1 ¯2 1 POLYDIV 1 ¯1
1 ¯1 0 0 0 0 0 0 0 0
      NUMBEROFTERMS←2
      1 ¯2 1 POLYDIV 1 ¯1
1 ¯1
      1 1.3 .4 POLYDIV 1 .8
1 0.5
      R←⎕EX 'NUMBEROFTERMS'
      1 POLYDIV 1,¯12↑.5
1 0 0 0 0 0 0 0 0 0 0 0 ¯0.5 0 0 0 0 0 0 0 0 0 0 0 0 0.25 0 0 0 0 0 0 0 0 0 0 0
      ¯0.125 0 0 0 0 0 0 0 0 0 0 0 0 0.0625 0
```

*** EXECUTE DETAILS UDESCRIBE 'ANO/POLYDIV.1' FOR MORE INFORMATION

TITLE: ANO/POLYMULT.1

TYPE: FUNCTION

SUMMARY: RETURNS THE PRODUCT OF TWO (VECTOR) POLYNOMIALS

TIMESTAMP: 1983-12-23 15:47:56

CATEGORIES: 5 NUMERIC CALCULATION
39 STATISTICS AND PROBABILITY

DESCRIPTION:

```
     ∇POLYMULT CALCULATES THE PRODUCT OF 2 POLYNOMIALS.
SYNTAX: R←A POLYMULT B
ARGUMENTS: TWO VECTORS, EACH REPRESENTING THE COEFFICIENTS OF A POLYNOMIAL OF
           ANY DEGREE. A POLYNOMIAL OF DEGREE D IS REPRESENTED BY A VECTOR OF
           LENGTH D+1, THE ELEMENTS OF THE VECTOR BEING THE COEFFICIENTS
           ASSOCIATED WITH EXPONENTS 0 THROUGH D RESPECTIVELY. ZERO
           COEFFICIENTS MUST BE INCLUDED. FOR EXAMPLE, ¯1+X*4 WOULD BE
           REPRESENTED AS THE VECTOR ¯1 0 0 0 1.
RESULT: A VECTOR, OF THE SAME FORMAT AS THAT OF THE 2 ARGUMENT VECTORS,
        REPRESENTING THE PRODUCT POLYNOMIAL. ITS LENGTH IS ALWAYS ¯1+(ρA)+ρB.
EXAMPLES:
      1 .8 POLYMULT 1 .5
1 1.3 0.4
      1 ¯1 POLYMULT 1 ¯1
1 ¯2 1
      1 ¯1 POLYMULT 1,¯12↑¯1
1 ¯1 0 0 0 0 0 0 0 0 0 0 ¯1 1
SOURCE: A GIFT FROM WGR, THIS FUNCTION ORIGINATED IN A QUOTE-QUAD ARTICLE BY
        JACK PRINS OF IBM. SINCE THE ARIMA MODEL METHODOLOGY OF BOX AND JENKINS
        REPRESENTS ALL COMPONENTS OF A TIME SERIES MODEL AS POLYNOMIALS IN A
        BACKSHIFT OPERATOR, POLYMULT AND ITS INVERSE FUNCTION POLYDIV ARE THE
        BASIS OF THE SHARP APL IMPLEMENTATION OF BOX-JENKINS.
NOTE: ⎕IO LOCALIZED. ARGUMENTS ARE RAVELLED, BUT NOT OTHERWISE VALIDATED.
```

- -

TITLE: BAP/PRINTTABLE.1

TYPE: FUNCTION

SUMMARY: FORMATS A CHARACTER STRING TO FIT ON PAGE LIKE)FNS

TIMESTAMP: 1983-11-17 17:02:56

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
25 FORMATTING
30 REPORT FORMATTING

DESCRIPTION:

```
FORMS A CHARACTER MATRIX RESULT WITH NUMBER OF COLUMNS ≤⎕PW.
THE INPUT CHARACTER MATRIX IS CUT AND SPLICED SO THAT THEMINIMUM NUMBER OF ROWS
FOR RESULT IS NEEDED.
   LEFT ARGUMENT IS TITLE(PUT ON EACH SUPERCOLUMN)
   RIGHT ARGUMENT IS CHARACTER MATRIX INPUT.
```

TITLE: CDB/ACC∆ADVANCE.1 CDB/ACC∆ADVANCE.1

TYPE: FUNCTION

SUMMARY: ACCUMULATE PERIODIC PAYMENTS IN ADVANCE AT GIVEN INTEREST RATES

TIMESTAMP: 1983-11-15 16:23:48

CATEGORIES: 38 BUSINESS AND FINANCE

DESCRIPTION:
RETURNS THE ACCUMULATED BALANCES AT THE END OF EACH TIME PERIOD, OF PERIODIC PAYMENTS MADE IN ADVANCE AT GIVEN INTEREST RATES.

```
RIGHT ARGUMENT ↔ SERIES OF PAYMENTS
LEFT ARGUMENT ↔ DECIMAL INTEREST RATES. THESE MAY OPTIONALLY INCLUDE INTEGER
                 REPLICATION FACTORS FOLLOWING EACH RATE.
RESULT  ↔ THE ACCUMULATED BALANCES.
```

EACH ARGUMENT IS STRETCHED IF NECESSARY TO MATCH THE LENGTH OF THE LONGEST ARGUMENT, BY REPLICATING THE LAST ELEMENT.

FOR EXAMPLE, TO FIND THE ACCUMULATED BALANCES OF A SERIES OF 10 PAYMENTS OF 100 PAYABLE IN ADVANCE, ASSUMING INTEREST OF .065 FOR 5 YEARS, AND .05 THEREAFTER:

```
.065 5 .05 ACC∆ADVANCE 10⍴100
```

- -

TITLE: CDB/ACC∆ARREARS.1 CDB/ACC∆ARREARS.1

TYPE: FUNCTION

SUMMARY: ACCUMULATE PERIODIC PAYMENTS IN ARREARS AT GIVEN INTEREST RATES

TIMESTAMP: 1983-11-15 16:30:26

CATEGORIES: 38 BUSINESS AND FINANCE

DESCRIPTION:
RETURNS THE ACCUMULATED BALANCES AT THE END OF EACH TIME PERIOD, OF PERIODIC PAYMENTS MADE IN ARREARS AT GIVEN INTEREST RATES.

```
RIGHT ARGUMENT ↔ SERIES OF PAYMENTS.
LEFT ARGUMENT ↔ DECIMAL INTEREST RATES. THESE MAY OPTIONALLY INCLUDE INTEGER
                 REPLICATION FACTORS FOLLOWING EACH RATE. NOTE, THE FIRST RATE
                 IS NOT SIGNIFICANT.
RESULT ↔ THE ACCUMULATED BALANCES.
```

EACH ARGUMENT IS STRETCHED IF NECESSARY TO MATCH THE LENGTH OF THE LONGEST ARGUMENT, BY REPLICATING THE LAST ELEMENT.

FOR EXAMPLE, TO FIND THE ACCUMULATED BALANCES OF A SERIES OF 10 PAYMENTS OF 100 PAYABLE IN ARREARS, ASSUMING INTEREST OF .065 FOR 5 YEARS, AND .05 THEREAFTER:

```
.065 5 .05 ACC∆ARREARS 10⍴100
```

TITLE: CDB/AMORT.1

TYPE: FUNCTION

SUMMARY: GENERATES A LEVEL PAYMENT AMORTIZATION TABLE

TIMESTAMP: 1983-11-15 15:53:55

CATEGORIES: 38 BUSINESS AND FINANCE

DESCRIPTION:

```
GENERATES A LEVEL PAYMENT AMORTIZATION TABLE.
LEFT ARGUMENT ↔ OPENING BALANCE
RIGHT ARGUMENT ↔ DECIMAL EFFECTIVE INTEREST RATES PER PERIOD, OF LENGTH
                 EQUAL TO THE NUMBER OF PERIODS
RESULT ↔ (LEVEL SINGLE PAYMENT)⊃AMORTIZATION TABLE, WHERE THE TABLE HAS 3
         COLUMNS: OUTSTANDING BALANCE AT BEGINNING OF EACH PERIOD
                  INTEREST PORTION OF PAYMENT
                  CAPITAL PORTION OF PAYMENT
```

PAYMENTS ARE ASSUMED MADE IN ARREARS

EG: 10000 AMORT 10⍴.085 ⋄ 10000 AMORT (5⍴.085),5⍴.08

- -

TITLE: CDB/CALENDAR.2

TYPE: FUNCTION

SUMMARY: RETURNS CALENDAR FOR YEAR AS 12 ELEMENT ENCLOSED VECTOR

TIMESTAMP: 1984-01-25 23:58:39

CATEGORIES: 37 TIMES AND DATES

DESCRIPTION:
RETURNS CALENDAR FOR GIVEN YEAR AS A 12 ELEMENT VECTOR; EACH ELEMENT IS AN ENCLOSED CHARACTER MATRIX, APPROPRIATELY FORMATTED. THE RIGHT ARGUMENT IS THE YEAR (ONLY THE LAST 2 DIGITS ARE NEEDED IF IN THIS CENTURY). FOR EXAMPLE, THE CALENDAR FOR JANUARY AND FEBRUARY 1984 IS:

```
      (CALENDAR 84)[⍳2]
          JAN                          FEB
 SU MO TU WE TH FR SA   SU MO TU WE TH FR SA
  1  2  3  4  5  6  7             1  2  3  4
  8  9 10 11 12 13 14    5  6  7  8  9 10 11
 15 16 17 18 19 20 21   12 13 14 15 16 17 18
 22 23 24 25 26 27 28   19 20 21 22 23 24 25
 29 30 31               26 27 28 29
```

TITLE: CDB/DOWSDOC.2

TYPE: FUNCTION

SUMMARY: COVER FUNCTION FOR <WSDOC> FROM WORKSPACE <7 WSDOC>

TIMESTAMP: 1984-06-24 11:08:18

CATEGORIES: 35 WORKSPACE TOOLS ⍝ E.G. WORKSPACE CROSS REFERENCE

DESCRIPTION:
SYNTAX: R←WSS DOWSDOC DELIVERY

DOCUMENT WORKSPACE(S) USING THE <WSDOC> FUNCTION FROM <7 WSDOC> AND SUBMIT A HSPRINT. THE STATE SETTINGS USED ARE LISTED IN THE FUNCTION.

<WSS> MAY BE A LIST OR MATRIX OF WORKSPACE NAMES. IF NOT GIVEN, THE CURRENT WORKSPACE IS USED. IF NO ACCOUNT NUMBER IS GIVEN, THE CURRENT ACCOUNT IS USED.

<DELIVERY> IS THE HSPRINT PRINT SPECS (OPTIONAL), AND DELIVERY INSTRUCTIONS. THE ONLY NON-DEFAULT PRINT SPEC IS: ERAS. <DELIVERY> IS ASSUMED TO BE VALID; THIS FUNCTION PERFORMS NO VALIDATION.

<DOWSDOC> REQUIRES THE CURRENT ACCOUNT TO BE MULTI-TASKED, AND USES THE WORKSPACES <7 WSTOFILE>, <7 WSDOC> AND <5 STASK>.

EG: '1 HSPRINT 5 AP124' DOWSDOC 'DO2⍝DELIVER TO JOHN DOE, IPSA YYZ19 THANKS'

TITLE: *CDB/FILECONTENTS.2*

TYPE: *FUNCTION*

SUMMARY: *DISPLAYS CONTENTS OF A FILE*

TIMESTAMP: 1984-06-09 11:16:42

CATEGORIES: 9 *FILES*

DESCRIPTION:
SYNTAX: (*PAUSE*) *FILECONTENTS FILERANGE*

PRINTS CONTENTS OF A FILE, LISTING VARIABLE COMPONENTS IN FULL, AND THE □*PNAMES OF PACKAGED COMPONENTS. ONLY THE COMPONENT NUMBER IS LISTED FOR EMPTY COMPONENTS.*

<FILERANGE> MAY BE A FILE NAME OR TIE NUMBER <FILE>, OR A LINKED PAIR: <FILE>⊃<RANGE>, OR A NUMERIC VECTOR: <FILE>,<RANGE> , WHERE:

<FILE> *IS A FILE NAME OR TIE NUMBER*
<RANGE> *IS OPTIONAL AND DETERMINES WHICH COMPONENTS ARE TO BE READ. IF NOT GIVEN, ALL ARE READ, OTHERWISE <RANGE> IS A* 1 *OR* 2 *ELEMENT VECTOR, AND THE FILE IS READ STARTING FROM* 1↑*RANGE, AND (OPTIONALLY) ENDING AT* 1↑*RANGE*

<PAUSE> IS OPTIONAL AND DETERMINES WHETHER THERE IS A PAUSE BETWEEN NON-EMPTY COMPONENTS. IF NOT GIVEN THERE IS NO PAUSE. IF PAUSE=1, *THEN YOU MUST ENTER <CARRIAGE RETURN> AFTER EACH COMPONENT IS DISPLAYED.*

TITLE: CDB/FNCOMPARE.1

TYPE: FUNCTION

SUMMARY: COMPARE TWO FUNCTIONS

TIMESTAMP: 1984-08-09 02:49:30

CATEGORIES: 28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
SYNTAX: R← FN1 FNCOMPARE FN2

COMPARES THE ⎕CR REPRESENTATIONS OF FUNCTIONS, RETURNING 'NO DIFFERENCE' OR (VECTOR) TEXT OF MISMATCHES AND SWAPPED LINES. ARGUMENTS MAY BE EITHER (NON-MATRIX) FUNCTION NAMES, OR ⎕CR'S OF FUNCTIONS. FOR EXAMPLE:

```
        ∇ FOO                         ∇ GOO
     [1]  ⍝ TEST <FNCOMPARE>     [1]   LINEB
     [2]   LINEA                 [2]   LINEC
     [3]   LINEB                 [3]   LINEA
     [4]   LINEC                      ∇
        ∇

     'FOO'FNCOMPARE 'GOO'
1 [0] FOO
2 [0] GOO
1 [1] ⍝ TEST <FNCOMPARE>
1 [2] LINEA
2 [3] LINEA
```

SHOWING: 1. DIFFERENT FUNCTION NAMES
2. LINE [1] OF <FOO> NOT FOUND IN <GOO>
3. THE LINE <LINEA> FOUND IN BOTH FUNCTIONS OCCURS BEFORE THE OTHER TWO LINES IN <FOO>, BUT AFTER THEM IN <GOO>.

- -

TITLE: CDB/INTEREST.1

TYPE: FUNCTION

SUMMARY: ALLOWS EASY SPECIFICATION OF VARYING INTEREST RATES

TIMESTAMP: 1983-11-15 16:02:49

CATEGORIES: 38 BUSINESS AND FINANCE

DESCRIPTION:
ALLOWS EASY SPECIFICATION OF VARYING INTEREST RATES.
RIGHT ARGUMENT IS A LIST OF DECIMAL INTEREST RATES ϵ[0,1), OPTIONALLY CONTAINING INTEGER REPLICATION FACTORS FOLLOWING ANY RATE.
FOR EXAMPLE, INTEREST RATES OF .085 FOR 5 YEARS, .08 FOR 3 YEARS AND .075 THEREAFTER CAN BE ENTERED AS:

```
INTEREST .085 5 .08 3 .075  ↔ .085 .085 .085 .085 .085 .08 .08 .08 .075
```

TITLE: CDB/QUOTA.1

TYPE: FUNCTION

SUMMARY: RETURNS DISPLAY OF CURRENT USER'S QUOTAS.

TIMESTAMP: 1984-01-21 22:53:11

CATEGORIES: 36 MEASURING USAGE AND CHARGES

DESCRIPTION:
RETURNS DISPLAY OF CURRENT USER'S QUOTAS, FOR EXAMPLE:

```
      QUOTA
USER CLASS                INTERNAL
WORKSPACE QUOTA           25
WORKSPACES SAVED          17
CPU LIMIT                 NONE
FILE SYSTEM VOLUME CLASS  11
FILE QUOTA                80
NUMBER OF FILES CREATED   25
FILE RESERVATION LIMIT    25000960
FILE BYTES RESERVED       11834368
```

TITLE: CDB/TALK.3

TYPE: FUNCTION

SUMMARY: ALLOWS INTERACTIVE COMMUNICATION WITH AN STASK

TIMESTAMP: 1984-10-21 15:43:02

CATEGORIES: 17 S-TASKS

DESCRIPTION:
<TALK> ALLOWS INTERACTIVE COMMUNICATION WITH AN STASK. THE SYNTAX IS:

```
     R← B TALK SIGNON
```

<B> IS OPTIONAL, AND DETERMINES THE RESULT. IF B=1, THEN THE RESULT IS THE ENTIRE DIALOGUE WITH THE STASK; OTHERWISE, THE RESULT IS EMPTY.

<SIGNON> IS ONE OF:

1. USERS ACCOUNT NUMBER (1↑⎕AI) - SIGNS ON AN STASK ON USERS ACCOUNT

2. ACCOUNT NUMBER AND PASSWORD, EG: ')1234567:SECRET' - SIGNS ON AN STASK ON THE GIVEN ACCOUNT.

3. '' (I.E. EMPTY) - PRINTS A BLOT AND PROMPTS FOR ACCOUNT NUMBER AND PASSWORD.

AFTER SIGNING ON AN STASK VIA <TALK>, SUBSEQUENT KEYBOARD ENTRIES ARE SENT DIRECTLY TO THE STASK. TO RETURN TO THE TTASK, EITHER SEND:)OFF, TO SIGNOFF THE STASK, OR ELSE SEND: O, BACKSPACE, U, BACKSPACE, T. AFTER ENTERING THE LATTER, YOU WILL BE PROMPTED AS TO WHETHER YOU WANT THE O/U/T TO BE TRANSMITTED TO THE STASK (AS AN INPUT INTERRUPT), OR WHETHER YOU WANT TO SUSPEND TRANSMISSIONS TO THE STASK. ON SUSPENSION, YOU CAN RESUME TRANSMISSIONS TO THE STASK BY ENTERING: RESUME.

*** EXECUTE DETAILS UDESCRIBE 'CDB/TALK.3' FOR MORE INFORMATION

TITLE: CDE/FILEACCESS.4

TYPE: ARRAY

SUMMARY: THE DEFINITION OF THIS UTILITY CONTAINS THE "CDE/FILEACCESS" MANUAL.

TIMESTAMP: 1983-10-18 08:52:29

CATEGORIES: 9 FILES
11 CONCURRENT USE OF FILES ⍝ FILE SHARING
32 FILE PERMISSION CONTROL

DESCRIPTION:
REFERENCE CARD

FUNCTION	SHORT NAME	ACTION
R←CLEARACCESS α	R←CA α	CLEAR ACCESS MATRIX (⎕STAC 0 3⍴0)
R←α GIVEACCESS ω	R←α GA ω	OR ω TO EXITING MATRICES
R←α SETACCESS ω	R←α SA ω	REPLACE SELECTED ENTRIES
R←α REMOVEACCESS ω	R←α RA ω	REMOVE SELECTED ENTRIES
R←LISTACCESS α	R←LA α	LIST ACCESS MATRICES
ω ← U ACCESS A [PASSNO P]		CREATE A FILE ACCESS MATRIX
R ← DACCESS X		DECODE PERMISSION NUMBERS

α REFERENCE TO ONE OR MORE FILES. IT MAY BE ONE OF
* A NUMERIC VECTOR OF FILE TIE NUMBERS
* A TWO COLUMN INTEGER MATRIX, THE FIRST COLUMN BEING THE FILE TIE NUMBERS, AND THE SECOND COLUMN BEING THE ASSOCIATED PASSNUMBERS
* CHARACTER VECTOR OF FILENAMES, DELIMITED BY COMMAS
* CHARACTER MATRIX OF FILENAMES WITH ONE NAME PER ROW (E.G ⎕LIB 1↑⎕AI)

ω A 2 OR 3 COLUMN INTEGER MATRIX IN ACCESS MATRIX FORMAT, WITH THE FOLLOWING MEANING:
1. COLUMN ... ACCOUNT NUMBER (0 MEANS 'ALL')
2. COLUMN ... ACCESS IN CODED FORMAT
3. COLUMN ... PASSNUMBER (IF OMMITED = 0)
NOTE: IN THE CASE OF <REMOVEACCESS> IT MAY ALSO BE AN INTEGER VECTOR OF USER ACCOUNT NUMBERS.

U INTEGER VECTOR OF USER ACCOUNT NUMBER, OR 'ALL'
A INTEGER VECTOR OF FILE PERMISSIONS (A ∈ ¯1,2*(⍳16)-⎕IO) OR 'FULL', OR A CHARACTER VECTOR OF FILE FUNCTION NAMES (E.G. 'APPEND,READ RDAC') DELIMITED BY COMMAS OR BLANKS. THE '⎕' MAY BE ELIDED.
P INTEGER SCALAR OR VECTOR OF PASSNUMBERS (MAY BE OMMITED)
X INTEGER SCALAR OR VECTOR OF FILE ACCESS CODES.
E.G.: X = 13 = 1+4+8 = ⎕READ+⎕ERASE+⎕APPEND
R RESULT OF THE FUNCTIONS

HOW IS AN ENTRY IDENTIFIED ? :
IN THE CASE OF <GIVEACCESS>, <SETACCESS> AND <REMOVEACCESS> THE RELEVANT ROW IN THE FILE ACCESS MATRIX (FROM THE SELECTED FILE) HAS TO BE SEARCHED IN ORDER TO DO THE OPERATIONS - A MATCH IS ONLY FOUND IF ACCOUNT NUMBER AND PASSNUMBER ARE EQUAL.

TITLE: CDE/FILEACCESS/ACCESS.2

TYPE: FUNCTION

SUMMARY: CREATES A FILE ACCESS MATRIX

TIMESTAMP: 1983-10-17 08:54:27

CATEGORIES: 9 FILES
11 CONCURRENT USE OF FILES ◘ FILE SHARING
32 FILE PERMISSION CONTROL

DESCRIPTION:
CREATING AN ACCESSMATRIX

THE SYSTEM ASSISTS YOU IN CREATING A FILE ACCESS MATRIX WITH THE FUNCTIONS <ACCESS> AND <PASSNO>. THEY ARE USED AS FOLLOWS:

SYNTAX R ← U ACCESS A [PASSNO P]

U INTEGER VECTOR OF ACCOUNT NUMBERS, OR 'ALL'

A DESIRED ACCESS WHICH WILL BE APPLIED TO ALL ACCOUNTS IN U. MAY BE ONE OF
* INTEGER VECTOR OF PERMISSION CODES (SEE SECTION 'FILE SYSTEM FUNCTIONS AND PERMISSIONS')
* CHARACTER VECTOR OF (ABBRIVIATED) FILE SYSTEM FUNCTION NAMES, DELIMITED BY COMMA. THE '□' MAY BE OMMITED.
* 'FULL'

P INTEGER SCALAR OR VECTOR OF PASSNUMBERS. IF YOU USE A VECTOR, IT'S LENGTH MUST MATCH WITH U. THE PASSNUMBER STATEMENT MAY BE OMMITED.

DESCRIPTION CREATES A FILE ACCESS MATRIX WHICH COULD BE PASSED TO THE FUNCTIONS <SETACCESS>, <GIVEACCESS> AND <REMOVEACCESS>.
RESULT RESULT IS A 2 OR 3 COLUMN ACCESS MATRIX, DEPENDING IF <PASSNO> IS USED OR NOT.
ERRORS SIGNALLED ERROR NUMBERS MAY BE 500 501 502 504 510.
EXAMPLES

```
          )LOAD ULIBRARY
SAVED  10.42.23 10/15/82
      UDEF 'CDE/FILEACCESS/*'
      1234567 ACCESS 1 16
1234567      17
      1234567 ACCESS 8 8 PASSNO 0
1234567       8       0
      1234567 ACCESS 8 8 PASSNO 99
1234567       8      99
      1234567 1002 ACCESS 'FULL' PASSNO 0
1234567      ¯1       0
1002         ¯1       0
       1234567 ACCESS 1 8 PASSNO  0 100
ILLEGAL PASSNUMBER LIST LENGTH
       1234567 ACCESS 1 8 PASSNO 0 100
                      ^
       'ALL' ACCESS 1
0 1
       'ALL' ACCESS 'READ'
```

```
0 1
        'ALL' ACCESS 'READ,HOLD,RDCI,RDAC' PASSNO 123
0 4673  123
```

TITLE: *CDE/FILEACCESS/CLEARACCESS.2*

TYPE: *FUNCTION*

SUMMARY: *CLEARS ACCESS MATRICES OF ALL REFERENCED FILES*

TIMESTAMP: 1983-10-18 07:59:06

CATEGORIES:
9 *FILES*
11 *CONCURRENT USE OF FILES ¤ FILE SHARING*
32 *FILE PERMISSION CONTROL*

DESCRIPTION:
CLEARACCESS

SYNTAX *R←CLEARACCESS ⍺*

RIGHT ARGUMENT *REFERENCE TO ONE OR MORE FILES*

DESCRIPTION *CLEARS THE ACCESS MATRICES OF ALL REFERENCED FILES, I.E. SETS THEM TO* 0 3⍴0. *THIS MEANS THAT THEN ONLY THE OWNER OF THE FILES HAS FULL ACCESS TO THEM WITH PASSNUMBER* 0. *YOU CAN PERFORM THIS ACTION ONLY TO FILES TO WHICH YOU HAVE* ⎕*STAC ACCESS.*

RESULT *A TWO COLUMN INTEGER RETURN CODE MATRIX, WITH ONE ROW PER REFERENCED FILE.*

ERRORS *POSSIBLE RETURN ERROR CODES ARE*

18 19 22 24

SIGNALLED ERROR NUMBER MAY BE 520.

EXAMPLE

```
      CLEARACCESS 'FILE1,FILE2'
1  0
0 24
```

TITLE: *CDE/FILEACCESS/DACCESS.1*

TYPE: *FUNCTION*

SUMMARY: *DECODE PERMISSION NUMBERS TO FILE FUNCTION NAMES*

TIMESTAMP: 1983-10-18 08:20:33

CATEGORIES:
9 *FILES*
11 *CONCURRENT USE OF FILES* ◘ *FILE SHARING*
32 *FILE PERMISSION CONTROL*

DESCRIPTION:
DACCESS

SYNTAX *R←DACCESS X*

RIGHT ARGUMENT *INTEGER SCALAR OR VECTOR OF FILE ACCESS CODES, AS DEFINED BY THE SECOND COLUMN IF THE FILE ACCESS MATRICES.*

DESCRIPTION *THIS FUNCTION TAKES INTEGER FILE ACCESS CODES AS IT'S ARGUMENTS AND DECODES THEM TO THE APPROPRIATE FILE FUNCTIONS NAMES WHICH THEY PERMIT. THE RESULT IS A CHARACTER VECTOR WITH EMBEDED CARRIAGE RETURNS, WITH ONE LINE PER ENTRY IN THE ARGUMENT.*

RESULT *A CHARACTER VECTOR.*

ERRORS *SIGNALLED ERROR NUMBER MAY BE*: 501

EXAMPLE

```
      DACCESS 123 ¯1 32768
□APPEND □DROP □HOLD □READ □REPLACE □SIZE □STIE □TIE
FULL
□SIZE □STIE
```

TYPE: *FUNCTION*

SUMMARY: *SETS THE ACCESS MATRICES OF THE REFERENCED FILE; OR'S IT IF AN ENTRY FOR AN ACCOUNT ALREADY EXISTS*

TIMESTAMP: 1983-10-18 08:00:51

CATEGORIES:
9 *FILES*
11 *CONCURRENT USE OF FILES ⍺ FILE SHARING*
32 *FILE PERMISSION CONTROL*

DESCRIPTION:
GIVEACCESS

SYNTAX *R←⍺ GIVEACCESS ⍵*

RIGHT ARGUMENT *2 OR 3 COLUMN INTEGER FILE ACCESS MATRIX*

LEFT ARGUMENT *REFERENCE TO ONE OR MORE FILES*

DESCRIPTION *ADDS THE ACCESS MATRIX IN ⍵ TO THE ACCESS MATRIX ON FILE. IF NOT ENTRY FOR AN INDIVIDUAL USER (THIS IS THE MATCH OF ACCOUNT NUMBER AND PASSNUMBER) IS FOUND, THEN A NEW ROW IS ADDED FOR THAT USER TO THE ACCESS MATRIX ON FILE. IF AN ENTRY FOR AN USER ALREADY EXISTS, THEN THE ENTRY ON FILE AND THE ONE IN ⍵ ARE OR'ED TOGETHER - THIS MEANS THAT A CHECK IS DONE IF THE ACCESS TO BE GRANTED IS ALREADY GIVEN OR NOT. ONLY IF IT WAS NOT GIVEN SO FAR, THEN IT IS ADDED (E.G. ⎕READ). THIS OPERATION IS PERFORMED FOR EACH ROW IN ⍵ (EACH USER). PLEASE NOTE AGAIN, THAT IF ⍵ IS ONLY A TWO COLUMN MATRIX (NO PASSNUMBERS) IT MEANS PASSNUMBERS OF ZERO (0). THEN ⍵ IS PROCESSED FOR ALL REFERENCED FILES.*
YOU CAN PERFORM THIS ACTION ONLY TO FILES TO WHICH YOU HAVE ⎕STAC ACCESS.

RESULT *A TWO COLUMN INTEGER RETURN CODE MATRIX, WITH ONE ROW PER REFERENCED FILE.*

ERRORS *POSSIBLE RETURN ERROR CODES ARE*

18 19 22 24

SIGNALLED ERROR NUMBERS MAY BE 520, 521.

EXAMPLE

```
      'FILE' GIVEACCESS 1 2⍴(1+⎕AI),¯1
1 0
```

TITLE: CDE/FILEACCESS/LISTACCESS.2

TYPE: FUNCTION

SUMMARY: LISTS THE FILE ACCESS MATRICES IN A GRAPHICAL WAY

TIMESTAMP: 1983-10-17 08:58:27

CATEGORIES:
9 FILES
11 CONCURRENT USE OF FILES ⍝ FILE SHARING
32 FILE PERMISSION CONTROL

DESCRIPTION:
LISTACCESS

SYNTAX R←LISTACCESS ⍺

RIGHT ARGUMENT REFERENCE TO ONE OR MORE FILES

DESCRIPTION LIST THE ACCESS MATRICES FOR THE REFERENCED FILES IN A DECODED AND GRAPHICAL WAY. IT SHOWS THE INTERSECTIONS OF EACH USER AND EACH FILE SYSTEM FUNCTION, AND PUTS A '*' IF THE PARTICULAR ACCESS IS GIVEN TO THAT USER. THE LISTING IS THE EXPLICIT RESULT OF THE FUNCTION. R IS A ONE COLUMN ENCLOSED MATRIX WITH ONE CELL (ROW) PER REFERENCED FILE. EACH CELL MAY BE EITHER A CHARACTER MATRIX (LISTING) OR A CHARACTER VECTOR WITH AN ERROR MESSAGE (E.G. '? FILE NAME ERROR').

ALL YOU NEED IS ⎕RDAC ACCESS TO THE FILES YOU WANT TO REPORT.

RESULT ONE COLUMN ENCLOSED MATRIX WITH REPORT

ERRORS SIGNALLED ERROR MAY BE 520

EXAMPLE PLEASE SEE SECTION 'EXAMPLES'.

- -

TITLE: CDE/FILEACCESS/PASSNO.2

TYPE: FUNCTION

SUMMARY: APPLY PASSNUMBER; CATENATES ENCLOSURE OF ⍺ AND ⍵

TIMESTAMP: 1983-10-17 08:36:29

CATEGORIES:
9 FILES
11 CONCURRENT USE OF FILES ⍝ FILE SHARING
32 FILE PERMISSION CONTROL

DESCRIPTION:
SEE FUNCTION 'CDE/FILEACCESS/ACCESS'

TITLE: *CDE/FILEACCESS/REMOVEACCESS.2*

TYPE: *FUNCTION*

SUMMARY: *REMOVES ALL OR SPECIFIC FILE ACCESS FOR SELECTED ACCOUNTS*

TIMESTAMP: 1983-10-18 08:12:57

CATEGORIES:
9 *FILES*
11 *CONCURRENT USE OF FILES ⍝ FILE SHARING*
32 *FILE PERMISSION CONTROL*

DESCRIPTION:
REMOVEACCESS

SYNTAX *R←⍺ REMOVEACCESS ⍵*

RIGHT ARGUMENT *2 OR 3 COLUMN INTEGER FILE ACCESS MATRIX, OR AN INTEGER ACCOUNT NUMBER VECTOR.*

LEFT ARGUMENT *REFERENCE TO ONE OR MORE FILES*

DESCRIPTION *REMOVES (WITHDRAWS) ENTRIES FROM THE ACCESS MATRIX ON FILE FOR SELECTED USERS. IF ⍵ IS A MATRIX, THEN WITHDRAWING IS DONE AS FOLLOWS: FOR EACH USER (THIS IS THE COMBINATION OF ACCOUNT NUMBER AND PASSNUMBER) THE MATCH(ES) ARE FOUND ON FILE. IF NO MATCH IS FOUND, THERE IS NO PROBLEM. IF A MATCH IS FOUND, THEN ONLY THOSE ACCESS PERMISSIONS SELECTED IN ⍵ ARE REMOVED (E.G. ONLY □APPEND+□REPLACE). ALL OTHER PERMISSIONS ARE LEFT UNCHANGED. IF NO MORE ACCESS FOR A USER IS LEFT, THE ROW IN THE ACCESS MATRIX IS DELETED. THEREFORE, REQUESTING TO REMOVE PERMISSION ¯1 WILL REMOVE ALL PERMISSIONS FOR THAT USERS. THIS OPERATION IS PERFORMED FOR EACH ROW IN ⍵.*
IF ⍵ IS A VECTOR, THEN IT IS TREATED AS AN ACCOUNT NUMBER VECTOR, AND ALL ENTRIES FOR THE SELECTED ACCOUNT NUMBERS ARE REMOVED COMPLETELY - THIS MEANS NO CHECK FOR PASSNUMBERS IS DONE.

RESULT *A TWO COLUMN INTEGER RETURN CODE MATRIX, WITH ONE ROW PER REFERENCED FILE.*

ERRORS *POSSIBLE RETURN ERROR CODES ARE*
18 19 22 24
SIGNALLED ERROR NUMBERS MAY BE 520, 521, 522.

EXAMPLE

```
      'FILE' REMOVEACCESS 2 3⍴0 ¯1 0 1726354 8 99
1 0
```

TYPE: FUNCTION

SUMMARY: SETS THE ACCESS MATRICES OF SELECTED FILES, REPLACES ENTRIES OF EXITSTING REFERENCED ACCOUNTS

TIMESTAMP: 1983-10-18 09:04:43

CATEGORIES:
9 FILES
11 CONCURRENT USE OF FILES ⍝ FILE SHARING
32 FILE PERMISSION CONTROL

DESCRIPTION:
SETACCESS

SYNTAX R←⍺ SETACCESS ⍵

RIGHT ARGUMENT 2 OR 3 COLUMN INTEGER FILE ACCESS MATRIX

LEFT ARGUMENT REFERENCE TO ONE OR MORE FILES

DESCRIPTION ADDS THE ACCESS MATRIX IN ⍵ TO THE ACCESS MATRIX ON FILE. IF NO ENTRY FOR AN INDIVIDUAL USER (THIS IS THE MATCH OF ACCOUNT NUMBER AND PASSNUMBER) IS FOUND, THEN A NEW ROW IS ADDED FOR THAT USER TO THE ACCESS MATRIX ON FILE. IF AN ENTRY FOR AN USER ALREADY EXISTS, THEN THE ENTRY ON FILE IS REPLACED BY THE ENTRY IN ⍵. THIS OPERATION IS PERFORMED FOR EACH ROW IN ⍵ (EACH USER). NOTE: IF ⍵ IS ONLY A TWO COLUMN MATRIX (NO PASSNUMBERS) IT MEANS PASSNUMBERS OF ZERO (0).
THEN ⍵ IS PROCESSED FOR ALL REFERENCED FILES.
YOU CAN PERFORM THIS ACTION ONLY TO FILES TO WHICH YOU HAVE □STAC ACCESS.

RESULT A TWO COLUMN INTEGER RETURN CODE MATRIX, WITH ONE ROW PER REFERENCED FILE.

ERRORS POSSIBLE RETURN ERROR CODES ARE

18 19 22 24

SIGNALLED ERROR NUMBERS MAY BE 520, 521.

EXAMPLE

```
      'FILE1,FILE2' SETACCESS 2 2⍴0 1 99 1
1 0
1 0
```

TITLE: *DBA/RCAT.1*

TYPE: *FUNCTION*

SUMMARY: *VERY FAST ROWWISE CATENATION OF CHARACTER VECTORS OR MATRICES*

TIMESTAMP: 1984-01-18 22:33:44

CATEGORIES: 2 *STRUCTURAL TRANSFORMATIONS* ⍝ *RESHAPING, CATENATING, TRANSPOSING, ETC.*
4 *ENCLOSED ARRAYS*

DESCRIPTION:
PERFORMS ROWWISE CATENATION OF CHARACTER ARRAYS. USES ⍪ *WITH ENCLOSED ARRAYS.*

TITLE: DJK/COVERFNS.1

TYPE: ARRAY

SUMMARY: A DESCRIPTION OF A SET OF UTILITIES FOR MAINTAINING FUNCTIONS ON FILE.

TIMESTAMP: 1983-10-20 22:59:09

CATEGORIES:
8 PACKAGES
9 FILES
13 FILE TOOLS
28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
IN MANY APPLICATIONS, IT IS DESIRABLE TO ALLOW USERS TO BE ABLE TO COPY A SINGLE FN FROM A WS, AND BE ABLE TO USE THIS FN IN THEIR OWN WSS. OFTEN, HOWEVER, THE FN TO BE COPIED WILL REQUIRE ONE OR MORE SUBFNS FOR ITS EXECUTION. IN SUCH CASES, THE FN)COPY'D BY THE USER MAY ⎕PDEF ⎕READ THE SUBFNS IT NEEDS FROM A FILE. THE OBJECTS ⎕PDEF'D WILL USUALLY BE LOCALIZED IN THE FN'S HEADER.

IN SOME APPLICATIONS, THE LIST OF SUBFNS REQUIRED MAY CHANGE WITH TIME. IF SO, THE ABOVE SCHEME IS NOT ADEQUATE. IT IS NECESSARY TO HAVE THE USER-COPIED FN CALL ANOTHER FN WHICH THEN ⎕PDEF'S THE SUBFNS. THE SECOND FN CONTAINS THE NAMES OF THE SUBFNS IN ITS HEADER. THE TWO FNS MIGHT LOOK LIKE THOSE BELOW.

```
     ∇ RESULT←ARG1 COVER ARG2;FN;TIENUM
 [1]   TIENUM←(⎕NAMES∧.='    1234567 FILENAME    ')/⎕NUMS
 [2]   ⍎(0=⍴TIENUM)/''''1234567 FILENAME'' ⎕STIE TIENUM←54321+⌈/0,⎕NUMS'
 [3]   'FN' ⎕PDEF ⎕READ TIENUM,1
 [4]   RESULT←ARG1 FN ARG2
 [5]   ⎕UNTIE(⍴⍴TIENUM)⊃TIENUM ⍝ UNTIE FILE IF IT WAS TIED BY LINE 2
     ∇

     ∇ RESULT←ARG1 FN ARG2;FN;SUBFN1;SUBFN2;SUBFN3
 [1]   ⎕PDEF ⎕READ((⎕NAMES∧.='    1234567 FILENAME    ')/⎕NUMS),23
 [2]   RESULT←ARG1 FN ARG2 ⍝ LINE 1 ⎕PDEF'S <FN> AND ITS SUBFNS
     ∇
```

<COVER> FIRST TIES FILE 1234567 FILENAME. IT THEN ⎕PDEF'S <FN>, WHICH IS ASSUMED TO BE IN A PACKAGE IN COMPONENT 1 OF THE FILE. THE <FN> FROM COMPONENT 1 ⎕PDEF'S A DIFFERENT <FN> FROM COMPONENT 23, AND THEN EXECUTES IT. THE SOLE PURPOSE OF THE COMPONENT-1 <FN> IS TO LOCALIZE THE NAMES OF THE SUBFNS REQUIRED BY THE 'REAL' <FN> IN COMPONENT 23.

THIS DOUBLE-COVER SCHEME WILL HANDLE SEVERAL USER-COPIED FNS. EACH USER-COPIED FN WOULD ⎕PDEF A DIFFERENT FN FROM COMPONENT 1. EACH FN IN COMPONENT 1 COULD READ FROM A DIFFERENT COMPONENT.

DJK/COVERFNS/* IS A SET OF UTILITIES DESIGNED TO REDUCE THE AMOUNT OF WORK REQUIRED TO MAINTAIN THE FILE. IT WAS WRITTEN BY DOUGLAS J. KEENAN. FOR INFORMATION ON THESE UTILITIES, EXECUTE UDESCRIBE 'DJK/COVERFNS/*'.

TITLE: DJK/COVERFNS/CFCHANGE.1 DJK/COVERFNS/CFCHANGE.1

TYPE: FUNCTION

SUMMARY: APPLIES <CH> TO FUNCTIONS IN A PACKAGE ON FILE.

TIMESTAMP: 1983-10-20 22:58:24

CATEGORIES:
8 PACKAGES
9 FILES
13 FILE TOOLS
24 EDITING
28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
COMP CFCHANGE FNNAMES

<CFCHANGE> APPLIES <CH> (SUCH AS FROM WS 4 CH) TO FUNCTIONS IN A PACKAGE ON FILE. THE LEFT ARGUMENT SHOULD BE OF THE FORM TIENUM,COMPNUM. FOR EXAMPLE, IF <FN1> AND <FN2> ARE PACKAGED IN COMPONENT 1 OF FILE 72, THEN

```
      72 1 CFCHANGE 'FN1 FN2'
```

WILL CALL <CH> FOR THE ⎕CR OF <FN1> AND <FN2>. IF <FNNAMES> IS '' THEN <CH> IS CALLED FOR ALL FUNCTIONS IN THE PACKAGE.

IT IS ASSUMED THAT <CH> EXISTS IN THE ACTIVE WS. IF NECESSARY, <CH> CAN BE COPIED FROM WS 4 CH.

<CFCHANGE> IS ⎕IO-INDEPENDENT.

- -

TITLE: DJK/COVERFNS/CFFIND.1 DJK/COVERFNS/CFFIND.1

TYPE: FUNCTION

SUMMARY: RETURNS THE NAMES OF ALL FUNCTIONS IN A PACKAGE ON FILE THAT LOCALIZE ONE OR MORE SPECIFIED NAMES.

TIMESTAMP: 1983-10-20 22:59:57

CATEGORIES:
8 PACKAGES
9 FILES
13 FILE TOOLS
28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
FNNAMES←COMP CFFIND NAMES

EXAMPLE: SUPPOSE COMPONENT 1 OF FILE 72 CONTAINS A PACKAGE; THEN

```
      72 1 CFFIND 'VTOM ARRAY'
```

WILL RETURN THE NAMES OF ALL FUNCTIONS IN THE PACKAGE THAT CONTAIN THE NAME 'VTOM' OR 'ARRAY' IN THEIR HEADER NAMELISTS, I.E. THAT INCLUDE, FOLLOWING ';', THE NAME 'VTOM' OR 'ARRAY' IN THEIR HEADERS.

<CFFIND> IS ⎕IO-INDEPENDENT.

TITLE: DJK/COVERFNS/CFREPLACE.1

TYPE: FUNCTION

SUMMARY: FOR FUNCTIONS IN A PACKAGE ON FILE: REPLACES NAMES IN A FUNCTION HEADER NAMELIST.

TIMESTAMP: 1983-10-20 23:00:50

CATEGORIES:
8 PACKAGES
9 FILES
13 FILE TOOLS
28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
MESSAGE←COMP CFREPLACE NAME BY NAMES

(<BY> IS THE SUBUTILITY DJK/COVERFNS/BY.) EXAMPLE: SUPPOSE COMPONENT 1 OF FILE 72 CONTAINS A PACKAGE; THEN

```
      72 1 CFREPLACE 'VTOM' BY 'VTOM STRINGFIND'
```

WILL CAUSE ';STRINGFIND' TO BE INCLUDED IN THE HEADER OF EACH FUNCTION IN THE PACKAGE THAT INCLUDED <VTOM> IN ITS HEADER NAMELIST (I.E. OF EACH FUNCTION THAT INCLUDED, FOLLOWING ';', THE NAME 'VTOM' IN ITS HEADER). SIMILARLY,

```
      72 1 CFREPLACE 'VTOM' BY ''
```

WILL CAUSE <VTOM> TO BE REMOVED FROM THE HEADER NAMELISTS OF ALL FUNCTIONS IN THE PACKAGE. IN GENERAL, EACH FUNCTION IN THE PACKAGE IN THE COMPONENT POINTED TO BY <COMP> WILL HAVE ITS HEADER SEARCHED FOR <NAME>. EACH SUCH OCCURENCE WILL THEN BE REPLACED BY <NAMES> (WITH SEMICOLONS INSERTED AS APPROPRIATE).

THE NAMES IN THE HEADER NAMELIST OF EACH FUNCTION THAT IS CHANGED WILL BE SORTED AND DUPLICATES WILL BE REMOVED.

THE RESULT, <MESSAGE>, IS A CHARACTER VECTOR INDICATING HOW MANY FUNCTIONS WERE ALTERED. <CFREPLACE> IS ⎕IO-INDEPENDENT.

TITLE: DJK/COVERFNS/CFRETRIEVE.1

TYPE: FUNCTION

SUMMARY: RETRIEVES THE COMPONENT THAT IS READ BY A FN (SEE THE DESCRIPTION OF DJK/COVERFNS FOR DETAILS).

TIMESTAMP: 1983-10-20 23:01:48

CATEGORIES:
8 PACKAGES
9 FILES
13 FILE TOOLS
28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
PACKAGE←COMP CFRETRIEVE FNNAME

FOR THE EXAMPLE GIVEN IN THE DESCRIPTION OF DJK/COVERFNS,

```
      72 1 CFRETRIEVE 'FN'
```

WOULD RETURN THE PACKAGE CONTAINED IN COMPONENT 23 (ASSUMING THE FILE TIE NUMBER IS 72). <FNNAME> SHOULD BE THE NAME OF A FUNCTION IN THE PACKAGE IN COMPONENT 1↑1↓COMP OF FILE 1↑COMP.

<CFRETRIEVE> LOOKS AT LINE 1 OF THE FUNCTION; THE LINE IS ASSUMED TO END WITH ',COMPONENTNUMBER'. PACKAGE ↔ ⎕READ (1↑COMP),COMPONENTNUMBER,2↓COMP.

<CFRETRIEVE> IS ⎕IO-INDEPENDENT.

TITLE: DJK/COVERFNS/CFSTORE.1

TYPE: FUNCTION

SUMMARY: PACKAGES AND STORES ON FILE APPROPRIATE OBJECTS IN THE ACTIVE WS.

TIMESTAMP: 1983-10-20 23:02:34

CATEGORIES:
8 PACKAGES
9 FILES
13 FILE TOOLS
28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
MESSAGE←COMP CFSTORE NAMES

THIS FUNCTION IS INTENDED TO BE USED WITH A DOUBLE-COVER SCHEME SUCH AS THE ONE OUTLINED IN THE DESCRIPTION FIELD OF DJK/COVERFNS. THE EXAMPLES ARE BASED ON <FN>, GIVEN THERE.

WHEN A CHANGE IS MADE TO THE 'REAL' <FN> OR ANY OF ITS SUBFUNCTIONS, <FN> AND ALL OF ITS SUBFUNCTIONS MUST BE PACKAGED AND PLACED IN COMPONENT 23 OF THE FILE. <CFSTORE> CAN AID IN DOING THIS. FOR EXAMPLE, SUPPOSE <FN> AND ALL ITS SUBFUNCTIONS EXIST IN THE ACTIVE WS. THEN

```
72 1 CFSTORE 'FN'
```

WILL PACKAGE <FN> AND ITS SUBFUNCTIONS AND THEN PUT THE PACKAGE IN COMPONENT 23. THE NAMES OF THE OBJECTS TO PACKAGE ARE DETERMINED BY LOOKING AT THE HEADER NAMELIST OF THE COMPONENT-1 <FN> (ONLY THOSE NAMES FOLLOWING A ';' ARE LOOKED AT). THE COMPONENT NUMBER IN WHICH TO PUT THE PACKAGE IS DETERMINED BY LOOKING AT LINE 1 OF OF THE COMPONENT-1 <FN>, WHICH SHOULD END WITH ',COMPONENTNUMBER'.

THE RIGHT ARGUMENT CAN CONTAIN MORE THAN ONE NAME. FOR EXAMPLE, IF THE PACKAGE IN COMPONENT 1 CONTAINS FUNCTIONS <FNONE> AND <FNTWO> THEN

```
72 1 CFSTORE 'FNONE FNTWO'
```

WOULD BE VALID. THE NAMES SPECIFIED IN THE RIGHT ARGUMENT DO NOT, HOWEVER, ALWAYS HAVE TO REFER TO FUNCTIONS IN COMPONENT 1. IN GENERAL,

```
COMP CFSTORE NAMES  ↔  COMP CFSTORE COMP CFFIND NAMES
```

WHERE <CFFIND> IS DJK/COVERFNS/CFFIND. THUS, IF A CHANGE IS MADE TO ANY OBJECT <FOO> THEN 72 1 CFSTORE 'FOO' IS EQUIVALENT TO CALLING <CFSTORE> FOR EACH FUNCTION IN COMPONENT 1 THAT INCLUDES <FOO> IN ITS HEADER NAMELIST. SIMILARLY, 72 1 CFSTORE 'FOO GOO' IS EQUIVALENT TO CALLING <CFSTORE> FOR EACH FUNCTION IN COMPONENT 1 THAT INCLUDES <FOO> OR <GOO> IN ITS HEADER NAMELIST.

```
72 1 CFSTORE ''
```

IS EQUIVALENT TO CALLING <CFSTORE> FOR EACH FUNCTION IN COMPONENT 1 OF FILE 72.

THE RESULT, <MESSAGE>, IS A CHARACTER VECTOR INDICATING HOW MANY COMPONENTS WERE REPLACED. <CFSTORE> IS ⎕IO-INDEPENDENT.

TITLE: DJK/CPU.1 DJK/CPU.1

TYPE: FUNCTION

SUMMARY: MEASURES THE NUMBER OF CPU UNITS REQUIRED TO EXECUTE EXPRESSIONS.

TIMESTAMP: 1983-08-28 22:45:57

CATEGORIES: 19 EXECUTION MONITORING
28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS
34 MEASURING TIME AND SPACE REQUIREMENTS
36 MEASURING USAGE AND CHARGES

DESCRIPTION:
MILLIUNITS←REPETITIONS CPU EXPRESSIONS

<CPU> MEASURES HOW MANY MILLIUNITS ARE REQUIRED TO EXECUTE ONE OR MORE EXPRESSIONS, AS IN THE FOLLOWING EXAMPLE.

```
      10 CPU '⍳1000'⊃'+/⍳1000'⊃'+/0.1+⍳1000'
22.4 35 84.7
```

THE RIGHT ARGUMENT SHOULD BE A VECTOR OF ENCLOSED CHARACTER VECTORS. THE LEFT ARGUMENT IS THE NUMBER OF TIMES EACH EXPRESSION IS TO BE EXECUTED. THE RESULT IS A NUMERIC VECTOR INDICATING HOW MANY MILLIUNITS ARE REQUIRED TO EXECUTE EACH EXPRESSION ONCE. THE RESULT IS INHERENTLY APPROXIMATE; INCREASING THE NUMBER OF REPETITIONS WILL USUALLY INCREASE THE ACCURACY OF THE RESULT. NOTE THAT THE RESULT CAN VARY WIDELY DEPENDING ON THE WORKSPACE IN WHICH <CPU> IS EXECUTED.

OVERHEAD INCURRED BY <CPU> IS NOT ISOLATED OUT OF THE RESULT. THUS THE ACTUAL NUMBER OF MILLIUNITS USED CAN BE OBTAINED VIA (N CPU EXPRESSIONS)-N CPU '0'.

EACH EXPRESSION MAY BE ANY APL EXPRESSION, BUT NOT A STATEMENT, I.E. IT MUST RETURN A RESULT AND MAY NOT CONTAIN DIAMONDS.

<CPU> IS ⎕IO-INDEPENDENT.

- -

TITLE: DJK/DATATYPE.1 DJK/DATATYPE.1

TYPE: FUNCTION

SUMMARY: DETERMINES THE INTERNAL DATA TYPE OF AN ARRAY (OR PACKAGE).

TIMESTAMP: 1984-01-26 00:27:57

CATEGORIES: 1 MISCELLANEOUS
34 MEASURING TIME AND SPACE REQUIREMENTS

DESCRIPTION:
TYPE←DATATYPE X

DETERMINES THE INTERNAL DATATYPE OF <X>. <X> MAY BE ANYTHING. THE RESULT IS ONE OF THE FOLLOWING: 'BOOL', 'CHAR', 'INT ', 'FLPT', 'CPLX', 'ENCL', AND 'PACK'. E.G. DATATYPE 0⍴1.2 ←→ 'FLPT'.

<DATATYPE> IS ⎕IO-INDEPENDENT.

TITLE: DJK/DECOMMENT.1 DJK/DECOMMENT.1

TYPE: FUNCTION

SUMMARY: REMOVES COMMENTS FROM SPECIFIED FUNCTIONS.

TIMESTAMP: 1984-12-08 03:45:02

CATEGORIES: 28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
BOOLEAN←DECOMMENT FNNAMES

REMOVES COMMENTS FROM EACH FUNCTION NAMED IN THE MATRIX NAMELIST <FNNAMES>. FULL-LINE COMMENTS ARE REPLACED WITH A SINGLE '⍝'. END-OF-LINE COMMENTS ARE COMPLETELY REMOVED.

BOOLEAN[I] INDICATES IF FNNAMES[I;] WAS SUCCESSFULLY DECOMMENTED. DECOMMENTING WILL NOT BE SUCCESSFUL IF FNNAMES[I;] REFERS TO A LOCKED FUNCTION, A FUNCTION ON THE EXECUTION STACK, OR TO SOMETHING OTHER THAN A FUNCTION.

<DECOMMENT> IS ⎕IO-INDEPENDENT.

- -

TITLE: DJK/EACH.2 DJK/EACH.2

TYPE: FUNCTION

SUMMARY: E.G. 'FOO ω×2' EACH 5⊃6⊃<⍳7 ↔ (FOO 5×2)⊃(FOO 6×2)⊃<FOO (⍳7)×2.

TIMESTAMP: 1984-08-15 00:29:45

CATEGORIES: 4 ENCLOSED ARRAYS

DESCRIPTION:
RESULT←α¨> ω WHERE ω IS A ARRAY AND α IS A MONADIC DIRECT-DEFINITION EXPRESSION (NOT A STATEMENT--I.E. IT MAY NOT CONTAIN DIAMONDS).

EXAMPLES

```
      ⎕IO←1
      ⎕PS←¯1 ¯1 0 ¯3

      ρ⎕←'⍳ω' EACH ⍳4
|1| |1 2| |1 2 3| |1 2 3 4|
4

      ρ'YOURFN ω' EACH ARRAY  ↔  ρARRAY
      'YOURFN ω' EACH ,ARRAY
  ↔   (YOURFN >ARRAY[1])⊃(YOURFN >ARRAY[2])⊃ ... ⊃<YOURFN >ARRAY[N]
```

<EACH> IS ⎕IO-INDEPENDENT.

TITLE: DJK/ENDSPOSE.1

TYPE: FUNCTION

SUMMARY: MOVES SPECIFIED AXES TO THE END OF THE SHAPE VECTOR.

TIMESTAMP: 1983-03-03 21:44:15

CATEGORIES: 2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.

DESCRIPTION:
TRANSPOSED←AXES ENDSPOSE ARRAY

MOVES THE SPECIFIED AXES TO THE END OF THE SHAPE VECTOR.

EXAMPLES
```
      ⎕IO←1
      ⍴1 ENDSPOSE 1 2 3 4 ⍴'A'
2 3 4 1
      ⍴3 2 ENDSPOSE 9 8 7 6 5 ⍴⍳1001
9 6 5 7 8
      ⎕IO←0
      ⍴0 ENDSPOSE ⍳8
8
```

- -

TITLE: DJK/FDELETE.1

TYPE: FUNCTION

SUMMARY: DELETES SPECIFIED COMPONENTS FROM A FILE.

TIMESTAMP: 1983-01-18 05:10:11

CATEGORIES: 9 FILES
12 FILE ORGANIZATION
13 FILE TOOLS
43 FILE PRIMITIVE SIMULATION

DESCRIPTION:
TIENUM FDELETE COMPNUMS

DELETES COMPONENT NUMBERS <COMPNUMS> FROM FILE NUMBER <TIENUM>; FINISHES WITH ⎕DROP TIENUM,-⍴,COMPNUMS. FOR EXAMPLE, IF FILE 7 HAD 21 COMPONENTS AND YOU WANTED TO DELETE COMPONETS 3, 9, AND 17, YOU WOULD EXECUTE THE FOLLOWING.

```
      7 FDELETE 3 9 17
```

AFTER EXECUTING THIS, THE FILE WOULD HAVE 18 COMPONENTS.

THE ELEMENTS OF <COMPNUMS> SHOULD BE DISTINCT.

<FDELETE> IS ⎕IO-INDEPENDENT.

TITLE: DJK/FIRSTLINES.2

TYPE: FUNCTION

SUMMARY: DISPLAYS THE HEADER AND FIRST LINE OF TEXT OF FILED MAILBOX MESSAGES.

TIMESTAMP: 1983-02-18 06:57:36

CATEGORIES:
9 FILES
10 MAILBOX
14 COMMUNICATION BETWEEN TASKS

CHANGES:
EXTENDED TO TRAVERSE THE FILE IN EITHER DIRECTION, AND TO BE ⎕IO-INDEPENDENT.

DESCRIPTION:
MAILFILE FIRSTLINES CNUM

PRINTS THE HEADER AND FIRST LINES OF ALL MESSAGES (IN THE MAIL FILE) WHOSE COMPONENT NUMBER IS AT LEAST <CNUM>, IF CNUM>0, OR AT MOST |CNUM, IF CNUM<0. FOR EXAMPLE, IF EACH COMPONENT OF FILE 66 CONTAINED A MAILBOX MESSAGE, THEN

```
      66 FIRSTLINES 8
```

WOULD DISPLAY THE HEADER AND FIRST LINE OF TEXT OF THE MESSAGES IN COMPONENTS 8, 9, 10, ... , (¯1+1↑1↓⎕SIZE 66), AND

```
      66 FIRSTLINES ¯6
```

WOULD DISPLAY THE HEADER AND FIRST LINE OF TEXT FOR COMPONENTS 6, 5, 4, ... , (1↑⎕SIZE 66).

<FIRSTLINES> IS ⎕IO-INDEPENDENT.

TITLE: DJK/FIRSTLINES.3

TYPE: FUNCTION

SUMMARY: DISPLAYS THE HEADER AND FIRST LINE OF TEXT OF FILED MAILBOX MESSAGES.

TIMESTAMP: 1983-08-07 00:31:12

CATEGORIES:
9 *FILES*
10 *MAILBOX*
14 *COMMUNICATION BETWEEN TASKS*

CHANGES:
ALLOWS ARBITRARILY-NUMBERED COMPONENTS TO BE DISPLAYED.

DESCRIPTION:
MAILFILE FIRSTLINES CNUMS

PRINTS THE HEADER AND FIRST LINES OF ALL MESSAGES (IN THE MAIL FILE) WHOSE COMPONENT NUMBER IS A MEMBER OF <CNUMS>. FOR EXAMPLE, IF THE COMPONENTS OF FILE 66 *CONTAIN MAILBOX MESSAGES, THEN*

66 *FIRSTLINES* 17 8

WOULD DISPLAY THE HEADER AND FIRST LINE OF TEXT OF THE MESSAGES IN COMPONENTS 17 *AND* 8.

<FIRSTLINES> IS □*IO-INDEPENDENT.*

TITLE: DJK/FNSUMMARY.1

TYPE: FUNCTION

SUMMARY: DISPLAYS SYNTAX AND LEADING FULL-LINE COMMENTS OF SPECIFIED FUNCTIONS.

TIMESTAMP: 1984-01-05 20:24:52

CATEGORIES: 28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS
35 WORKSPACE TOOLS ⍝ E.G. WORKSPACE CROSS REFERENCE

DESCRIPTION:
SUMMARIES←FNSUMMARY FNNAMES

THE RESULT IS A CHARACTER MATRIX HOLDING SUMMARIES OF THE SPECIFIED FUNCTIONS. A SUMMARY CONSISTS OF THE FUNCTION SYNTAX AND ANY LEADING FULL-LINE COMMENTS. SHOWN BELOW ARE TWO FUNCTIONS AND THEIR SUMMARIES.

```
 Z←L FUNC1 R;A;B                          Z←LEFT FUNC2 RIGHT;A;B
 ⍝ THIS IS A SAMPLE FUNCTION TO           ⍝ THIS IS ANOTHER SAMPLE FUNCTION.
 ⍝ DEMONSTRATE HOW <FNSUMMARY> WORKS.     Z←LEFT+RIGHT
 A←L+R ⍝ THIS IS COMMENT
 ⍝ THIS IS ALSO A COMMENT.

      ⍴⎕←FNSUMMARY 2 5⍴'FUNC1FUNC2'
   Z←L FUNC1 R  ⍝ THIS IS A SAMPLE FUNCTION TO
                ⍝ DEMONSTRATE HOW <FNSUMMARY> WORKS.
Z←LEFT FUNC2 RIGHT  ⍝ THIS IS ANOTHER SAMPLE FUNCTION.
3 54
```

AS SHOWN, THE FUNCTION NAMES ARE ALLIGNED. COMMENTS ARE NOT ALLIGNED TO ALLOW A MORE COMPACT LISTING. THE FUNCTION IS HANDY WHEN INSPECTING A WORKSPACE QUICKLY, FOR THE FIRST TIME.

THE RIGHT ARGUMENT SHOULD BE A MATRIX NAMELIST OF FUNCTION NAMES. <FNSUMMARY> IS ⎕IO-INDEPENDENT.

TITLE: DJK/FORMATTS.2

TYPE: FUNCTION

SUMMARY: E.G. FORMATTS 1982 9 26 5 52 4 37 ↔ '1982-09-26 05:52:04.037'.

TIMESTAMP: 1983-08-07 22:50:27

CATEGORIES: 25 FORMATTING
37 TIMES AND DATES

CHANGES:
ADHERES TO ISO STANDARDS.

DESCRIPTION:
S←FORMATTS TS

E.G. FORMATTS 1982 9 26 5 52 4 37 ↔ '1982-09-26 05:52:04.037'.

<TS> SHOULD BE A VECTOR OF SEVEN INTEGERS. THE RESULTANT TIMESTAMP ADHERES TO ISO STANDARDS 3307 (TIMES) AND 2014 (DATES).

<FORMATTS> IS □IO-INDEPENDENT.

TITLE: DJK/INROWS.1

TYPE: FUNCTION

SUMMARY: FINDS WHICH ROWS OF A MATRIX CONTAIN A STRING ANYWHERE WITHIN THEM.

TIMESTAMP: 1985-01-01 22:58:28

CATEGORIES:
7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF
24 EDITING
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
INDICES←MATRIX INROWS STRING

<INROWS> RETURNS THE INDICES OF ALL ROWS OF <MATRIX> THAT CONTAIN THE <STRING> ANYWHERE WITHIN THEM. SOME EXAMPLES ARE SHOWN BELOW.

```
      MAT←4 4⍴'ABCDEFG ABCDXYXY'
      MAT
ABCD
EFG
ABCD
XYXY
      MAT INROWS 'BC'
1 3
      MAT INROWS ' '
2
      MAT INROWS 'XY'
4 4
      (3 3⍴⍳9) INROWS 4 5 6  ⍝ ARGUMENTS MAY BE CHARACTER OR NUMERIC
2
      MAT INROWS ''  ⍝ EMPTY STRING MATCHES EVERYWHERE, AS PER SAUCE/STRINGFIND
1 2 3 4
```

SAUCE/STRINGFIND IS USED AS A SUBUTILITY; SO THE ARGUMENTS SHOULD CONTAIN ONLY NUMBERS OR CHARACTERS, NOT ENCLOSURES.

```
      ⎕IO←0
      MAT INROWS 'BC'
0 2
```

AS SHOWN, <INROWS> IS ⎕IO-RESPONSIVE.

TITLE: DJK/PAUSE.1 DJK/PAUSE.1

TYPE: FUNCTION

SUMMARY: <PAUSE> SUSPENDS. THE USER MAY RESUME EXECUTION BY ENTERING 'RESUME'.

TIMESTAMP: 1984-01-25 22:53:47

CATEGORIES:
1 MISCELLANEOUS
20 TERMINAL INPUT/OUTPUT
21 TERMINAL INPUT

DESCRIPTION:
EXAMPLE:

```
      PAUSE
TO RESUME, ENTER  'RESUME'.
      2+2  ⍝ USER DOES SOME STUFF
4
      RESUME
```

<PAUSE> IS INTENDED TO BE CALLED BY AN APPLICATION PROGRAM ON USER REQUEST. THE APPLICATION THEN 'PAUSES'. EXECUTION IS CONTINUED WHEN THE USER EXECUTES <RESUME>. <RESUME> IS A NILADIC FUNCTION LOCALLY DEFINED BY <PAUSE>.

IT IS SAFE, IN THAT EXECUTION WILL NEVER RESUME AT AN INCORRECT POINT. THIS HOLDS EVEN IF THE USER HAS INTRODUCED OTHER SUSPENDED FUNCTIONS WHILE PAUSED.

DJK/PAUSE IS ⎕IO-INDEPENDENT. IT WAS JOINTLY AUTHORED BY DOUGLAS J. KEENAN AND J. HENRI SCHUELER.

- -

TITLE: DJK/PEXECUTE.1 DJK/PEXECUTE.1

TYPE: FUNCTION

SUMMARY: EXECUTES AN EXPRESSION 'WITHIN' A PACKAGE.

TIMESTAMP: 1984-03-16 19:33:48

CATEGORIES: 8 PACKAGES

DESCRIPTION:
RESULT←PACKAGE PEXECUTE EXPRESSION

EXECUTES <EXPRESSION> WITHIN <PACKAGE>. THAT IS, A (LOCAL) FUNCTION IS DEFINED THAT LOCALIZES THE NAMES IN <PACKAGE>, DEFINES THE PACKAGE, AND THEN EXECUTES <EXPRESSION>. <EXPRESSION> SHOULD RETURN A <RESULT>.

<PEXECUTE> IS ⎕IO-INDEPENDENT.

```
TITLE:       DJK/RANDOMIZE.1                                          DJK/RANDOMIZE.1

TYPE:        FUNCTION

SUMMARY:     FINDS A 'RANDOM' VALUE FOR ⎕RL, USING 2↓⎕TS.

TIMESTAMP:   1983-08-18 15:56:43

CATEGORIES:  39  STATISTICS AND PROBABILITY
             40  SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
QRL←RANDOMIZE

FINDS A 'RANDOM' VALUE FOR ⎕RL.  THE VALUE, <QRL>, IS DEPENDENT ON 2↓⎕TS.  THE
RANGE OF VALUES IS ESSENTIALLY THE DOMAIN OF ⎕RL.   EACH ELEMENT OF THE RANGE
HAS ABOUT THE SAME PROBABILITY OF BEING CHOSEN.

<RANDOMIZE> IS ⎕IO-INDEPENDENT.
```

- -

```
TITLE:       DJK/SPACE.1                                              DJK/SPACE.1

TYPE:        FUNCTION

SUMMARY:     CALCULATES AN UPPER BOUND (WITHIN 1K) ON THE BYTES REQUIRED TO EXECUTE AN EXPRESSION.

TIMESTAMP:   1983-08-09 15:32:02

CATEGORIES:  19  EXECUTION MONITORING
             28  DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS
             34  MEASURING TIME AND SPACE REQUIREMENTS

DESCRIPTION:
BYTES←SPACE EXPRESSION

CALCULATES AN UPPER BOUND ON THE NUMBER OF BYTES OF WORKSPACE REQUIRED TO
EXECUTE <EXPRESSION>.  THE UPPER BOUND IS WITHIN 1 KBYTE OF THE ACTUAL NUMBER
OF BYTES REQUIRED.

EXAMPLES:
      SPACE '⍳1000'
4217
      SPACE '0.1+⍳1000'
12392

<EXPRESSION> MAY BE ANY APL EXPRESSION, BUT NOT A STATEMENT, I.E. IT MUST
RETURN A RESULT AND MAY NOT CONTAIN DIAMONDS.

<SPACE> IS ⎕IO-INDEPENDENT.
```

TITLE: DLF/DFN.1

TYPE: FUNCTION

SUMMARY: TAKE ⎕CR OF FUNCTION; RETURN ⎕CR WITH ASSIGNED VARIABLES LOCALIZED

TIMESTAMP: 1984-04-16 20:08:10

CATEGORIES: 24 EDITING
27 TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS
28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
INPUT IS A CHARACTER MATRIX, ASSUMED TO BE CANONICAL MATRIX REPRESENTATION OF A FUNCTION. THE RESULT IS A CHARACTER MATRIX OF THE SAME NUMBER OF ROWS, BUT PERHAPS MORE COLUMNS. ONLY THE FIRST ROW IS ALTERED (THE OTHERS MAY HAVE BLANKS APPENDED). THE FIRST ROW NOW LOCALIZES THOSE VARIABLES ASSIGNED A VALUE WITHIN THE BODY OF THE FUNCTION.
A TYPICAL USAGE MY BE: ⎕FX DFN ⎕CR 'SOMEFUNCTION'
*** QUIRKS, SHORTCOMINGS:
IF VARIABLES ARE ALREADY LOCALIZED THAT ARE NOT ASSIGNED VALUES, THESE LOCALIZATIONS WILL BE LOST. YOU CAN AVOID THIS BY DEFINING A LINE THAT IS NOT EXECUTED, THAT APPEARS TO ASSIGN THEM VALUES.
THE FUNCTION WILL NOT CATCH ASSIGNMENTS THAT OCCUR WITHIN ⍎ OR SUBFUNCTIONS CALLED, OR THROUGH OTHER MECHANISMS THAN ← (FOR EXAMPLE ⎕PDEF, ⎕FX, OR SHARED VARIABLES).
AUTHOR - DOUG FORKES

- -

TITLE: DONW/NAME∆SORT.2

TYPE: FUNCTION

SUMMARY: SORTS MATRIX OF COMPANY NAMES, EXCLUDING LEADING DEFINITE ARTICLES

TIMESTAMP: 1984-01-20 20:52:19

CATEGORIES: 6 SORTING AND GRADING

DESCRIPTION:
R←NAME∆SORT MAT

WILL PUT A 2-DIMENSIONAL ARRAY OF COMPANY (OR OTHER) NAMES INTO ALPHABETICAL ORDER. LEADING 'SPECIAL CASE' WORDS SUCH AS: LE, LA, LES, L' , AND THE ARE IGNORED, AS ARE PUNCTUATION AND SPECIAL CHARACTERS. VERY USEFUL FOR SORTING A LIST CONTAINING A LOT OF FRENCH COMPANY NAMES. THE CONCEPT FOLLOWS THAT OF BELL CANADA IN SORTING NAMES FOR THE TELEPHONE DIRECTORY

TITLE: GLO/PARA.1

TYPE: FUNCTION

SUMMARY: FORMATS A VECTOR OF TEXT TO WITHIN A SPECIFIED WIDTH.

TIMESTAMP: 1984-03-15 03:33:04

CATEGORIES: 25 FORMATTING
27 TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS

DESCRIPTION:
Z←W PARA TEXT

INPUT SPECIFICATIONS
(1) <W> IS A POSITIVE INTEGER SCALAR.
(2) <TEXT> IS A CHARACTER VECTOR WHICH DOES NOT BEGIN WITH A BLANK, AND CONTAINS NO NEWLINES, BACKSPACES, OR OTHER NON-ENTERABLE CHARACTERS.

OUTPUT SPECIFICATIONS
(1) <Z> IS A CHARACTER VECTOR WITH EMBEDDED NEWLINES, WHICH WILL APPEAR AS A PARAGRAPH WHEN DISPLAYED.
(2) EACH LINE OF <Z> WILL CONTAIN AS MANY COMPLETE WORDS AS WILL FIT ON THAT LINE SUBJECT TO THE CONDITION THAT IF THE LAST NON-BLANK CHARACTER (EXCLUDING THE LINE-ENDING NEWLINE) IS IN POSITION P, THEN P≤W.
(3) IF <TEXT> CONTAINS A WORD OF LENGTH GREATER THAN <W>, THEN THAT WORD WILL BE BROKEN SO AS TO APPEAR ON CONSECUTIVE LINES OF <Z>.
(4) NO LINE IN THE PARAGRAPH <Z> WILL COMMENCE WITH A BLANK.
(5) NO CHARACTERS WILL BE DELETED FROM <TEXT>. THUS, IF THE EMBEDDED NEWLINES WERE REMOVED FROM <Z>, THE RESULT WOULD BE IDENTICAL WITH <TEXT>.

EXAMPLE
SUPPOSE <TEXT> IS SPECIFIED AS FOLLOWS.

```
TEXT←'THE QUICK BROWN FOX JUMPED OVER THE LAZY DOG WHILE '
TEXT←TEXT,'THE HUE AND CRY WENT OUT FOR A '
TEXT←TEXT,'PARAGRAPH FORMATTING PROGRAM.'
```

<PARA> WILL RETURN THE RESULTS SHOWN BELOW. THE JOT (∘) HAS BEEN USED TO MARK THE POSITION OF THE NEWLINES.

```
      12 PARA TEXT
THE QUICK ∘
BROWN FOX ∘
JUMPED OVER ∘
THE LAZY DOG ∘
WHILE THE ∘
HUE AND CRY ∘
WENT OUT FOR ∘
A PARAGRAPH ∘
FORMATTING ∘
PROGRAM.

      24 PARA TEXT
THE QUICK BROWN FOX ∘
JUMPED OVER THE LAZY DOG ∘
WHILE THE HUE AND CRY ∘
WENT OUT FOR A PARAGRAPH ∘
FORMATTING PROGRAM.
```

⎕IO IS LOCALIZED AND SET TO 1.

<PARA> WAS WRITTEN BY GEORGE LOUNT. IT WON SHARP APL CONTEST THREE.

TITLE: HUI/APLTOCOURIER.1

TYPE: FUNCTION

SUMMARY: CONVERTS TEXT IN APL FONT TO COURIER FONT.

TIMESTAMP: 1984-12-18 03:59:42

CATEGORIES:
20 TERMINAL INPUT/OUTPUT
24 EDITING
27 TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS
46 MODIFYING ARRAYS ⍝ INDEXED ASSIGNMENT, SUBSTRING REPLACEMENT, ETC.

DESCRIPTION:
RINP←APLTOCOURIER RINP

THE ARGUMENT SHOULD BE A CHARACTER VECTOR. IT IS TREATED AS A VECTOR OF TEXT IN APL FONT. THE RESULT IS THE TEXT IN COURIER FONT. IN PARTICULAR, EACH CHARACTER IN THE FIRST ROW BELOW WILL BE REPLACED BY THE CHARACTER BENEATH IT; ALL OTHER CHARACTERS WILL REMAIN UNCHANGED.

```
$[]();\+-×÷*|<=>!?~_ABCDEFGHIJKLMNOPQRSTUVWXYZ¯':
≤←→∨∧[⊢÷+X/≠⊣;×:¨\$-α⊥∩⌊∊_∇∆⍳∘'⎕|⊤○⋆?ρ⌈~↓∪ω⊃↑⊂+](
```

THE TABLE IS TAKEN FROM DIABLO PRINT WHEELS 38107 (COURIER) AND 38150 (APL).

<APLTOCOURIER> IS ⎕IO-INDEPENDENT.

- -

TITLE: HUI/CLASSIFY.1

TYPE: FUNCTION

SUMMARY: INPUT: RANGES (INTERVALS) AND SOME NUMBERS. OUTPUT: WHICH INTERVAL EACH NUMBER IS IN.

TIMESTAMP: 1984-12-15 20:58:52

CATEGORIES:
6 SORTING AND GRADING
7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF
39 STATISTICS AND PROBABILITY
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
R←ENDPT CLASSIFY DATA

CLASSIFICATION OF <DATA> INTO INTERVALS (RANGES). THE INTERVAL ENDPOINTS ARE (-INFINITY),ENDPTS,(+INFINITY). AN INTERVAL IS THE SET LOWER≤X<UPPER.
ρR ←→ ρDATA. EACH ELEMENT OF <R> IS AN INDEX INTO THE INTERVALS.

EXAMPLE (⎕IO←1): ¯5.5 0 100 CLASSIFY ¯6 2.1 2001 ←→ 1 3 4.

IT IS ASSUMED THAT <ENDPTS> IS IN ASCENDING ORDER.

THE RESULT OF <CLASSIFY> VARIES APPROPRIATELY ACCORDING TO ⎕IO.

TITLE: HUI/COURIERTOAPL.1

TYPE: FUNCTION

SUMMARY: CONVERTS TEXT IN COURIER FONT TO APL FONT.

TIMESTAMP: 1984-12-18 04:01:06

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
24 EDITING
27 TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS
46 MODIFYING ARRAYS ⍝ INDEXED ASSIGNMENT, SUBSTRING REPLACEMENT, ETC.

DESCRIPTION:
RINP←COURIERTOAPL RINP

THE ARGUMENT SHOULD BE A CHARACTER VECTOR. IT IS TREATED AS A VECTOR OF TEXT IN COURIER FONT. THE RESULT IS THE TEXT IN APL FONT. IN PARTICULAR, EACH CHARACTER IN THE FIRST ROW BELOW WILL BE REPLACED BY THE CHARACTER BENEATH IT; ALL OTHER CHARACTERS WILL REMAIN UNCHANGED.

```
$[]();\+-×÷⋆|<=>!?~_ABCDEFGHIJKLMNOPQRSTUVWXYZ¯':
≤←→∨∧[⊢÷+X/≠⊣;×:¨\$-⍺⊥∩⌊∊_∇∆⍳∘'⎕|⊤○⋆?⍴⌈~↓∪⍵⊃↑⊂+](
```

THE TABLE IS TAKEN FROM DIABLO PRINT WHEELS 38107 (COURIER) AND 38150 (APL).

<COURIERTOAPL> IS ⎕IO-INDEPENDENT.

- -

TITLE: HUI/DEPRECIATE.1

TYPE: FUNCTION

SUMMARY: STRAIGHT-LINE DEPRECIATION.

TIMESTAMP: 1984-12-15 20:13:27

CATEGORIES: 5 NUMERIC CALCULATION
38 BUSINESS AND FINANCE

DESCRIPTION:
D←TERM DEPRECIATE AMT

<TERM> IS NUMBER OF PERIODS FOR STRAIGHT-LINE DEPRECIATION.
<AMT> ARE AMOUNTS TO BE DEPRECIATED IN EACH PERIOD.
<D> ARE DEPRECIATION IN EACH PERIOD.

<DEPRECIATE> IS ⎕IO-INDEPENDENT.

TITLE: HUI/DIOPHANTINE.1

TYPE: FUNCTION

SUMMARY: SOLVES C = A+.×X IN POSITIVE INTEGERS.

TIMESTAMP: 1984-12-15 20:13:38

CATEGORIES: 5 NUMERIC CALCULATION

DESCRIPTION:
X←C DIOPHANTINE A

<DIOPHANTINE> ASSUMES THAT <C> IS A SINGLE POSITIVE INTEGER AND THAT <A> IS A VECTOR OF POSITIVE INTEGERS. THE RESULT IS A VECTOR OF ALL POSITIVE INTEGER SOLUTIONS OF THE EQUATION C = A+.×X.

- -

TITLE: HUI/GLOBAL.1

TYPE: FUNCTION

SUMMARY: RETURNS A MATRIX OF GLOBAL IDENTIFIERS REFERENCED BY A GIVEN FUNCTION

TIMESTAMP: 1984-07-25 15:30:29

CATEGORIES: 28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
SYNTAX : G ← GLOBAL 'F'
<F> IS THE QUOTED NAME OF THE FUNCTION TO BE PROCESSED

FUNCTION : RETURNS A MATRIX <G> OF GLOBAL IDENTIFIERS REFERENCED BY THE GIVEN FUNCTION <F>. THE FUNCTION DOES NOT CHECK ⍎ STATEMENTS.

NOTES : 1. ⎕IO IS LOCALIZED AND SET TO 1
2. UTILITY WRITTEN BY HUI, WHO HAS LEFT IPSA, AND SUBMITTED BY MK

TITLE: *HUI/INWORDS.1* — *HUI/INWORDS.1*

TYPE: *FUNCTION*

SUMMARY: *CONVERTS AN INTEGER INTO ENGLISH WORDS*

TIMESTAMP: 1984-07-20 16:38:10

CATEGORIES: 25 *FORMATTING*
27 *TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS*

DESCRIPTION:
SYNTAX : W←INWORDS I
<I> IS A SINGLE NON-NEGATIVE INTEGER

FUNCTION : CONVERTS A NON-NEGATIVE INTEGER INTO ITS ENGLISH WORD EQUIVALENT.

EXAMPLE: INWORDS 23456 ←→ TWENTY-THREE THOUSAND, FOUR HUNDRED AND FIFTY-SIX.

IT IS ASSUMED THAT I < 10*18.

NOTE : UTILITY WRITTEN BY HUI, WHO HAS SINCE LEFT IPSA, AND SUBMITTED BY MK.

- -

TITLE: *HUI/MAVG.1* — *HUI/MAVG.1*

TYPE: *FUNCTION*

SUMMARY: *COMPUTES THE K-STEP MOVING AVERAGE ON THE LAST AXIS OF AN ARRAY.*

TIMESTAMP: 1984-12-15 20:13:48

CATEGORIES: 5 *NUMERIC CALCULATION*
38 *BUSINESS AND FINANCE*
39 *STATISTICS AND PROBABILITY*

DESCRIPTION:
R←K MAVG A

THE RESULT IS THE K-STEP MOVING AVERAGE ON THE LAST AXIS OF THE NUMERIC ARRAY <A>.

<MAVG> IS ⎕IO-INDEPENDENT.

TITLE: HUI/PASTE.1

TYPE: FUNCTION

SUMMARY: GIVEN TWO VECTORS OF FORMATTED TEXT, WILL PUT THEM TOGETHER SIDE BY SIDE.

TIMESTAMP: 1984-12-15 20:14:00

CATEGORIES:
2 STRUCTURAL TRANSFORMATIONS ◘ RESHAPING, CATENATING, TRANSPOSING, ETC.
25 FORMATTING
27 TEXT PROCESSING ◘ E.G. SPELLING CHECKERS

DESCRIPTION:
Z←A PASTE B

<A> AND <B> SHOULD BE VECTORS OF FORMATTED TEXT, SUCH AS THE TEXT FROM A MAILBOX MESSAGE. THE RESULT IS A VECTOR OF TEXT THAT, WHEN DISPLAYED, SHOWS <A> AND <B> SIDE BY SIDE, AS SHOWN BELOW.

```
      A
THIS IS SOME SAMPLE TEXT.  IT IS
A VECTOR, WITH EMBEDDED CARRAIGE
RETURNS AND POSSIBLY SOME
BACKSPACES.
      B
THIS IS SOME OTHER TEXT.
THE UNDERLINING IN THE
OTHER TEXT WAS DONE WITH
BACKSPACES.  NOTE HOW THE
ALIGNMENT IS STILL
CORRECT.
      A PASTE B
THIS IS SOME SAMPLE TEXT.  IT IS     THIS IS SOME OTHER TEXT.
A VECTOR, WITH EMBEDDED CARRAIGE     THE UNDERLINING IN THE
RETURNS AND POSSIBLY SOME            OTHER TEXT WAS DONE WITH
BACKSPACES.                          BACKSPACES.  NOTE HOW THE
                                     ALIGNMENT IS STILL
                                     CORRECT.
```

<PASTE> IS ORIGIN-1 DEPENDENT.

TITLE: HUI/PERMINV.1

TYPE: FUNCTION

SUMMARY: COMPUTES INVERSE PERMUTATIONS.

TIMESTAMP: 1984-12-15 20:14:11

CATEGORIES: 5 NUMERIC CALCULATION
6 SORTING AND GRADING

DESCRIPTION:
Z←PERMINV P

<P> SHOULD BE A NUMERIC ARRAY. EACH ROW OF <P> SHOULD BE A PERMUTATION OF ι¯1↑ρP. EACH ROW OF THE RESULT IS THE INVERSE PERMUTATION OF THE CORRESPONDING ROW OF <P>. AN EXAMPLE IS SHOWN BELOW.

```
      P
8 2 4 3 7 5 1 6 9
      PERMINV P
7 2 4 3 6 8 5 1 9
      P[PERMINV P]
1 2 3 4 5 6 7 8 9
```

<PERMINV> IS ORIGIN-1 DEPENDENT.

- -

TITLE: HUI/POLY.1

TYPE: FUNCTION

SUMMARY: FINDS THE COEFFICIENTS OF A POLYNOMIAL HAVING SPECIFIED ROOTS.

TIMESTAMP: 1984-12-15 20:36:19

CATEGORIES: 5 NUMERIC CALCULATION

DESCRIPTION:
C←POLY X

THE RESULT IS A VECTOR OF POLYNOMIAL COEFFICIENTS, WITH THE HIGHEST POWER FIRST. THE ROOTS OF THE POLYNOMIAL ARE EXACTLY THOSE SPECIFIED IN THE VECTOR <X>.

EXAMPLES: POLY 2 2 ¯0.5 ↔ 1 ¯3.5 2 2; POLY 0J1 0J¯1 ↔ 1 0 1.

<POLY> IS ⎕IO-INDEPENDENT.

TITLE: HUI/RESET.1 HUI/RESET.1

TYPE: FUNCTION

SUMMARY: RESETS STOP AND TRACE VECTORS OF FUNCTIONS.

TIMESTAMP: 1985-01-02 22:56:45

CATEGORIES: 19 EXECUTION MONITORING
26 EXECUTION CONTROL
28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS
29 DEBUGGING
35 WORKSPACE TOOLS ⍝ E.G. WORKSPACE CROSS REFERENCE

DESCRIPTION:
RESET NL

SETS THE STOP (S∆) AND TRACE (T∆) VECTORS OF ALL THE FUNCTIONS IN THE MATRIX NAMELIST <NL> TO ⍳0. FOR EXAMPLE, RESET ⎕NL 3 WOULD CLEAR THE STOP AND TRACE VECTORS OF ALL (VISIBLE) FUNCTIONS IN THE WS.

<RESET> IS ⎕IO-INDEPENDENT.

- -

TITLE: HUI/SORTLOCAL.1 HUI/SORTLOCAL.1

TYPE: FUNCTION

SUMMARY: SORTS THE LOCAL VARIABLES IN A FUNCTION HEADER

TIMESTAMP: 1984-07-20 17:02:49

CATEGORIES: 6 SORTING AND GRADING
28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
<u>SYNTAX</u> : Z←SORTLOCAL S
<S> IS THE NAME OF AN UNLOCKED FUNCTION
<Z> HAS THE SAME RESULTS AS ⎕FX

<u>FUNCTION</u> : SORTS THE LOCAL VARIABLES IN A FUNCTION HEADER

<u>EXAMPLE</u> :
```
      ∇X←FOO IN;Z;D;H;C;R;A
       ....
      ∇

      SORTLOCAL 'FOO'
FOO
      ∇FOO[⎕]∇
    ∇ X←FOO IN;A;C;D;H;R;Z
[1]   ....
    ∇
```

<u>NOTE</u> : 1. UTILITY WILL RETURN ERROR MESSAGES FOR LOCKED FNS OR A VARIABLE INPUT
2. THE UTILITY IS ⎕IO INDEPENDENT
3. UTILITY WRITTEN BY HUI, WHO HAS SINCE LEFT IPSA, SUBMITTED BY MK.

TITLE: *HUI/XREF.1*

TYPE: *FUNCTION*

SUMMARY: *RETURNS A LISTING OF THE XREF OF A FUNCTION*

TIMESTAMP: 1984-07-20 17:31:03

CATEGORIES: 28 *DEFINED FUNCTIONS* ⍝ *UTILITIES DEALING WITH DEFINED FUNCTIONS*

DESCRIPTION:
SYNTAX : *L←P XREF F*
<F> IS NAME OF AN UNLOCKED FUNCTION
<P> IS THE WIDTH OF THE RESULTING LISTING <L>
IT IS OPTIONAL AND DEFAULTS TO ⎕PW

FUNCTION : *RETURNS A LISTING OF THE XREF OF A FUNCTION. THE XREF IS A SORTED LIST OF NAMES (BOTH VARIABLE AND FUNCTION NAMES) FOLLOWED BY THE LINE-NUMBERS WHERE THEY ARE REFERENCED.*

EXAMPLE :

```
      60 XREF 'XREF'
→       4  6 20
⍎       7
⎕CR     4
⎕IO     0  3
⎕NC     5  7
B       0  5  5  9  9  9 10 11 11 11 11 11 12 12 12 14 14 15
       16 17 19 19 20 21 21 22 22 23 31 31
 .
 .
 .
X       0  4 11 13 13 15 15 19 21 22 22 26 28 29 30 31
Y       0  4  4  4  5  5 12 12 13 13 18 18 18 19 19 19 21 22
       22 29 30 31 31
```

NOTE : 1. *⎕IO IS LOCALIZED AND SET TO* 1
2. *XREF WILL INCLUDE THE NAMES OF VARS/FNS REFERENCED IN A ⍎ STATEMENT BUT NOT IF THEY ARE NESTED WITHIN SEVERAL LEVELS OF ⍎.*
3. *UTILTITY WRITTEN BY HUI, WHO HAS SINCE LEFT IPSA, SUBMITTED BY MK.*

TITLE: JEW/WSCOMPARE.1

TYPE: FUNCTION

SUMMARY: COMPARES TWO WORKSPACES.

TIMESTAMP: 1984-10-29 22:59:59

CATEGORIES: 28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS
35 WORKSPACE TOOLS ⍝ E.G. WORKSPACE CROSS REFERENCE

DESCRIPTION:
THIS FUNCTION COMPARES TWO WORKSPACES THAT HAVE BEEN STORED ON FILE BY ITS SUBUTILITY <WSTOFILE>. USE OF <WSCOMPARE> AND <WSTOFILE> IS DEMONSTRATED BELOW.

```
      UDEFINE 'JEW/WSCOMPARE'  ⍝ DEFINE THE UTILITY AND ITS SUBUTILITIES
      )SAVE SOMEPLACE  ⍝ <WSTOFILE> CAN NOW BE )COPY'D FROM THE WS SOMEPLACE

      )LOAD MYFIRSTWS
      )COPY SOMEPLACE WSTOFILE  ⍝ GET <WSTOFILE>, A SUBUTILITY OF JEW/WSCOMPARE
      WSTOFILE  ⍝ CREATES A FILE AND APPENDS THE WS TO IT

      )LOAD MYSECONDWS
      )COPY SOMEPLACE WSTOFILE
      WSTOFILE
```

THERE ARE NOW TWO FILES, EACH HOLDING A WORKSPACE. (INCIDENTLY, THESE FILES MAY BE USED AS SOURCE FILES BY 7 WSDOC.) TO COMPARE THE TWO WORKSPACES....

```
      TIE1 WSCOMPARE TIE2  ⍝ <TIE1> AND <TIE2> ARE THE TIE NUMBERS OF THE FILES
                           ⍝ CREATED BY <WSTOFILE>
```

EXAMPLE OF THE WAY FUNCTIONS ARE COMPARED

```
WS1 CONTAINS:                WS2 CONTAINS:
    ∇DUMMY                       ∇DUMMY
[1]  LINE 1                  [1]  LINE 1
[2] ⍝ COMMENT                [2]  LINE 3
[3]  LINE 3                  [3]  DIFFERENT LINE 4
[4]  LINE 4                  [4]  LINE 5
    ∇                            ∇
```

WSCOMPARE PRODUCES THE FOLLOWING OUTPUT.

```
DUMMY    : TEXT DIFFERS
[Δ2]       ⍝ COMMENT
[Δ4]        LINE 4
[4.001←3]  DIFFERENT LINE 4
[4.002←4]  LINE 5
```

THIS CAN BE THOUGHT OF AS A DESCRIPTION OF HOW TO CHANGE THE WS1 FUNCTION INTO THE WS2 FUNCTION. IN THE ABOVE EXAMPLE, DELETE LINES 2 AND 4 OF THE WS1 FUNCTION AND INSERT LINES 3 AND 4 OF THE WS2 FUNCTION AS LINES 4.001 AND 4.002 RESPECTIVELY.

VARIABLES ARE ALSO COMPARED. THIS INCLUDES THE FOLLOWING SYSTEM VARIABLES: ⎕CT, ⎕IO, ⎕LX, ⎕PP, ⎕PS, ⎕PW, ⎕RL, AND ⎕TRAP. GROUPS ARE NOT COMPARED.

JEW/WSCOMPARE IS BASED ON A WSCOMPARE WRITTEN BY MIKE SYMES, LATER MODIFIED BY JOHN BURGER AND SACHS.

TITLE: LHG/SS.1

TYPE: FUNCTION

SUMMARY: GENERAL STRING SEARCH PRIMITIVE

TIMESTAMP: 1984-05-02 18:52:05

CATEGORIES: 7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
THIS PROGRAM COMPUTES THE LOCATIONS OF A CHARACTER STRING WITHIN A TEXT VECTOR. THE CHARACTER STRING IS ASSUMED TO BE A NON-EMPTY VECTOR.

THE LEFT ARGUMENT TO <SS> IS THE TEXT VECTOR BEING SEARCHED. THE RIGHT ARGUMENT IS THE SUBSTRING BEING SOUGHT. THE RESULT IS AN ORIGIN-SENSITIVE INTEGER VECTOR CONTAINING THE STARTING INDEX OF EACH MATCH OF THE SUBSTRING WITHIN THE TEXT VECTOR.

- -

TITLE: LHG/∆RPLC.1

TYPE: FUNCTION

SUMMARY: GENERAL REPLACE OF ONE STRING BY ANOTHER.

TIMESTAMP: 1984-05-02 18:52:25

CATEGORIES: 7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF
24 EDITING
27 TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS
46 MODIFYING ARRAYS ⍝ INDEXED ASSIGNMENT, SUBSTRING REPLACEMENT, ETC.

DESCRIPTION:
THIS FUNCTION IS A GENERAL STRING REPLACEMENT UTILITY. IT REPLACES ALL OCCURRENCES OF A PARTICULAR TARGET WITHIN A TEXT VECTOR BY ANOTHER STRING. OVERLAPPING OCCURRENCES OF THE TARGET STRING ARE IGNORED, SO THE RESULTING TEXT VECTOR ALWAYS APPEARS AS EXPECTED.

THE LEFT ARGUMENT TO <∆RPLC> IS THE TEXT VECTOR IN WHICH MATCHES ARE TO BE SOUGHT AND MODIFIED. THE RIGHT ARGUMENT IS A TWO-ELEMENT NESTED ARRAY CONTAINING THE TARGET STRING, FOLLOWED BY THE REPLACEMENT STRING. THE RESULT OF THE PROGRAM IS THE FINAL TEXT VECTOR, WITH ALL NON-OVERLAPPING OCCURRENCES OF THE TARGET STRING REPLACED BY THE NEW STRING.

<∆RPLC> IS ORIGIN-1 DEPENDENT.

TITLE: LLF/LISTFNS.1

TYPE: FUNCTION

SUMMARY: FORMATTED LISTING OF FUNCTIONS NAMED IN RIGHT ARGUMENT

TIMESTAMP: 1983-11-18 21:09:04

CATEGORIES: 28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS
35 WORKSPACE TOOLS ⍝ E.G. WORKSPACE CROSS REFERENCE

DESCRIPTION:
LISTFNS PROVIDES A 1⎕FD LISTING OF FUNCTIONS, WITH INTERFUNCTION SPACING AND HEADERS. THE WS LIBRARY NUMBER AND NAME ARE APPENDED AT THE BOTTOM OF THE LIST. IT WORKS IN ⎕IO 0 OR 1; IS PROTECTED AGAINST MOST ERRONEOUS RIGHT ARGUMENTS THE USER MAY SUPPLY.
THE FUNCTION NAMES IN THE RIGHT ARGUMENT MAY BE SUPPLIED AS A VECTOR, WITH THE NAMES SEPARATED BY BLANK OR COMMA, OR AS A MATRIX WITH ONE FUNCTION NAME PER ROW.
/LINDA FURROW (MBOX LLF)

- -

TITLE: LLF/LISTFNSPREFIX.1

TYPE: FUNCTION

SUMMARY: LIST ALL FUNCTIONS IN THE WS WHOSE NAMES BEGIN WITH PREFIX IN RIGHT ARGUMENT

TIMESTAMP: 1984-01-24 16:58:36

CATEGORIES: 28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS
35 WORKSPACE TOOLS ⍝ E.G. WORKSPACE CROSS REFERENCE

DESCRIPTION:
LISTFNSPREFIX PROVIDES A FORMATTED LISTING OF ALL FUNCTIONS IN THE WORKSPACE WHOSE NAMES BEGIN WITH THE PREFIX SPECIFIED IN THE RIGHT ARGUMENT. EACH FUNCTION LIST IS PRECEDED BY BLANK LINES AND A HEADER; THE FUNCTION IS IN 1⎕FD FORMAT; THE WS LIBRARY NUMBER AND NAME, AND THE SPECIFIED PREFIX, ARE PRINTED AT THE BOTTOM OF THE LIST.
IT WORKS IN ⎕IO 0 OR 1; IS PROTECTED AGAINST MOST ERRONEOUS RIGHT ARGUMENTS THE USER MAY SUPPLY.
/LINDA FURROW (MBOX LLF)

TITLE: LLF/LISTGRP.1

TYPE: FUNCTION

SUMMARY: FORMATTED LISTING OF OBJECTS (NOT PACKAGES) IN GROUP NAMED IN RIGHT ARGUMENT

TIMESTAMP: 1983-11-18 21:18:12

CATEGORIES: 28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
LISTGRP PROVIDES A FORMATTED LISTING OF OBJECTS IN THE GROUP, WITH INTEROBJECT SPACING AND HEADERS. THE WS LIBRARY NUMBER AND NAME ARE APPENDED AT THE BOTTOM OF THE LISTING, ALONG WITH THE GROUP NAME. FUNCTIONS ARE DISPLAYED IN 1⎕FD FORMAT; VARIABLES BY ⍕ (THORN) FORMATTING.
LISTGRP WORKS IN ⎕IO 0 OR 1; IS PROTECTED AGAINST MOST ERRORS IN RIGHT ARGUMENT AND GROUP CONTENTS.
/LINDA FURROW (MBOX LLF)

- -

TITLE: MGF/FFIB.1

TYPE: FUNCTION

SUMMARY: FAST FIBONACCI FUNCTION; FFIB (⍳7)-⎕IO ←→ 0 1 1 2 3 5 8

TIMESTAMP: 1984-07-31 07:47:28

CATEGORIES: 1 MISCELLANEOUS
5 NUMERIC CALCULATION

DESCRIPTION:
CALCULATES THE N'TH FIBONACCI NUMBER FOR ANY INTEGER ARRAY ARGUMENT ⍵ WITH 362 ≤ ⌈/,⍵. FRACTIONAL AND NEGATIVE ARGUMENTS ARE ALLOWED, BUT CONSIDERED MEANINGLESS.

EXAMPLE:

```
      FFIB (⍳7)-⎕IO
0 1 1 2 3 5 8
      FFIB 20 50 100 200
6765 1.258626903E10 3.542248482E20 2.80571173E41
```

THE ALGORITHM IS BASED ON THE FACT THAT (FFIB N+1)÷FFIB N APPROACHES 0.5×1+5*0.5 WHEN N APPROACHES INFINITY.
THEREFORE, THE FUNCTION USES NO LOOPS AND NO RECURSION AND RUNS THEREFORE IN CONSTANT TIME FOR ANY SCALAR ARGUMENT. MOREOVER, THE FUNCTION IS OF RANK 0.

TITLE: *MGF/PKGMATCH.1*

TYPE: *FUNCTION*

SUMMARY: *AS R←α≡ω, BUT ACCEPTS ALSO PACKAGES AS ARGUMENTS.*

TIMESTAMP: 1983-08-02 11:00:57

CATEGORIES: 8 *PACKAGES*

DESCRIPTION:

R←α PKGMATCH ω IS IDENTICAL WITH R←α≡ω FOR ARRAYS.
ADDITIONALY, IT TAKES ALSO PACKAGES AS ARGUMENTS.
TWO PACKAGES ARE EQUAL IF THEY CONTAIN THE SAME NAMES AND THE SAME REFERENTS.
NOTE THAT THIS DOES NOT IMPLY THAT 1 ↔ (⎕*PNAMES* α) ≡ ⎕*PNAMES* ω, *SINCE*
THE ORDER OF THE RESULT OF ⎕*PNAMES IS ARBITRARY.*

IF BOTH PACKAGE CONTAIN ONE OR MORE REFERENTS WHICH ARE LOCKED FUNCTIONS
WITH THE SAME NAME, AND EVERYTHING ELSE IS EQUAL, THE PACKAGES ARE
CONSIDERED EQUAL. THIS IS NOT THE CASE IF ONE PACKAGE CONTAINS THE
SAME FUNCTION(S) AS THE OTHER, BUT LOCKED.

PKGMATCH IS SUBJECT TO ⎕*CT.*

TITLE: MGF/TOPOSORT.1

TYPE: FUNCTION

SUMMARY: TOPOLOGICAL SORT OF PRECEDENCE MATRIX.

TIMESTAMP: 1984-03-20 14:40:58

CATEGORIES: 6 SORTING AND GRADING
41 BOOLEAN ARRAYS

DESCRIPTION:

R←TOPOSORT ⍵

TOPOSORT TAKES A SQUARE BOOLEAN MATRIX AS ARGUMENT AND CALCULATES A GRADING INDEX. THE ARGUMENT ⍵ REPRESENTS A PRECEDENCE RELATION, ⍵[I;J] ↔ 1 INDICATES THAT J IS A PREDECESSOR OF I.

THE RESULT R IS AN INTEGER VECTOR THAT CAN SERVE AS AN INDEX ⍵[R;R] SO THAT THE MATRIX IS TOPOLOGICALLY SORTED. IF THERE IS A LOOP IN THE RELATION, THEN (⍴R) < ''⍴⍴⍵, OTHERWISE (⍴R) = ''⍴⍴⍵. THEREFORE THIS FUNCTION CAN SERVE ALSO AS A LOOP DETECTOR.

THE FUNCTION IS ⎕IO SENSITIVE, IN THE SAME WAY AS ⍋ IS.

WATCH OUT: A TOPOLOGICAL ORDER IS NOT UNIQUE, THE ALGORITHM PICKS THE TOPMOST FROM ALL ROWS WITHOUT PREDECESSOR.

APPLICATION: TOPOLOGICAL SORTS HAVE A VARIETY OF APPLICATIONS, THE RELATION CAN FOR EXAMPLE CONTAIN JOBS WHICH REQUIRE PRECEDING COMPLETITION OF OTHER JOBS, THE RESULT PROPOSES THEN AN ORDER IN WHICH THOSE JOBS COULD BE DONE.

ALTHOUGH THE ALGORITHM IS CLASSICAL, THE IMPLEMENTATION IS NEW AND NOT WELL TESTED. THE FUNCTION WAS WRITTEN BY MARTIN GFELLER, MAILBOX 'MGF'.

*** EXECUTE DETAILS UDESCRIBE 'MGF/TOPOSORT.1' FOR MORE INFORMATION

TITLE: MIKE/NTASK/RAN.1

TYPE: FUNCTION

SUMMARY: EXECUTES FUNCTION 'NTASKWORK' AS A RESTARTABLE NTASK. TRANSFERS NAMED ITEMS TO AND FROM NTASK.

TIMESTAMP: 1983-05-25 16:28:55

CATEGORIES: 14 COMMUNICATION BETWEEN TASKS
18 N-TASKS

DESCRIPTION:
MIKE/NTASK/RAN IS USED TO EXECUTE THE FUNCTION 'NTASKWORK' IN AN NTASK ENVIRONMENT. THE RIGHT ARG OF THE FUNCTION ALLOWS SPECIFICATION OF ITEMS [VARS/FNS] TO BE PASSED TO THE NTASK AND TO BE PASSED BACK FROM THE NTASK. [IF ANY,IN EITHER CASE]. ALL THAT IS REQUIRED IS:

1. A COPY OF FUNCTION 'RAN' IN THE MASTER AND SLAVE (NTASK) TASKS
2. ⎕LX←'0 RAN⍵0' IN THE SLAVE (NTASK) TASK

I HAVE AVOIDED REFERING TO 'TTASK' AND 'NTASK' ALTHOUGH THAT IS THE NORMAL SITUATION. HOWEVER THE NTASK INITIATED USING 'RAN' MAY BE INITIATED FROM T,N,B OR STASK. THEREFORE I REFER TO ONE TASK AS THE MASTER AND ONE AS THE SLAVE. THE SLAVE WILL ALWAYS BE AN NTASK BUT THE MASTER CAN BE ANY TASK TYPE.

FOR MORE INFORMATION, SEE THE DETAILS FIELD.

*** EXECUTE DETAILS UDESCRIBE 'MIKE/NTASK/RAN.1' FOR MORE INFORMATION

TITLE: MIKE/STASK/EXEC.1

TYPE: FUNCTION

SUMMARY: EXECUTES RIGHT ARGUMENT AS AN STASK ON ACCOUNT SPECIFIED BY LEFT ARGUMENT.

TIMESTAMP: 1983-05-24 18:16:13

CATEGORIES: 14 COMMUNICATION BETWEEN TASKS
17 S-TASKS

DESCRIPTION:
RESULT←L EXEC R

THE CONTENTS OF THE CHARACTER ARRAY 'R' ARE EXECUTED AS AN STASK ON ACCOUNT SPECIFIED BY 'L' AND ANY 'TERMINAL' OUTPUT CAUSED BY EXECUTING 'R' IS COLLECTED AS A CHARACTER ARRAY IN 'RESULT'.

EG - RESULT←'1234567:LOCK'EXEC')LIB◇⎕LIB ⎕AI[⎕IO]'
'RESULT' WILL CONTAIN THE RESULT OF)LIB FOLLOWED BY ⎕LIB ⎕AI[⎕IO]
THE '◇' IS TREATED AS A STATEMENT SEPARATOR IN THE USUAL WAY.

EG - RESULT←'1234567:LOCK'EXEC')LOAD 666 BOX◇SEND◇TO MIKE◇SOME TEXT◇ ◇SEND◇ '
'RESULT' WILL CONTAIN THE USUAL 'FILED' MESSAGE, A MESSAGE WILL HAVE BEEN SENT FROM THE CODE BELONGING TO ACCOUNT 1234567 TO 'MIKE'. THE MESSAGE WILL BE 'SOME TEXT'.

THE LEFT ARGUMENT 'L' CAN BE EITHER 'ACCOUNT:LOCK' OR ACCOUNT [NUMERIC] IF THE LATTER, YOU WILL BE PROMPTED FOR THE PASSWORD. EG - 1234567 EXEC'⎕AI'

THE RIGHT ARGUMENT 'R' CAN BE ''. IF THIS IS THE CASE YOU ARE LEFT IN IMMEDIATE EXECUTION WITHIN THE STASK. EXIT FROM THIS BY TYPING)OFF WHILE IN THE STASK,
THE INPUT PROMPT IS ACCOUNT TASKID: FOR EXAMPLE 1234567 4321:
FROM THIS SITUATION YOU COULD RUN EXEC'' AGAIN, AS IN '7654321:LOCK'EXEC''
THE INPUT PROMPT WILL THEN BE -
7654321 2343:: NOTE THAT THERE ARE NOW 2 COLONS. IE 2 LEVELS DEEP.

IF THE LEFT ARGUMENT 'L' IS OMITTED, IT IS TAKEN TO BE THE ACCOUNT NUMBER YOU ARE CURRENTLY SIGNED ON TO. 'NUMBER IN USE' WILL USUALLY BE THE RESULT UNLESS THE ACCOUNT HAS MULTI-TTASK PERMISSION OR YOU ARE RUNNING EXEC IN AN N OR B TASK.

IF THE FIRST CHARACTER OF 'R' IS * THEN)CLEAR IS AUTOMATICALLY EXECUTED AS THE FIRST COMMAND TO THE STASK.

TO PREVENT A COMMAND RETURNING OUTPUT TO 'RESULT' PRECEED IT BY THE CHARACTER +

TO PRINT THE RESULT OF A COMMAND WITHIN EXEC AN NOT SEND IT TO 'RESULT', PRECEED IT BY THE CHARACTER +

TO SEND AN ATTENTION AFTER A COMMAND HAS BEEN EXECUTED, MAKE ! THE LAST CHARACTER OF THE COMMAND.

EG - RESULT←'1234567:LOCK'EXEC'*+)LOAD 666 BOX◇UNREAD◇+PREVIEW◇)MSG 4321 HI!'

BREAK (SHORT DELAY) BREAK ALSO SENDS AN ATTENTION TO THE STASK.

TITLE: *MJAB/ALPHABETIZE.1* *MJAB/ALPHABETIZE.1*

TYPE: *FUNCTION*

SUMMARY: *ALPHABETIZES MATRIX NAMELIST PUTTING 'B̲AT' BETWEEN 'BAT' AND 'CAT'.*

TIMESTAMP: 1983-03-29 13:57:53

CATEGORIES:
6 *SORTING AND GRADING*
27 *TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS*

DESCRIPTION:
ALPHABETIZES A LIST OF WORDS (MATRIX RIGHT ARGUMENT) USING AN ALPHABET WHICH CASUSES AN UNDERLINED CHARACTER TO BE RIGHT AFTER THE CORRESPONDING NONUNDERLINED CHARACTER INSTEAD OF AFTER ALL UNDERLINE CHARACTERS AS APL SORTS THINGS. ALSO, SPACE IS AHEAD OF 'A'.

- -

TITLE: *MJAB/BSTABLE.2* *MJAB/BSTABLE.2*

TYPE: *ARRAY*

SUMMARY: *TABLE OF VALID OVERSTRIKES*

TIMESTAMP: 1984-01-31 23:29:23

CATEGORIES:
1 *MISCELLANEOUS*
20 *TERMINAL INPUT/OUTPUT*

DESCRIPTION:
CONTAINS THE VALID OVERSTRIKES IN BOTH A-BS-B AND B-BS-A FORM.

- -

TITLE: *MJAB/BSTRANSLATE.1* *MJAB/BSTRANSLATE.1*

TYPE: *FUNCTION*

SUMMARY: *CHANGES CHARACTER STRING WITH IMBEDDED BACKSPACES USED TO REPRESENT OVERSTIKES INTO TRUE OVERSTRIKES*

TIMESTAMP: 1983-03-31 14:47:20

CATEGORIES:
1 *MISCELLANEOUS*
20 *TERMINAL INPUT/OUTPUT*
24 *EDITING*
31 *FULL-SCREEN HANDLING*

DESCRIPTION:
TRANSLATES A VECTOR OF ASCII CHARACTERS WITH IMBEDDED BACKSPACES USED TO REPRESENT APL CHARACTERS INTO A (SHORTER) VECTOR OF ⎕AV ELEMENTS AND NO IMBEDDED BACKSPACES. USEFUL FOR TRANSLATING SOMETHING READ FROM A SCREEN OR TRANSMITTED BY SOME DEVICE INTO A FORM USABLE BY 1 ⎕FD.

TITLE: *MJAB/ENC.1* *MJAB/ENC.1*

TYPE: *FUNCTION*

SUMMARY: *BREAKS UP SIMPLE TEXT VECTOR INTO ENCLOSED VECTOR OF WORDS.*

TIMESTAMP: 1983-03-29 15:06:39

CATEGORIES:
2 *STRUCTURAL TRANSFORMATIONS* ⍝ *RESHAPING, CATENATING, TRANSPOSING, ETC.*
3 *PARTITIONED ARRAY HANDLING*
4 *ENCLOSED ARRAYS*
24 *EDITING*

DESCRIPTION:
ACCEPT A STRING OF COMMANDS AND SEPERATE THEM SO THEY CAN BE USED WITH DYADIC ⍳ ETC. FIRST USED IN MASSPORT'S STATISTICAL REPORTING SYSTEM. WILL BE REPLACED BY THINGS LIKE CUT OPERATORS ONE DAY. SEPERATES AT COMMAS BECAUSE MASSPORT WAS A MAGIC BASED SYSTEM AND MAGIC SERIES CODES ARE SEPERATED BY COMMAS.

- -

TITLE: *MJAB/FDEFINE.1* *MJAB/FDEFINE.1*

TYPE: *FUNCTION*

SUMMARY: *TAKES VECTOR OF ASCII CHARACTERS REPRESENTING AN APL FUNCTION AND DEFINES FUNCTION.*

TIMESTAMP: 1983-03-31 14:53:06

CATEGORIES:
27 *TEXT PROCESSING* ⍝ *E.G. SPELLING CHECKERS*
28 *DEFINED FUNCTIONS* ⍝ *UTILITIES DEALING WITH DEFINED FUNCTIONS*

DESCRIPTION:
TAKES A VECTOR OF ASCII CHARACTERS SUCH AS MIGHT BE OBTAINED BY READING THE SCREEN OF A CRT WHICH CONTAINED THE DEFINITION OF A FUNCTION (PERHAPS FROM A MAILBOX MESSAGE) AND DEFINES THE FUNCTION.

TITLE: *MJAB/FDELETE.1* *MJAB/FDELETE.1*

TYPE: *FUNCTION*

SUMMARY: *TIES ALL FILES ON OWNERS ACCOUNT AND LOOPS THROUGH ASKING WHETHER TO DELETE THEM.*

TIMESTAMP: 1983-03-30 13:17:58

CATEGORIES: 9 *FILES*
13 *FILE TOOLS*

DESCRIPTION:
HANDY WHEN THERE ARE A LOT OF FILES YOU WANT TO DELETE ON AN ACCOUNT AND THEY ALL HAVE NAMES LIKE '∆9745H374Q' WHICH YOU DON'T FEEL LIKE TYPING. THE ⎕SIZE OF THE FILE IS DISPLAYED ALONG WITH ITS NAME TO LET YOU NO HOW MUCH YOU ARE DELETING. A YES ANSWER MEANS TO ERASE THE FILE. RCM'S ASK FN IS USED, BUT SINCE THIS IS A UTIL FOR THE PROGRAMMER AND NOT MEANT TO BE PART OF AN APPLICATION, NO CHECKING IS DONE FOR 'HELP' AND 'STOP'.

TITLE: *MJAB/HDS/CHARCHANGE.1* *MJAB/HDS/CHARCHANGE.1*

TYPE: *FUNCTION*

SUMMARY: *CHANGES MESSAGE CHARACTER ON HDS108*

TIMESTAMP: 1983-03-31 11:28:12

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
31 *FULL-SCREEN HANDLING*

DESCRIPTION:
⎕ARBOUTS SEQUENCE WHICH CHANGES MESSAGE CHARACTER ON HDS108. LEFT ARG IS OLD CHAR. RIGHT ARG IS NEW ONE.

TITLE: *MJAB/HDS/FINDPOS.1* *MJAB/HDS/FINDPOS.1*

TYPE: *FUNCTION*

SUMMARY: *REPORTS POSITION OF CURSOR ON HDS108 SCREEN.*

TIMESTAMP: 1983-03-31 11:33:15

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
31 *FULL-SCREEN HANDLING*

DESCRIPTION:
OFTEN YOU NEED TO KNOW WHERE THE CURSOR IS IN ORDER TO KNOW HOW TO GET TO WHERE YOU WANT TO GO NEXT. THIS RETURNS THE ROW AND COL.

TITLE: *MJAB/HDS/KEYCODE.1*

TYPE: *FUNCTION*

SUMMARY: *TRANSLATES HDS*108 *KEY NUMBER INTO ⎕ARBOUT CODE FOR THAT KEY.*

TIMESTAMP: 1983-03-30 14:02:41

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
31 *FULL-SCREEN HANDLING*

DESCRIPTION:
TO DEFINE FN KEYS UNDER FUNCTION CONTROL YOU MUST REFER TO THEM BY A HIGHLY FORGETABLE NUMBER THAT DOES NOT CORRESPOND TO THE KEY NUMBER IN A STRAIGHT-FORWARD WAY. THE ARGUMENT TO THIS FUNCTION IS A 2 *ELEMENT VECTOR. THE FIRST IS A* 0 *OR* 1 *INDICATING LOWER OR UPPER CASE. THE SECOND IS THE KEYNUMBER ON THE HDS*108 *KEYBOARD. THESE RANGE FROM* ¯4 *(INSERT) TO* 14 *(F*14*). THE RESULT IS THE NUMBER USED TO REFER TO THAT KEY IN FNKEY DEFINITION.*

TITLE: *MJAB/HDS/KEYSET.1*

TYPE: *FUNCTION*

SUMMARY: *SETS A KEY ON THE HDS* 108.

TIMESTAMP: 1983-03-30 14:18:42

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
31 *FULL-SCREEN HANDLING*

DESCRIPTION:
LEFT ARGUMENT IDENTIFIES THE KEY. THE LEFT ARGUMENT IS A 2*-ELEMENT VECTOR. THE FIRST ELEMENT IS* 0 *FOR UNSHIFTED OR* 1 *FOR SHIFTED. THE SECOND ELEMENT IS THE KWY NUMBER. KEYS ARE NUMBERED FROM* ¯4 *(THE INSERT KEY) TO* 14 *(THE KEY LABELED F*14*). THE RIGHT ARGUMENT IS EITHER A STRING OF CHARACTERS IN WHICH CASE PUSHING THE KEY WILL BE EQUIVELANT TO TYPING THOSE CHAACTERS AT THE KEYBOARD, OR ELSE IT IS A NUMERIC VECTOR IN WHICH CASE PUSHING THE KEY WILL BE EQUIVELENT TO ⎕ARBOUTING THOSE NUMBERS.*

E.G.
```
0 13 KEYSET ')LOAD 666 BOX'   ⍝SET UNSHIFTED F13 TO SAVE YOU TYPING
1 13 KEYSET 27 34             ⍝SET SHIFTED F13 TO GIVE YOU 132 COLUMN DISPLAY
```

WHEN USING CHARACTER ARGUMENTS, REMEMBER THAT TO THE HDS ⌽ *IS THREE CHARS.*

TITLE: MJAB/HDS/MOVECURSOR.1 MJAB/HDS/MOVECURSOR.1

TYPE: FUNCTION

SUMMARY: RETURNS ⎕ARBOUT SEQUENCE NEEDED TO MOVE CURSOR ON HDS108 TO POSITION IN ARGUMENT.

TIMESTAMP: 1983-03-31 11:48:30

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
31 FULL-SCREEN HANDLING

DESCRIPTION:
SEE COMMENTS IN FN BODY.

- -

TITLE: MJAB/HDS/SAVESCREEN.2 MJAB/HDS/SAVESCREEN.2

TYPE: FUNCTION

SUMMARY: SAVES THE CONTENTS OF THE HDS108 SCREEN AS AN APL CHARACTER VECTOR.

TIMESTAMP: 1983-03-31 13:39:48

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
31 FULL-SCREEN HANDLING

DESCRIPTION:
VERY HANDY FOR READING INFORMATION OFF THE SCREEN (WHICH MAY HAVE BEEN TYPED IN LOCAL MODE, BE OUTPUT FROM A PROGRAM ON ANOTHER SYSTEM, ETC.) PROMPT FOR POSITION TO START AT APPEARS IN STATUS LINE. USER POSITIONS CURSOR AND PRESSES CARRIAGE RETURN TO START TRANSMISSION. EVERYTHING BETWEEN THE CURSOR POSITION AND THE END OF TEXT IS TRANSMITTED AND STORED IN THE VECTOR RESULT. CARRIAGE RETURNS ARE IMBEDDED FOR LINE BREAKS.

- -

TITLE: MJAB/HDS/SENDSCREEN.2 MJAB/HDS/SENDSCREEN.2

TYPE: FUNCTION

SUMMARY: SENDS CHARACTERS FROM HDS SCREEN TO APL.

TIMESTAMP: 1983-03-31 14:07:44

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
31 FULL-SCREEN HANDLING

DESCRIPTION:
LEFT ARGUMENT IS XY COORDINATES OF POSITION TO START SENDING FROM.
RIGHT ARGUMENT IS POSITION OF END OF TEXT TO BE SENT.
FUNCTION SENDS THE CHARACTERS ON THE SCREEN TO APL.
PRIMARILY USEFUL AS A SUBUTILITY OF FUNCTIONS WHICH READ THE SCREEN IN ORDER TO DO SOMETHING WITH THE DATA ON IT.

TITLE: MJAB/HDS/SENDUPTO.2 MJAB/HDS/SENDUPTO.2

TYPE: FUNCTION

SUMMARY: SENDS CHARACTER FROM CURRENT CURSOR POSITION TO ARGUMENT POSITION.

TIMESTAMP: 1983-03-31 14:14:13

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
31 FULL-SCREEN HANDLING

DESCRIPTION:
LEFT ARGUMENT CONTROLS WHETHER ALL CHARACTERS OR ONLY UNPROTECTED CHARACTERS ARE SENT. 4 SEND ONLY UNPROTECTED, 6 SEND ALL.
RIGHT ARGUMENT IS XY-COORDINATE OF POSITION TO SEND UP TO. THE POSITION AT X,Y IS NOT SENT.

- -

TITLE: MJAB/HSPASK.5 MJAB/HSPASK.5

TYPE: FUNCTION

SUMMARY: PROMPTS USER FOR LOCATION (REMOTE OR TORONTO) TO PRINT. GETS DELIVERY INSTRUCTIONS. SUBMITS HSPREQ.

TIMESTAMP: 1983-03-29 15:20:43

CATEGORIES: 1 MISCELLANEOUS
9 FILES

DESCRIPTION:
USED BY AN APPLICATIONS PROGRAMMER TO REQUEST THE INFORMATION NEEDED TO SUBMIT A HIGHSPEED PRINT REQUEST FOR THE USER OF THE APPLICATION.

LEFT ARGUMENT (IF PRESENT): THE PRINT SPECS REQUIRED BY THE APPLICATION (OTHER THAN REMO). DEFAULTS TO 'FAST,ERAS'.

RIGHT ARGUMENT: THE TIE NUMBER OF THE FILE TO BE PRINTED.

RESULT: THE RESULT RETURNED BY THE 'HSP' FUNCTION

THE USER IS PROMPTED FOR A LOCATION AT WHICH TO PRINT THE OUTPUT. IF THEY ANSWER 'HELP' OR WITH A RESPONSE WHICH DOES NOT CORRESPOND TO A CURRENTLY VALID REMOTE NODE, THEY ARE SHOWN ALL THE POSSIBILITIES. THE HSP SYSTEM IS DIRECTLY INTEROGATED TO OBTAIN THE NODE LIST SO IT SHOULD ALWAYS BE UP TO DATE.

<HSPASK> ASSUMES THE FUNCTION <HSP> IS VISIBLE IN THE WORKSPACE.

SINCE PCB'S PERMISIVE SEARCHER 'PCB/INDEX' IS USED, MISSPELLINGS ARE HANDLED BY LOOKING FOR A NEAR MISS AND ASKING THE USER TO VERIFY IT. FOR INSTANCE, THE ENTRY 'AMST' WOULD YIELD THE PROMPT 'DID YOU MEAN AMSD?'

TITLE: MJAB/ON.1 MJAB/ON.1

TYPE: FUNCTION

SUMMARY: CREATES A MATRIX RESULT WITH ALL ROWS OF LEFT ARG BEFORE ALL ROWS OF RIGHT.

TIMESTAMP: 1983-03-29 13:28:16

CATEGORIES: 2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.

DESCRIPTION:
FOR ARGUMENTS OF RANK 0 1 OR 2. ALWAYS YIELDS RANK 2 RESULT. NEVER HAS MORE ROWS IN RESULT THAN THE SUM OF THE NUMBER OF ROWS IN EACH ARGUMENT (UNLIKE SOME ON FUNCTIONS). ASSUMES ARGUMENTS ARE BOTH NUMERIC OR BOTH CHARACTER.

TITLE: MJAB/TEXTLOOP.1 MJAB/TEXTLOOP.1

TYPE: FUNCTION

SUMMARY: ACCEPTS MULTIPLE LINES OF TEXT.

TIMESTAMP: 1983-03-29 13:15:56

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
21 TERMINAL INPUT

DESCRIPTION:
TO BE USED AS A SUBUTILITY. COLLECTS MULTIPLE LINES OF TEXT FROM THE USER. RESULT IS A VECTOR WITH CARRIAGE RETURNS TO SEPERATE LINES. STARTS ACCEPTING ⍞ ENTRY WHEN CALLED. STOPS ACCEPTING TEXT AFTER USER TYPES TWO SUCCESIVE CARRIAGE RETURNS.

TITLE: MK/LOCATE.1

TYPE: FUNCTION

SUMMARY: MATCHES A WORD WITH A STRING OF WORDS, RETURNING ITS LOCATION IF FOUND

TIMESTAMP: 1984-07-20 18:26:16

CATEGORIES: 7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF

DESCRIPTION:

SYNTAX : INDEX←WORD LOCATE STRING
<WORD> IS ANY CHARACTER STRING WITH NO IMBEDDED BLANK SPACES
<STRING> IS A STRING OF WORDS (SIMILAR TO WORD ABOVE), SEPARATED BY A SINGLE SPACE
<INDEX> IS NUMERIC AND IS
¯2 FOR AMBIGUOUS MATCH OF <WORD> IN <STRING>
¯1 FOR <WORD> UNMATCHED IN <STRING>
0 FOR <WORD> BLANK OR NULL
INDEX OR INDICES OF <WORD> LOCATED IN <STRING>ING

FUNCTION : <LOCATE> CAN BE USED IN 2 WAYS :

1. TO FIND THE INDEX OF WORD IN STRING
EG.
```
INPUT←'HELP'
INPUT LOCATE 'STOP HELP DELETE MOVE COPY'
2
```
THE LOCATION OF <INPUT> IN THE STRING OF WORDS IS RETURNED

2. TO FIND THE WORDS IN STRING WHICH MATCH <WORD>
EG.
```
STRING←'THE QUICK BLUE FOX JUMPS OVER THE LAZY DOGGY'
'THE' LOCATE STRING
1 7
```
THE LOCATION(S) OF <WORD> IN THE GIVEN STRING IS RETURNED

NOTE : 1. ⎕IO IS LOCALIZED AND SET TO 1

2. LOCATE SEARCHES FOR AN EXACT MATCH OF THE WORD IN STRING BEFORE IT SEARCHES FOR ABBREVIATIONS.
EG.
```
    'A' LOCATE 'AAABB A AACCC' ↔ 2
   'AA' LOCATE 'AAABB A AACCC' ↔ ¯2  ⍝ AMBIGUOUS
  'AAA' LOCATE 'AAABB A AACCC' ↔ 1
```

3. THIS UTILITY SUBMITTED BY M.K.LEONG.

TITLE: MRAB/FORMATΔTS.1

TYPE: FUNCTION

SUMMARY: CONVERTS ⎕TS-STYLE TIMES INTO ISO-COMPATIBLE CHARACTER REPRESENTATIONS.

TIMESTAMP: 1984-02-11 03:50:56

CATEGORIES: 25 FORMATTING
37 TIMES AND DATES

DESCRIPTION:
SYNTAX: R←FORMATΔTS ω

WHERE ω IS AN N-DIMENSIONAL ARRAY OF NUMBERS CONTAINING (×/¯1↑ρω)
DATES AND TIMES ALONG ITS LAST DIMENSION; EACH TIMESTAMP
IS A VECTOR OF INTEGERS IN THE USUAL ⎕TS ORDER, OR
FIVE INTEGERS FOLLOWED BY A REAL NUMBER OF SECONDS (WITH
FRACTIONAL PART POSSIBLY NON-ZERO), AND
R IS A (1⌈ρρω)-DIMENSIONAL ARRAY OF CHARACTERS REPRESENTING
THE CORRESPONDING DATES IN ω.

EXAMPLES:

```
      ρ⎕←FORMATΔTS 1984 2 10 21 48 4 37
1984-02-10 21:48:04.037
23

      ρ⎕←FORMATΔTS 1984 2 10 21 48 4.037 ⍝ ALTERNATE FORM
1984-02-10 21:48:04.037                  ⍝ OF FIRST EXAMPLE
23

      ρ⎕←FORMATΔTS 2 3ρ1984 2 2 1967 7 1
1984-02-02
1967-07-01
2 10

      ρ⎕←FORMATΔTS 0 0 0 2  ⍝ NO DATE, TIME-OF-DAY ONLY
02:00
5
```

THE FUNCTION SELECTS AN APPROPRIATE FORMAT TO FIT THE MOST DEMANDING TIMESTAMP ACCORDING TO THE LAST DIMENSION OF THE ARGUMENT, THE PRESENCE OR ABSENCE OF POSITIVE NUMBERS IN THE YEAR, MONTH AND DAY POSITIONS, AND OF A NON-ZERO FRACTIONAL PART IN THE SECONDS POSITION WHEN 6↔¯1↑ρω. NEGATIVE NUMBERS ARE TREATED AS ZEROES. FRACTIONAL PARTS (OTHER THAN IN THE SIXTH POSITION IFF 6↔¯1↑ρω) ARE IGNORED. IF 7≤¯1↑ρω, THE SEVENTH POSITION IS TREATED AS MILLISECONDS, AND ANYTHING THAT MIGHT FOLLOW IS IGNORED. THE WIDTH OF THE YEAR FIELD IN THE RESULT IS DETERMINED BY THE HIGHEST YEAR IN THE ARGUMENT.

THE POSSIBILITIES ARE AS FOLLOW:

'YEAR-MM-DD'	''
'YEAR-MM-DD HH:MN'	'HH:MN'
'YEAR-MM-DD HH:MN:SS'	'MM:MN:SS'
'YEAR-MM-DD HH:MN:SS.SSS'	'HH:MN:SS.SSS'

<FORMATΔTS> IS ⎕IO-INDEPENDENT AND SELF-SUFFICIENT.

MICHEL BOUCHARD, I.P.SHARP ASSOCIES LTEE, OTTAWA (MAILBOX: MRAB)

TITLE: MRAB/NAME∆SORT.1

TYPE: FUNCTION

SUMMARY: RETURNS VECTOR OF ROW INDICES TO SORT A MATRIX OF NAMES, TEL.DIRECTORY-STYLE

TIMESTAMP: 1984-01-20 22:32:53

CATEGORIES: 6 SORTING AND GRADING

DESCRIPTION:
SYNTAX: R←NAME∆SORT ⍵

WHERE ⍵ IS A 2-DIMENSIONAL MATRIX OF CHARACTERS WHOSE ROWS ARE NAMES TO BE SORTED, AND
R IS A VECTOR OF INDICES SUCH THAT ⍵[R;] WILL BE A MATRIX OF ALPHABETICALLY SORTED NAMES.

THIS FUNCTION APPROXIMATES THE KIND OF SORTING USED IN MOST TELEPHONE DIRECTORIES IN THE FOLLOWING WAYS:

- HYPHENS AND APOSTROPHES ARE COMPLETELY IGNORED,
- ANY SUB-STRING, WITHIN A NAME, THAT IS FORMED OF NON-ALPHANUMERIC CHARACTERS (I.E. CHARACTERS NOT IN 'ABC...Z01...9') IS TREATED AS A SINGLE SPACE,
- LEADING DEFINITE ARTICLES (ENGLISH AND FRENCH: THE, L', LE, LA, LES) ARE IGNORED. (THE FUNCTION CAN BE ADJUSTED FOR OTHER LANGUAGES.)

SHORTCOMINGS AND DIFFERENCES FROM BELL CANADA'S ALGORITHM:

- IN ITS PRESENT FORM, THE FUNCTION ONLY SUPPORTS ONE FONT: THE REGULAR APL ALPHABET, NOT UNDERSCORED. IF MULTIPLE-FONT SUPPORT IS DESIRED, A TRANSLITERATION OF THE ARGUMENT WOULD BE PREFERABLE TO AN ADJUSTMENT OF THE CODE.
- ABBREVIATIONS AND NUMBERS ARE NOT SPELLED OUT PRIOR TO SORTING; DIGITS ARE TREATED AS POSITIONS 27 THROUGH 36 OF THE ALPHABET, AND AN ABBREVIATION SUCH AS 'ST.' (FOR 'SAINT') WILL BE LISTED BETWEEN 'SS' AND 'SU'.
- THE FUNCTION IS LIKELY TO BLOW UP ON 'WS FULL' IF ITS ARGUMENT IS LARGE RELATIVE TO UNUSED WS (I.E. IF ⎕WA<3×4 ⎕WS'⍵' ⍝ GUESSTIMATE).

THIS FUNCTION IS BEHAVIORALLY IDENTICAL TO 'DONW/NAME∆SORT.2', BUT USES A DIFFERENT ALGORITHM, AND IS COMPLETELY SELF-SUFFICIENT (I.E. DOES NOT REQUIRE SUB-UTILITIES).

MICHEL BOUCHARD, I.P. SHARP ASSOCIES LTEE, OTTAWA (MAILBOX: MRAB)

TITLE: MRAB/ΔBOX.1

TYPE: FUNCTION

SUMMARY: SIMULATION OF APL.68000'S ⎕BOX; MAKE MATRIX FROM VECTOR OR VECTOR FROM MATRIX

TIMESTAMP: 1984-03-09 21:02:14

CATEGORIES: 2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.
25 FORMATTING

DESCRIPTION:
SYNTAX: R ← ΔBOX ω
OR: R ← α ΔBOX ω

WHERE ω IS THE OBJECT TO BE TRANSFORMED; IF ω IS A SCALAR OR A VECTOR, ΔBOX RETURNS A 2-DIMENSIONAL ARRAY, AND IF THE RANK OF ω IS TWO OR MORE, ΔBOX RETURNS A VECTOR.

α IS AN OPTIONAL SCALAR OR VECTOR OF ONE OR TWO ELEMENTS INDICATING A SEPARATOR AND A FILL ELEMENT, IN THAT ORDER; DEFAULT α ↔ 2↑0ρω

R IS THE EXPLICIT RESULT; IF 2≤ρρω, R IS A VECTOR OF THE ROWS OF ω WITH TRAILING FILL ELEMENTS (α[2]) REMOVED, AND EACH ROW SEPARATED FROM THE NEXT BY α[1]; IF 1≥ρρω, R IS A MATRIX WHOSE ROWS ARE THE SUB-STRINGS CONTAINED WITHIN TWO SEPAPATORS (α[1]), FILLED IF NECESSARY WITH α[2].

EXAMPLES:

```
      ⎕←Z←ΔBOX'HELLO OUT THERE'
HELLO
OUT
THERE
      ρZ
3 5
      ρ⎕←ΔBOX Z
HELLO OUT THERE
15
      '*'ΔBOX Z
HELLO*OUT*THERE
      '/∘'ΔBOX'THIS//IS/THE///STORY'
THIS∘
∘∘∘∘∘
IS∘∘∘
THE∘∘
∘∘∘∘∘
∘∘∘∘∘
STORY
      ρΔBOX''
1 0
      ¯1 0 ΔBOX 3 5ρ(1 2 3 0 0),(4 4 4 4 4),(0 0 5 6 7)
1 2 3 ¯1 4 4 4 4 4 ¯1 0 0 5 6 7
```

ΔBOX IS THOROUGHLY TESTED AND ⎕IO-INDEPENDENT.

MICHEL BOUCHARD, I.P. SHARP ASSOCIES LTEE, OTTAWA (MAILBOX: MRAB)

TITLE: MRAB/ΔPCR.1

TYPE: FUNCTION

SUMMARY: EXTRACT MATRIX REPRESENTATION OF A FUNCTION FROM A PACKAGE (SIMILAR TO ⎕PVAL)

TIMESTAMP: 1984-03-15 00:20:14

CATEGORIES:
8 PACKAGES
28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
SYNTAX: R ← ⍺ ΔPCR ⍵

WHERE: ⍵ IS A PACKAGE

⍺ IS A CHARACTER VECTOR (OR SCALAR) CONTAINING ONE NAME WHOSE REFERENT IN ⍵ IS A FUNCTION

R IS THE MATRIX REPRESENTATION OF THE FUNCTION IN ⍵ NAMED IN ⍺
(IF THE FUNCTION IS LOCKED, THEN R ↔ 0 0⍴' ')

BASICALLY, ΔPCR IS TO PACKAGED FUNCTIONS WHAT ⎕PVAL IS TO PACKAGED VARIABLES.
IT IS WELL TESTED, SELF-SUFFICIENT, AND HAS NO SIDE EFFECTS.
MICHEL BOUCHARD, I.P. SHARP ASSOCIES LTEE., OTTAWA (MAILBOX: MRAB)

TITLE: MRAB/ΔPVM.1

TYPE: FUNCTION

SUMMARY: VECTOR TO MATRIX ACCORDING TO PARTITIONING BOOLEAN VECTOR

TIMESTAMP: 1984-04-18 20:51:43

CATEGORIES:
2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.
3 PARTITIONED ARRAY HANDLING
25 FORMATTING

DESCRIPTION:
SYNTAX: R←⍺ ΔPVM ⍵

WHERE
⍵ IS A VECTOR, CHARACTER OR NUMERIC, AND
⍺ IS A BOOLEAN VECTOR OF OF LENGTH (⍴⍵) IN WHICH THE 1'S CORRESPOND TO THE FIRST ELEMENT OF EACH PARTITION.
R IS A MATRIX WITH AS MANY ROWS AS THERE ARE PARTITIONS, AND AS MANY COLUMNS AS NEEDED TO SATISFY THE LONGEST PARTITION. PARTITIONS ARE LEFT JUSTIFIED, AND PADDED WITH (1↑0⍴⍵).

NOTE: THE FIRST ELEMENT OF THE LEFT ARGUMENT IS IGNORED AS IT IS PRESUMED TO BE 1 (I.E. THE FIRST ELEMENT OF THE FIRST PARTITION IS ALWAYS THE FIRST OF THE RIGHT ARGUMENT.

EXAMPLE:

```
      1 0 0 1 1 0 ΔPVM ⍳6
1 2 3
4 0 0
5 6 0
```

MICHEL BOUCHARD, I.P. SHARP ASSOCIES LTEE, OTTAWA [MAILBOX: MRAB]

TITLE: MTH/UNIQUEROWS.1

TYPE: FUNCTION

SUMMARY: FINDS ALL UNIQUE ROWS IN A TWO-DIMENSIONAL CHARACTER MATRIX

TIMESTAMP: 1984-05-15 03:16:42

CATEGORIES: 7 SEARCHING ∩ INCLUDING MEMBERSHIP AND INDEX-OF
40 SELECTING FROM ARRAYS ∩ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
THE FUNCTION ACCEPTS A RIGHT ARGUMENT OF A TWO-DIMENSIONAL CHARACTER MATRIX. ANY OTHER RIGHT ARGUMENT TO THE FUNCTION WILL CAUSE A RETURN TO IMMEDIATE EXECUTION. THE RESULT IS A TWO-DIMENSIONAL CHARACTER MATRIX OF ALL UNIQUE ROWS FOUND IN THE RIGHT ARGUMENT.

EXAMPLE:

```
          T←4 4⍴'TESTDOG TESTCAT '
          T
    TEST
    DOG
    TEST
    CAT
          UNIQUEROWS T
    TEST
    DOG
    CAT
```

THE FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: PCB/CH∆RAVEL.1

TYPE: FUNCTION

SUMMARY: REPRESENTS AN ARRAY AS CHARACTER VECTOR, TRAILING BLANKS REMOVED.

TIMESTAMP: 1983-03-08 13:42:46

CATEGORIES: 2 STRUCTURAL TRANSFORMATIONS ∩ RESHAPING, CATENATING, TRANSPOSING, ETC.

DESCRIPTION:
REPRESENTS AN ARRAY AS CHARACTER VECTOR, TRAILING BLANKS REMOVED. THE ARRAY MAY BE OF ANY RANK, WITH EXTRA CRS TO INDICATE BREAKS BETWEEN PLANES, ETC.

TITLE: PCB/DATEDECODE.1 PCB/DATEDECODE.1

TYPE: FUNCTION

SUMMARY: RETURNS CHARACTER ARRAY OF DATES IN STANDARD FORMAT, FROM ARRAY OF DAYNUMBERS.

TIMESTAMP: 1983-02-27 23:50:31

CATEGORIES: 37 TIMES AND DATES

DESCRIPTION:
CONVERTS JULIAN DAYNUMBERS TO STANDARD CHARACTER REPRESENTATIONS, IN ANY ORDER (YY/MM/DD DD/MM/YY OR MM/DD/YY, AS SPECIFIED BY OPTIONAL LEFT ASRGUMENT). REPRESENTS 0 IN THE ARGUMENT BY BLANK IN THER RESULT. REPRESENTS THE DAYNUMBER {PAST} BY THE CHARACTERS 'PRIOR' AND THE DAYNUMBER {FUTURE} BY THE CHARACTERS 'CURRENT'.

- -

TITLE: PCB/DATEENCODE.1 PCB/DATEENCODE.1

TYPE: FUNCTION

SUMMARY: RETURNS NUMERIC ARRAY OF JULIAN DAYNUMBERS, FROM CHARACTER ARRAY OF FORMATTED DATES.

TIMESTAMP: 1983-02-27 22:53:42

CATEGORIES: 37 TIMES AND DATES

DESCRIPTION:
CALCULATES JULIAN DAYNUMBERS FROM STANDARD CHARACTER FORMS. THE ORDER (DD MM YY, MM DD YY, ETC) EXPECTED IN THE ARGUMENT IS SELECTED BY THE OPTIONAL LEFT ARGUMENT. THE FUNCTION RETURNS DAYNUMBER 0 FOR A DATE WHOSE REPRESENTATION IS BLANK. IT RETURNS THE LARGEST REPRESENTABLE INTEGER FOR A DATE REPRESENTED BY THE CHARACTERS 'CURRENT' AND THE SMALLEST REPRESENTABLE INTEGER (STORED AS THE GLOBAL {PAST} AND {FUTURE} FOR A DATE REPRESENTED IN THE ARGUMENT BY THE CHARACTERS 'PRIOR' . THIS FUNCTION IS THE INVERSE OF {PCB/DATEDECODE}. THE NUMERIC PART OF THE CALCULATION IS DONE BY {PCB/DNO}.

- -

TITLE: PCB/DATEREP.1 PCB/DATEREP.1

TYPE: FUNCTION

SUMMARY: RETURNS 3-ELEMENT REPRESENTATION OF EACH OF ARRAY OF JULIAN DAYNUMBERS

TIMESTAMP: 1983-02-27 20:40:37

CATEGORIES: 37 TIMES AND DATES

DESCRIPTION:
CONVERTS JULIAN DAYNUMBERS TO CONVENTIONAL NUMERIC REPRESENTATIONS. THESE ARE RETURNED IN THE FORM YYYY MM DD UNLESS A DIFFERENT ORDER IS INDICATED BY THE OPTIONAL LEFT ARGUMENT. (TO CONVERT TO CHARACER FORM WITH SLASHES ETC, USE THE FN PCB/DATEDECODE, WHICH CALLS THIS FN FOR THE NUMERIC WORK.) GENERAL DISCUSSION OF THESE DATE FUNCTIONS IS IN THE 'DESCRIBE' OF WS 880 DATES.

TITLE: PCB/DEFAULT.1 PCB/DEFAULT.1

TYPE: FUNCTION

SUMMARY: LEFT AGRUMENT IS CHARACTER VECTOR CONTAINING NAME OR EXPRESSION.

TIMESTAMP: 1983-02-27 20:15:42

CATEGORIES: 1 MISCELLANEOUS

DESCRIPTION:
RIGHT ARGUMENT PROVIDES DEFAULT VALUE TO BE EMPLOYED IF THE NAME OR EXPRESSION IN THE LEFT ARGUMENT IS ABSENT OR INVALID.

- -

TITLE: PCB/DISF.3 PCB/DISF.3

TYPE: FUNCTION

SUMMARY: FORMATTED DISPLAY OF FUNCTIONS IN PACKAGE ARGUMENT

TIMESTAMP: 1983-11-23 16:53:27

CATEGORIES: 28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
FORMATS FUNCTION DISPLAYS SO LABELS, BODY, AND COMMENTS HAVE AN ALIGNMENT COMMON TO ALL THE FUNCTIONS PACKED IN THE RIGHT ARGUMENT. THE OPTIONAL LEFT ARGUMENT IS A NUMERIC VECTOR SETTING THE NUMBER OF CHARACTERS PERMITTED FOR LINE NUMBERS, FOR LABELS, AND FOR THE BODY OF THE FUNCTION (COMMENTS ARE WHATEVER IS LFT AFTER THAT).VERSION 3 RETURNS A MATRIX RESULT, 1 ROW PER LINE OF THE FUNCTION.

- -

TITLE: PCB/DNO.1 PCB/DNO.1

TYPE: FUNCTION

SUMMARY: RETURNS JULIAN DAYNUMBERS FOR ARRAY OF DATES. LEFT ARG SPECIFIES DATE FORMAT.

TIMESTAMP: 1983-02-27 20:25:31

CATEGORIES: 37 TIMES AND DATES

DESCRIPTION:
TAKES ARRAY OF DATES REPRESENTED AS 3-ELEMENT VECTORS YYYY MM DD (OR OTHER PERMUTATION INDICATED BY THE OPTIONAL LEFT ARGUMENT) AND RETURNS AN ARRAY IN WHICH EACH DATE IS REPRESENTED BY A SCALAR JULIAN DAYNUMBER.
(NOTE: THESE ARE TRUE JULIAN DAYNUMBER, WITH ORIGIN ABOUT 4000 BC.)

TITLE: *PCB/FNEXTRACT.1* *PCB/FNEXTRACT.1*

TYPE: *FUNCTION*

SUMMARY: *EXTRACT ⎕CR OF A FUNCTION IN A PACKAGE.*

TIMESTAMP: 1983-02-28 00:51:36

CATEGORIES: 8 *PACKAGES*
28 *DEFINED FUNCTIONS* ⍝ *UTILITIES DEALING WITH DEFINED FUNCTIONS*

DESCRIPTION:
LEFT ARGUMENT IS NAME OF FUNCTION, RIGHT ARGUMENT PACKAGE CONTAINING THAT FUNCTION. RESULT IS ⎕CR OF THE FUNCTION.

\- -

TITLE: *PCB/FNNAME.1* *PCB/FNNAME.1*

TYPE: *FUNCTION*

SUMMARY: *EXTRACTS FUNCTIONS NAME FROM ITS CANONICAL REPRESENTATION*

TIMESTAMP: 1983-02-28 00:40:30

CATEGORIES: 28 *DEFINED FUNCTIONS* ⍝ *UTILITIES DEALING WITH DEFINED FUNCTIONS*

DESCRIPTION:
SCANS SYNTAX OF HEADER LINE TO PICK OUT THE FUNCTION NAME.

\- -

TITLE: *PCB/FNPACK.1* *PCB/FNPACK.1*

TYPE: *FUNCTION*

SUMMARY: *RETURNS PACKAGE CONTAINING FUNCTION FROM ⎕CR ARGUMENT*

TIMESTAMP: 1983-02-28 00:40:44

CATEGORIES: 8 *PACKAGES*
28 *DEFINED FUNCTIONS* ⍝ *UTILITIES DEALING WITH DEFINED FUNCTIONS*

DESCRIPTION:
ARGUMENT IS CANONICAL REPRESENTATION OF A SINGLE FUNCTION. RESULT IS A PACKAGE CONTAINING THAT DEFINITION AS A FUNCTION (RATHER THAN AS AN ARRAY).

TITLE: PCB/INDEX.1

TYPE: FUNCTION

SUMMARY: RETURN ROWS OF MATRIX ⍺ WHERE ROWS OF MATRIX ⍵ FOUND, WITH SIMILARITY CHECKING.

TIMESTAMP: 1983-03-03 08:37:46

CATEGORIES: 7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF

DESCRIPTION:
THIS IS A VERY ELABORATE TABLE LOOKUP, WITH LOTS OF OPTIONS, INCLUDING THE OPTION TO MATCH PERFECTLY, MATCH FROM THE BEGINNING, OR MATCH ANYWHERE, AND FOR INTERACTIVE CHECKING OF SIMILAR ENTRIES WHEN A MATCH IS NOT FOUND, AND PROVISION TO AMEND THE MASTER LIST TO INCLUDE MISSING ENTRIES.

USED FOR LOOKING UP VECTOR (OR MATRIX) ARGUMENT TO IDENTIFY AT WHICH ROW OF THE 'MASTER' MATRIX THE ITEMS IN THE RIGHT ARGUMENT ARE FOUND.

INCLUDES PROVISION FOR CONVERSATIONAL CHECKING OF MISSPELLINGS, AND TO AMEND THE MASTER MATRIX FOR MISSING ITEMS.

LEFT ARGUMENT: NAME (IN QUOTES) OF MATRIX REFERENCE TABLE
RIGHT ARGUMENT: VECTOR OR MATRIX OF ITEMS TO BE LOOKED UP

CONDITIONED BY 4-ELEMENT VECTOR FC AND DON'T-CARE VECTOR DC. DC MAY CONTAIN PLACE-HOLDER CHARACTERS FOR WHICH ANYTHING IS AN ACCEPTABLE MATCH

FC[0] 0: MATCH ANYWHERE
1: MATCH FROM THE START
2: MATCH COMPLETELY
FC[1] 0: MASTER MAY CONTAIN DUPLICATE ENTRIES; RETURN LOCATIONS OF ALL MATCHES
1: ROWS OF MASTER ARE UNIQUE
2: ROWS OF MASTER ARE UNIQUE AND GUARANTEED TO CONTAIN ITEMS OF RIGHT ARG
FC[2] 0: RETURN RESULT REGARDLESS
1: PROMPT USER IF RESULT NOT FOUND OR DISCREPANT
FC[3] 0: LEAVE MASTER UNCHANGED
1: IF ITEM NOT FOUND, AMEND MASTER TO INCLUDE IT
2: AMEND, AND REPORT AMENDMENT
FOR ELEMENTS EXPECTED BUT NOT FOUND, RETURNS (1↑⍴MASTER)+⎕IO

RESULT IS VECTOR WITH ONE ELEMENT FOR EACH ITEM IN THE RIGHT ARGUMENT. HOWEVER, WHEN MASTER DOES NOT CONTAIN UNIQUE ROWS, RIGHT ARGUMENT MAY CONTAIN ONLY ONE ENTRY (AS VECTOR OR ONE-ROW MATRIX) AND LENGTH OF RESULT IS NUMBER OF INSTANCES FOUND.

TITLE: PCB/MATRIX.1 PCB/MATRIX.1

TYPE: FUNCTION

SUMMARY: ASSURES RESULT IS A MATRIX

TIMESTAMP: 1983-03-05 10:05:07

CATEGORIES: 2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.

DESCRIPTION:
RESHAPE AN ARRAY ARGUMENT SO THAT THE RESULT IS ALWAYS A MATRIX, WITH THE SAME
NUMBER OF COLUMNS AS THE ARGUMENT, AND AS MANY ROWS AS NECESSARY TO ACCOMODATE
ALL THE OTHER AXES.

- -

TITLE: PCB/NOTEMPTY.1 PCB/NOTEMPTY.1

TYPE: FUNCTION

SUMMARY: BOOLEAN WITH 1 WHERE FIRST AXIS CONTAINS NON-ZERO OR NON-BLANK

TIMESTAMP: 1983-02-27 22:32:42

CATEGORIES: 40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.
41 BOOLEAN ARRAYS

DESCRIPTION:
LOCATE ROWS OF A MATRIX (OR ELEMENTS OF A VECTOR, ETC) WHICH ARE ALL BLANK
OR ALL ZERO (USUALLY SO THEY CAN BE DISREGARDED DURING PROCESSING).

- -

TITLE: PCB/NUB.1 PCB/NUB.1

TYPE: FUNCTION

SUMMARY: ELIMINATE DUPLICATES

TIMESTAMP: 1983-03-05 10:07:03

CATEGORIES: 40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
RAVELS ARGUMENT, RETURNS VECTOR OF DISTINCT ELEMENTS

TITLE: PCB/ON.1

TYPE: FUNCTION

SUMMARY: JOINS TWO ARRAYS ONE ABOVE THE OTHER, CONVERTING TYPE WHERE THEY ARE MIXED

TIMESTAMP: 1983-02-27 20:08:19

CATEGORIES: 2 STRUCTURAL TRANSFORMATIONS ¤ RESHAPING, CATENATING, TRANSPOSING, ETC.

DESCRIPTION:
WHERE ONE ARGUMENT IS EMPTY, RETURNS THE OTHER. WHERE NEITHER ARGUMENT IS EMPTY, BUT TYPES DIFFER, CONVERTS NUMERIC TO CHARACTER. WHERE ONE ARGUMENT IS NARROWER, WIDENS IT TO MATCH THE WIDTH OF THE OTHER. RESULT IS A MATRIX (UNLESS AN EMPTY VECTOR IS JOINED TO A VECTOR).

- -

TITLE: PCB/RDCIDECODE.2

TYPE: FUNCTION

SUMMARY: RETURNS CHARACTER ARRAY OF FORMATTED DATES AND TIMES FOR ⎕RDCI TIMESTAMPS

TIMESTAMP: 1983-11-23 16:35:59

CATEGORIES:
9 FILES
13 FILE TOOLS
37 TIMES AND DATES

DESCRIPTION:
SHOWS YEAR,MONTH,DAY WITH SLASHES, HOUR, MIN, SEC, MILLISECOND FROM NUMERIC ARGUMENT ENCODED AS 60THS OF AS SECOND SINCE MARCH 1, 1960.

REVIEWS:
[1984-07-29 22:29:26]

THERE SEEMS TO BE A SLIGHT PROBLEM WITH IT.

```
      RDCIDECODE 45609034080
1984/04/02  0:56:07.***
```

(THE NUMBER 45609034080 WAS TAKEN FROM THE RESULT OF ⎕RDCI.) WGR/FTT WORKS CORRECTLY; IT RETURNS A SEVEN-ELEMENT VECTOR OF INTEGERS, ⎕TS-STYLE. DJK/FORMATTS OR MRAB/FORMAT∆TS COULD BE USED TO CONVERT THIS TO A CHARACTER ARRAY.

--DOUGLAS J. KEENAN

TITLE: *PCB/RDCIENCODE.1*

TYPE: *FUNCTION*

SUMMARY: *RETURNS ⎕RDCI ENDCODINGS FOR ARRAY OF DATES AND TIMES IN ⎕TS FORMAT*

TIMESTAMP: 1983-02-28 00:17:40

CATEGORIES:
9 *FILES*
13 *FILE TOOLS*
37 *TIMES AND DATES*

DESCRIPTION:
LAST AXIS OF ARGUMENT IS ⎕TS; IF SUPPLIED WITH LAST AXIS HAVING LENGTH LESS THAN 7, TRAILING ZEROS ARE ASSUMED. RESULT IS ENCODING IN 60THS OF A SECOND SINCE MARCH 1 1960 (AS USED IN ⎕RDCI).

TITLE: PESCH/FILE.1

TYPE: FUNCTION

SUMMARY: ALLOWS EXPRESSIONS WITH FILE PRIMITIVES REFERRING TO FILES BY NAME

TIMESTAMP: 1983-03-26 01:02:29

CATEGORIES: 9 FILES
43 FILE PRIMITIVE SIMULATION

DESCRIPTION:
SYNTAX: TIENO←[HOW] FILE NAME

IF A FILE IS TIED, ''FILE'' WILL REPORT ITS TIE NUMBER. OTHERWISE IT WILL TIE IT TO A NEW TIE NUMBER, AND RETURN THAT AS RESULT.

THIS PERMITS WRITING EXPRESSIONS USING THE STANDARD SHARP APL FILE PRIMITIVES THAT REFER TO FILES BY NAME RATHER THAN TIE NUMBER: FOR EXAMPLE, I MIGHT READ COMP. 3 OF THE FILE '504 UTERMS' WITH THE EXPRESSION

```
      ⎕READ(FILE'504 UTERMS'),3
```

TIE NUMBERS CHOSEN BY ''FILE'' ARE ALWAYS GREATER THAN OR EQUAL TO 500000 (AND AS CLOSE TO THAT NUMBER AS POSSIBLE).

WHEN INVOKED DYADICALLY, THE LEFT ARGUMENT CONTROLS HOW TO TIE THE FILE. AN EMPTY LEFT ARGUMENT RESULTS IN EXCLUSIVE TIE (⎕TIE) BEING USED; A LEFT ARGUMENT 'S' RESULTS IN SHARE-TIE (⎕STIE) BEING USED. OTHER LEFT ARGUMENTS WILL CAUSE ERRORS.

MONADIC USE IS EQUIVALENT TO DYADIC USE WITH THE LEFT ARGUMENT 'S'.

IF THE RIGHT ARGUMENT REFERS TO A NONEXISTENT OR INACCESSIBLE FILE, 0 IS RETURNED AS A RESULT.

''FILE'' AVOIDS EXHAUSTING THE FILE TIE QUOTA BY UNTIEING ANOTHER FILE IF NECESSARY. THEREFORE FILE SHOULD ONLY BE USED IN SYSTEMS WHICH EITHER:

(A) ALWAYS REFER TO A FILE THROUGH THIS FUNCTION WHEN USING IT; E.G.,

```
      ⎕READ(FILE FID),N    RATHER THAN    ⎕READ TNO,N
```

(B) USE EVENT TRAPPING TO INVOKE THIS FUNCTION WHENEVER A FILE NEEDED TURNS OUT TO HAVE BEEN UNTIED.

SEE 'PESCH/TIE' FOR A SLIGHTLY CHEAPER VARIANT OF THIS FUNCTION.

TITLE: PESCH/TIE.1

TYPE: FUNCTION

SUMMARY: CREATES OR FINDS TIE NUMBER GIVEN FILE NAME; AVOIDS EXHAUSTING FILE TIE QUOTA

TIMESTAMP: 1983-03-26 00:44:12

CATEGORIES: 9 FILES
43 FILE PRIMITIVE SIMULATION

DESCRIPTION:
SYNTAX: TIENO←[HOW] TIE STDNAME

IF A FILE IS TIED, TIE WILL REPORT ITS TIE NUMBER.
OTHERWISE IT WILL TIE IT TO A NEW TIE NUMBER, AND RETURN THAT AS RESULT.
TIE NUMBERS CHOSEN BY TIE ARE ALWAYS GREATER THAN OR EQUAL TO 500000.

WHEN INVOKED DYADICALLY, THE LEFT ARGUMENT CONTROLS HOW TO TIE THE FILE.
AN EMPTY LEFT ARGUMENT RESULTS IN EXCLUSIVE TIE (⎕TIE) BEING USED; A LEFT
ARGUMENT 'S' RESULTS IN SHARE-TIE (⎕STIE) BEING USED. OTHER LEFT ARGUMENTS
WILL CAUSE ERRORS.

MONADIC USE IS EQUIVALENT TO DYADIC USE WITH THE LEFT ARGUMENT 'S'.

IF THE RIGHT ARGUMENT REFERS TO A NONEXISTENT OR INACCESSIBLE FILE, OR IF IT IS
NOT IN STANDARD FILE-NAME FORM, 0 IS RETURNED AS A RESULT.

TIE AVOIDS EXHAUSTING FILE TIE QUOTA BY UNTIEING ANOTHER FILE IF NECESSARY.
THEREFORE TIE SHOULD ONLY BE USED IN SYSTEMS WHICH EITHER:
(A) ALWAYS REFER TO A FILE THROUGH TIE WHEN USING IT; E.G.,
⎕READ(TIE FID),N RATHER THAN ⎕READ TNO,N
(B) USE EVENT TRAPPING TO INVOKE TIE WHENEVER A FILE NEEDED TURNS OUT TO
HAVE BEEN UNTIED.

**NOTE: TIE ASSUMES ITS ARGUMENT IS IN 'STANDARD FORM', AS PRODUCED E.G.
BY ROHAN/MAKE∆STD∆FILENAME IN THIS UTILITY LIBRARY. SEE 'PESCH/FILE' FOR
A VERSION OF TIE THAT ACCEPTS ANY FORM OF FILE NAME.

TITLE: PKI/ADMPRINT.1 PKI/ADMPRINT.1

TYPE: FUNCTION

SUMMARY: USES ADMPRINT TO QUEUE A PRINT REQUEST, FOR PROCESSING ON A 3279.

TIMESTAMP: 1984-11-14 17:25:19

CATEGORIES:
20 TERMINAL INPUT/OUTPUT
31 FULL-SCREEN HANDLING
45 GRAPHICS AND PLOTTING
47 FULL-SCREEN GRAPHICS

DESCRIPTION:
SUBMITS A PRINT REQUEST TO A 3287 PRINTER. THE RIGHT ARGUMENT IS THE TEXT TO BE PRINTED, THE LEFT ARGUMENT IS THE ADDRESS OF THE PRINTER TO WHICH YOU WISH THE OUTPUT TO BE DIRECTED. FOR EXAMPLE, TO PRINT A MATRIX OF TEXT TO BE PRINTED ON 3287 IN THE ZOO TERMINAL ROOM:

```
      'IC250H15' ADMPRINT 20 80⍴'EXAMPLE OF TEXT TO BE PRINTED'
```

TITLE: PKI/ARBIO/OVERSTRIKES.3 PKI/ARBIO/OVERSTRIKES.3

TYPE: ARRAY

SUMMARY: AN INTEGER MATRIX SPECIFYING WHICH PAIRS OF APL CHARACTERS ARE VALID OVERSTRUCK CHARACTERS

TIMESTAMP: 1984-11-27 16:22:31

CATEGORIES:
20 TERMINAL INPUT/OUTPUT
21 TERMINAL INPUT
27 TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS

DESCRIPTION:
AN INTEGER MATRIX OF SHAPE 3,N SPECIFYING WHICH PAIRS OF APL CHARACTERS ARE VALID APL OVERSTRIKES.

THE FIRST TWO ROWS CONTAIN THE ORIGIN-0 INDICES OF THE CHARACTERS THAT MAKE UP THE OVERSTRIKE. THE THIRD ROW CONTAINS THE ORIGIN-0 INDICES OF THE CHARACTERS THAT ARE FORMED BY BEING OVERSTRUCK.

```
FOR EXAMPLE, ⎕AV[⎕IO+OVERSTRIKES[;⎕IO+47]] ↔ '_AA',
         AND ⎕AV[⎕IO+OVERSTRIKES[;⎕IO+19]] ↔ '\○⍉'.
```

'○\⍉' IS NOT AN ELEMENT OF <OVERSTRIKES>. THE ORDER '\○⍉' IS CHOSEN OVER '○\⍉' SINCE THE FORMER IS WHAT IS PRINTED BY SHARP APL ON OUTPUT.

PKI/ARBIO/OVERSTRIKES IS A MORE COMPLETE TABLE THAN RCM/ARBIO/OVERSTRIKES. IN PARTICULAR, IT CONTAINS SOME VALID COMBINATIONS THAT ARE NOT INCLUDED IN RCM/ARBIO/OVERSTRIKES, SUCH AS '∨∧⋄' AND 'FLE', AS WELL AS COMBINATIONS THAT ARE ALLOWED BUT DO NOT PRODUCE A NEW CHARACTER, SUCH AS '.??' AND ':;;'. NOTE THAT, UNLIKE RCM/ARBIO/OVERSTRIKES, THE THIRD ROW DOES NOT CONTAIN DISTINCT ELEMENTS, SINCE THE SAME CHARACTER CAN BE CREATE USING DIFFERENT OVERSTRIKE COMBINATIONS. FOR EXAMPLE, OVERSTRIKING ',:' PRODUCES ';', AS DOES OVERSTRIKING '.;'.

THE VALUE OF THIS VARIABLE IS EASILY CHANGED VIA RCM/ARBIO/CREATE∆OVERSTRIKE.

TITLE: PKI/FSCLS.1 PKI/FSCLS.1

TYPE: FUNCTION

SUMMARY: CANCELS OR ENQUEUES A FILE FOR PRINTING BY ADMPRINT.

TIMESTAMP: 1984-11-14 17:18:56

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
31 FULL-SCREEN HANDLING
45 GRAPHICS AND PLOTTING
47 FULL-SCREEN GRAPHICS

DESCRIPTION:
THIS FUNCTION IS USED TO CLOSE AN OPEN QUEUED PRINTING DEVICE, USUALLY A 3287. THE RIGHT ARGUMENT IS EITHER 0 OR 1. 0 INDICATES THAT THE PRINT REQUEST IS NOT TO BE PROCESSED. 1 INDICATES THAT THE FILE SHOULD BE SUBMITTED FOR PRINTING. A TYPICAL SEQUENCE OF CALLS MIGHT BE:

```
FSOPEN 'IC250H15' ⍝ OPEN VTAM ID 'IC250H15'
                  ⍝ AT SHARP, THIS IS THE 3287 IN THE ZOO TERMINAL ROOM.
FSLOG TEXT ⍝ ADD TEXT TO PRINT FILE.
GSLOG ⍝ ADD GRAPHICS IMAGE ON CURRENT AP126 PAGE TO PRINT FILE.
FSCLS 1 ⍝ ENQUEUE THE FILE FOR PRINTING BY ADMPRINT.
```

TITLE: PKI/FSLOG.1 PKI/FSLOG.1

TYPE: FUNCTION

SUMMARY: ADDS TEXT TO QUEUED PRINTER FILE OPENED USING FSOPEN.

TIMESTAMP: 1984-11-14 17:17:34

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
31 FULL-SCREEN HANDLING
45 GRAPHICS AND PLOTTING
47 FULL-SCREEN GRAPHICS

DESCRIPTION:
THIS FUNCTION IS USED TO PLACE A VECTOR OR MATRIX OF TEXT TO BE PRINTED ON THE CURRENT OPEN QUEUED PRINTER, USUALLY AN IBM 3287. THE PRINT REQUEST IS SUBMITTED OR CANCELLED BY CALLING FSCLS. FOR EXAMPLE, THE FOLLOWING SEQUENCE OF CALLS SUBMITS A PRINT REQUEST

```
FSOPEN 'IC250H15' ⍝ OPEN VTAM ID 'IC250H15'
                  ⍝ AT SHARP, THIS IS THE 3287 IN THE ZOO TERMINAL ROOM.
FSLOG TEXT ⍝ ADD TEXT TO PRINT FILE.
GSLOG ⍝ ADD GRAPHICS IMAGE ON CURRENT AP126 PAGE TO PRINT FILE.
FSCLS 1 ⍝ ENQUEUE THE FILE FOR PRINTING BY ADMPRINT.
```

TITLE: PKI/FSOPEN.1

TYPE: FUNCTION

SUMMARY: OPEN QUEUED PRINTING DEVICE FOR OUTPUT (USUALLY IBM 3287) USING AP126.

TIMESTAMP: 1984-11-14 17:13:32

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
31 FULL-SCREEN HANDLING
45 GRAPHICS AND PLOTTING
47 FULL-SCREEN GRAPHICS

DESCRIPTION:
THIS FUNCTION CAN BE USED TO OPEN A QUEUED PRINTING DEVICE, SUCH AS AN IBM 3287, TO RECEIVE OUTPUT. THIS FUNCTION USES AP126 TO PERFORM THE CALL.
A TYPICAL USE OF FSOPEN WOULD BE AS FOLLOWS:

```
      FSOPEN 'IC250H15' ⍝ OPEN VTAM ID 'IC250H15'
                        ⍝ AT SHARP, THIS IS THE 3287 IN THE ZOO TERMINAL ROOM.
      FSLOG TEXT ⍝ ADD TEXT TO PRINT FILE.
      GSLOG ⍝ ADD GRAPHICS IMAGE ON CURRENT AP126 PAGE TO PRINT FILE.
      FSCLS 1 ⍝ ENQUEUE THE FILE FOR PRINTING BY ADMPRINT.
```

- -

TITLE: PKI/GDDM.1

TYPE: FUNCTION

SUMMARY: UTILITY TO PERFORM AP126 CALLS, WITH ERROR CHECKING.

TIMESTAMP: 1984-11-13 17:57:35

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
31 FULL-SCREEN HANDLING
45 GRAPHICS AND PLOTTING
47 FULL-SCREEN GRAPHICS

DESCRIPTION:
GDDM IS A DYADIC FUNCTION USED TO PERFORM AP126 CALLS. THE RIGHT ARGUMENT CONSISTS OF A NUMERIC VECTOR OF AP126 CONTROL SEQUENCES. THE LEFT ARGUMENT, IF INCLUDED, IS A CHARACTER VECTOR OF DATA.
GDDM PERFORMS THE SPECIFIED CALL, FIRST ESTABLISHING A SHARE WITH AP126 IF NONE IS EXTANT. VARIABLES CTLG1 AND DATG1 ARE GLOBALS SHARED WITH AP126. VARIABLES CTLG2 AND DATG2, ALSO GLOBAL, CONTAIN RETURN CODES SET BY AP126. IF AN ERROR HAS OCCURRED GDDM WILL QUERY AP126 FOR FURTHER DETAILS OF THE ERROR. DATG2 WILL CONTAIN A DESCRIPTION OF THE ERROR, AND THE ERROR IS SIGNALLED BY EVENT 126. IF AN ATTEMPT IS MADE TO SHARE WITH AP126 FROM A NON-IBM 3270 TYPE TERMINAL, OR IF AP126 IS NOT CURRENTLY RUNNING ON THE SYSTEM, THIS IS SIGNALLED BY EVENT 500.

TITLE: PKI/SHADEAT.1 — PKI/SHADEAT.1

TYPE: FUNCTION

SUMMARY: SHADES LOCUS OF POINTS AT SPECIFIED X LOCATIONS.

TIMESTAMP: 1983-11-25 18:10:56

CATEGORIES: 45 GRAPHICS AND PLOTTING

DESCRIPTION:
SYNTAX: R←X SHADEAT CXY
THIS FUNCTION IS INTENDED FOR USE IN WS 3 SAGA, OR IN ANY OTHER GENERAL PURPOSE GRAPHICS SYSTEM SUCH AS 3 GRAPHICS. THE RIGHT ARGUMENT IS A 3 COLUMN MATRIX OF POINTS. THE SECOND AND THIRD COLUMNS CONTAIN X AND Y COORDINATES. THE FIRST COLUMN WILL BE 0 OR ¯4 TO INDICATE THE START OF EACH NEW POLYGON, 1 OTHERWISE.
THIS IS A 'STANDARD' FORMAT USED THROUGHOUT SHARP GRAPHICS SYSTEMS.
EACH POLYGON DESCRIBED IN THE RIGHT ARGUMENT MUST BE CLOSED - IE. THE FIRST AND LAST POINTS OF EACH POLYGON MUST BE THE SAME. THE LEFT ARGUMENT IS A VECTOR OF X POINTS AT WHICH YOU WANT A VERTICAL SHADING LINE TO BE DRAWN. FOR EXAMPLE, IF VARIABLE POLY IS A 3 COL MATRIX WHICH DESCRIBES A CIRCLE, YOU CAN DRAW VERTICAL LINES INSIDE THE CIRCLE AT THE X VALUES 2 4 AND 5 BY DOING

```
      DRAW 2 4 5 SHADEAT POLY
```

THE RESULT OF SHADEAT IS A 3 COLUMN MATRIX IN THE FORMAT DESCRIBED ABOVE, WHICH MAY CONVENIENTLY BE PASSED TO THE DRAW FUNCTION CONTAINED IN WS 3 SAGA OR 3 GRAPHICS.

TITLE: PLA/ENCVECFRMAT.1 — PLA/ENCVECFRMAT.1

TYPE: FUNCTION

SUMMARY: ENCLOSED VECTOR FROM SIMPLE MATRIX.

TIMESTAMP: 1983-07-29 08:14:17

CATEGORIES: 4 ENCLOSED ARRAYS

DESCRIPTION:
R←ENCVECFRMAT ω

ENCLOSED VECTOR FROM SIMPLE MATRIX. TRAILING BLANKS OR ZEROES ARE REMOVED FROM EACH ROW BEFORE ENCLOSING.

TITLE: PLEB/SC.1

TYPE: FUNCTION

SUMMARY: SHIFTS TRAILING COMMENTS OF FUNCTION TO SPECIFIED COLUMN

TIMESTAMP: 1984-01-24 18:14:28

CATEGORIES: 28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:

R←W SC FN: FN IS A FUNCTION NAME, W IS THE COLUMN NUMBER, R IS THE RESULT OF 3 ⎕FD. SHIFTS ALL TRAILING COMMENTS IN FUNCTION FN RIGHT TO START AT COLUMN W. IF COMMENT STARTS AFTER COLUMN W, IT IS SHIFTED TO THE LEFTMOST POSSIBLE POSITION. ORIGINAL FUNCTION WRITTEN BY SAD.

TITLE: RCM/ARBIO/ARBINPUT.1

TYPE: FUNCTION

SUMMARY: SUBSTITUTE FOR ⍞; GETS INPUT FROM A TERMINAL.

TIMESTAMP: 1983-02-28 14:40:59

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
21 TERMINAL INPUT

DESCRIPTION:
TEXT←ARBINPUT

SUBSTITUTE FOR ⍞; GETS INPUT FROM A TERMINAL. <TEXT> IS THE CHARACTER VECTOR ENTERED AT THE TERMINAL.

THE GLOBAL VARIABLES <OVERSTRIKES> AND <TRANSLATE> ARE USED. THE FORMER SPECIFIES WHICH OVERSTIKES ARE ACCEPTABLE. ACCPTABLE OVERSTRIKES NEED NOT BE APL CHARACTERS; ONLY TWO CHARACTERS MAY BE OVERSTRUCK UPON EACH OTHER, WITH THE EXCEPTION OF O-BS-U-BS-T, WHICH GENERATES AN INTERRUPT IN ⎕ARBIN. <TRANSLATE> IS THE USUAL TRANSLATE TABLE OF ⎕ARBIN.

IF THE USER ENTERS AN UNACCEPTABLE OVERSTRIKE, 'CHARACTER ERROR' IS PRINTED AND THE USER IS PLACED BACK IN INPUT MODE.

DEFAULT VALUES AND DESCRIPTIONS OF <OVERSTRIKES> AND <TRANSLATE> ARE AVAILABLE IN RCM/ARBIO/OVERSTRIKES AND RCM/ARBIO/TRANSLATE.

REVIEWS:
[1984-11-29 13:41:12]

THE OVERSTRIKE TABLE IN 'RCM/ARBIO/OVERSTRIKES' IS FAR FROM COMPLETE. IT DOES NOT INCLUDE SOME LEGAL ENTRIES SUCH AS '∨∧◇', NOR DOES IT INCLUDE PSEUDO-OVERSTRIKE COMBINATIONS SUCH AS ',:;' OR 'EFE'.
MOREOVER, ARBINPUT DOES NOT WORK IF YOU TRY TO ENTER JUST A SINGLE OVERSTRUCK CHARACTER. TO REPRODUCE THIS BUG, TRY

```
      A←ARBINPUT
A,BS,_ ⍝ IE. ENTER 3 KEYSTROKES: A, BACKSPACE, UNDERSCORE.
```

THE RESULT IS A←'' RATHER THAN A←'A'. /PKI

TITLE: RCM/ARBIO/ARBIN∆EDIT.1

TYPE: FUNCTION

SUMMARY: REMOVES 'DELETED' STRINGS IN A VECTOR CONTAINING BACKSPACES AND LINEFEEDS.

TIMESTAMP: 1983-02-28 14:27:05

CATEGORIES: 24 EDITING
27 TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS

DESCRIPTION:
TEXT←ARBIN∆EDIT TEXT

THE INPUT SHOULD BE A CHARACTER VECTOR, TYPICALLY OBTAINED VIA ⎕ARBIN. THE RESULT IS ALSO A CHARACTER VECTOR, WITH CERTAIN SUBSTRINGS IN THE ARGUMENT DELETED. THE DELETED STRINGS ARE THOSE WHICH YOU WOULD EXPECT TO BE DELETED VIA BACKSPACES AND LINEFEEDS. THE BEHVIOUR IS SIMILAR TO THAT OF THE DEL (FUNCTION) EDITOR'S.

THE FUNCTION LOCALIZES ⎕IO AND SETS IT TO 0.

REVIEWS:
[1983-05-05 21:02:00]

ARBIN∆EDIT DOESN'T WORK AS ONE WOULD EXPECT IN THE FOLLOWING CASE:

```
      ARBIN∆EDIT 'ABC',⎕AV[159+⎕IO],'DEF'
ABC
```

I HAVE MENTIONED THIS TO RCM, AND HE IS LOOKING INTO IT.

/ETHAN SEIDEL

TITLE: RCM/ARBIO/ARBIN∆OVERSTRIKE.1

TYPE: FUNCTION

SUMMARY: MAPS A SEQUENCE OF OVERSTRUCK APL CHARS INTO A SINGLE CHAR.

TIMESTAMP: 1983-02-28 14:25:32

CATEGORIES:
20 TERMINAL INPUT/OUTPUT
24 EDITING
27 TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS
31 FULL-SCREEN HANDLING

DESCRIPTION:
TEXT←OVERSTRIKES ARBIN∆OVERSTRIKE TEXT

MAPS A SEQUENCE OF APL CHARACTERS INTO A SINGLE CHARACTER.

THE RIGHT ARGUMENT IS A CHARACTER VECTOR, LIKELY ENTERED VIA ⎕ARBIN. THE RESULT IS SIMILAR TO THE RIGHT ARGUMENT, WITH OVERSTRUCK CHARACTERS MAPPED INTO A SINGLE CHARACTER.

<OVERSTRIKES> IS A MATRIX SPECIFYING WHICH OVERSTRIKES ARE ACCEPTABLE. ITS DEFAULT VALUE IS THE GLOBOL VARIABLE <OVERSTRIKES>. ONLY TWO DISTINCT CHARACTERS MAY BE OVERSTRUCK UPON EACH OTHER.

SEE RCM/ARBIO/OVERSTRIKES FOR THE DEFAULT VALUE AND DESCRIPTION OF <OVERSTRIKES>.

THE RESULT IS EITHER THE APPROPRIATELY ALTERED CHARACTER VECTOR OR AN INTEGER MATRIX OF SHAPE N,2 GIVING THE ORIGIN-0 INDICES OF INVALID OVERSTRUCK CHARACTERS.

THE FUNCTION LOCALIZES ⎕IO AND SETS THE LOCAL ⎕IO TO 0.

REVIEWS:
[1983-05-05 21:21:27]

1) IN THE DESCRIPTION, THERE IS TALK OF A 'DEFAULT' VALUE FOR THE LEFT ARGUMENT, OVERSTRIKES. I THINK 'SAMPLE' VALUE IS WHAT IS MEANT, FOR IF YOU USE THIS FN WITHOUT A LEFT ARGUMENT, YOU GET AN ERROR.

2) THE RIGHT ARGUMENT MUST END WITH A CARRIAGE RETURN.

3) EXTRA BACKSPACES ARE NOT HANDLED WELL:

```
      ⍴OVERSTRIKES ARBIN∆OVERSTRIKE 'A',⎕AV[158],'_',⎕AV[158 156]
2 55
```

I WOULD'VE EXPECTED SOMETHING OF SHAPE N,2 GIVEN THE DESCRIPTION. I GOT OTHER SURPRISING SHAPES FROM INVALID OVERSTRIKES:

```
      ⍴OVERSTRIKES ARBIN∆OVERSTRIKE 'A',⎕AV[158],'B',⎕AV[156]
3 55
```

I'VE MENTIONED THESE PROBLEMS TO RCM, AND HE'S LOOKING INTO IT.

/ETHAN SEIDEL

TITLE: RCM/ARBIO/ARBOUTPUT.1

TYPE: FUNCTION

SUMMARY: PRINTS TEXT AT A TERMINAL; A SUBSTITUTE FOR <⍞←>.

TIMESTAMP: 1983-02-28 14:35:20

CATEGORIES: 20 TERMINAL INPUT/OUTPUT

DESCRIPTION:
ARBOUTPUT TEXT

PRINTS TEXT AT A TERMINAL; A SUBSTITUTE FOR < ⍞← >. <TEXT> MUST HAVE RANK ≤2. IT MAY INCLUDE EMBEDDED CARRIAGE RETURNS AND LINEFEEDS. LINES WITH LENGTH >⎕PW ARE BROKEN INTO SEVERAL LINES, WITH OVERLAP LINES INDENTED SIX SPACES. IDLES ARE INSERTED AND OVERSTRIKES ARE MAPPED INTO SEQUENCES OF NON-OVERSTRUCK CHARACTERS (WITH EMBEDDED BACKSPACES).

THE GLOBAL VARIABLES <OVERSTRIKES>, WHOSE DEFAULT VALUE IS IN RCM/ARBIO/OVERSTRIKES, AND <TRANSLATE>, WHOSE DEFAULT VALUE IS IN RCM/ARBIO/TRANSLATE ARE USED. SEE THEIR DESCRIPTIONS FOR DETAILS ON GETTING THE CORRECT VALUES FOR A PARTICULAR TERMINAL, ETC. THE FUNCTION RCM/ARBIO/ARBOUTPUT WILL INITIALIZE THEM.

- -

TITLE: RCM/ARBIO/ARBOUT∆CARRIAGE.1 RCM/ARBIO/ARBOUT∆CARRIAGE.1

TYPE: FUNCTION

SUMMARY: INSERTS CARRIAGE CONTROL CHARACTERS INTO TEXT; LIKELY USED WITH ⎕ARBOUT.

TIMESTAMP: 1983-02-28 14:26:29

CATEGORIES: 20 TERMINAL INPUT/OUTPUT

DESCRIPTION:
TEXT←CONTROL ARBOUT∆CARRIAGE TEXT

THIS FUNCTION INSERTS CARRIAGE CONTROL CODES INTO THE RIGHT ARGUMENT, WHICH MUST BE CHARACTER AND HAVE RANK ≤2. THE LEFT ARGUMENT IS AN INTEGER VECTOR OF ONE OR TWO ELEMENTS. THE FIRST ELEMENT SPECIFIES THE PRINT WIDTH OF THE OUTPUT DEVICE. IF THE SECOND ELEMENT IS 0 OR NON-EXISTENT, THE NUMBER OF IDLES INSERTED AT THE END OF EACH LINE WILL BE BASED ON THE LENGTH OF THE LINE. IF THE SECOND ELEMENT IS 1, THE NUMBER OF IDLES INSERTED AT THE END OF EACH LINE WILL BE BASED ON THE PRINT WIDTH OF THE OUTPUT DEVICE.

THE FIRST METHOD HAS A SOMEWHAT GREATER CPU COST, BUT A LOWER CHARACTER AND CONNECT CHARGE THAN THE SECOND.

THE FUNCTION LOCALIZES ⎕IO AND SETS THE LOCAL VALUE TO 0.

TITLE: RCM/ARBIO/ARBOUTΔOVERSTRIKE.1

TYPE: FUNCTION

SUMMARY: MAPS A SINGLE OVERSTRUCK APL CHARACTER INTO A SEQUENCE OF NON-OVERSTRUCK APL CHARACTERS.

TIMESTAMP: 1983-02-28 14:25:46

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
31 FULL-SCREEN HANDLING

DESCRIPTION:
TEXT←OVERSTRIKES ARBOUTΔOVERSTRIKE TEXT

MAPS A SINGLE OVERSTRUCK APL CHARACTER INTO A SEQUENCE OF NON-OVERSTRUCK APL CHARACTERS, LIKELY FOR USE WITH □ARBOUT.

THE RIGHT ARGUMENT IS A CHARACTER VECTOR CONTAINING APL CHARACTERS.

THE LEFT ARGUMENT IS AN INTEGER MATRIX OF SHAPE 3,N DESIGNATING WHICH OVERSTRIKES ARE VALID. SEE RCM/ARBIO/OVERSTRIKES FOR THE DEFALUT VALUE AND A COMPLETE DESCRIPTION.

THE RESULT IS THE APPROPRIATELY ALTERED CHARACTER VECTOR.

THE FUNCTION LOCALIZES □IO AND SETS THE LOCAL VALUE TO 0.

TITLE: RCM/ARBIO/OVERSTRIKES.1

TYPE: ARRAY

SUMMARY: AN INTEGER MATRIX SPECIFYING WHICH PAIRS OF APL CHARACTERS ARE VALID OVERSTRUCK CHARACTERS.

TIMESTAMP: 1983-02-28 14:27:54

CATEGORIES:
20 TERMINAL INPUT/OUTPUT
21 TERMINAL INPUT
27 TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS

DESCRIPTION:
AN INTEGER MATRIX OF SHAPE 3,N SPECIFYING WHICH PAIRS OF APL CHARACTERS ARE VALID APL OVERSTRIKES.

THE FIRST TWO ROWS CONTAIN THE ORIGIN-0 INDICES OF THE CHARACTERS THAT MAKE UP THE OVERSTRIKE. THE THIRD ROW CONTAINS THE ORIGIN-0 INDICES OF THE CHARACTERS THAT ARE FORMED BY BEING OVERSTRUCK.

FOR EXAMPLE, ⎕AV[OVERSTRIKES[;⎕IO]] ↔ '_AA', AND ⎕AV[⎕IO+OVERSTRIKES[;⎕IO+32]] ↔ '\○⍀'.

'○\⍀' IS NOT AN ELEMENT OF <OVERSTRIKES>, I.E. THE LAST ROW OF <OVERSTRIKES> CONTAINS DISTINCT ELEMENTS. THE ORDER '\○⍀' IS CHOSEN OVER '○\⍀' SINCE THE FORMER IS WHAT IS PRINTED BY SHARP APL ON OUTPUT.

THE VALUE OF THIS VARIABLE IS EASILY CHANGED VIA RCM/ARBIO/CREATE∆OVERSTRIKE.

REVIEWS:
[1983-05-05 21:31:18]

IN THE DESCRIPTION, THERE IS MENTION OF '○\⍀' VERSUS '\○⍀' IN THIS ARRAY. ONE GETS THE IMPRESSION THAT THIS AFFECTS THE WORKINGS OF ARBIN∆OVERSTRIKE, BUT THIS IS NOT THE CASE. IT IS ALSO STATED THAT THE ORDER OF '\○⍀' IS CHOSEN BECAUSE '\' IS PRINTED BEFORE '○' DURING OUTPUT. IT SEEMS TO ME THAT '○' IS PRINTED BEFORE '\'.

I MENTIONED THESE ITEMS TO RCM, AND HE'S LOOKING INTO THEM.

/ETHAN SEIDEL

TITLE: RCM/ARBIO/TRANSLATE.1

TYPE: ARRAY

SUMMARY: A TABLE OF THE CHARS PRINTED BY CERTAIN TERMINALS UPON RECEIPT OF 7 BIT ASCII TRANSMISSION CODES.

TIMESTAMP: 1983-02-28 14:28:49

CATEGORIES: 20 TERMINAL INPUT/OUTPUT

DESCRIPTION:
A 128 ELEMENT CHARACTER VECTOR. THE VALUES IN THE TABLE ARE REALLY TERMINAL DEPENDENT, AND THE TABLE SHOULD THUS NORMALLY BE OBTAINED FROM RCM/TERMFNS/<TERMINAL>/TRANSLATE. THIS UTILITY EXISTS MAINLY TO DEMONSTRATE THE USE OF THE VARIABLE.

THE ORIGIN-0 POSITION OF AN APL CHARACTER IN THE TABLE INDICATES THE ASCII TRANSMISSION CODE WHICH IS SENT TO THE TERMINAL TO CAUSE THE TERMINAL TO PRINT THE CHARACTER. THAT IS, IF <X> IS A 7 BIT ASCII CODE SENT TO THE TERMINAL, TRANSLATE[⎕IO+2⍳X] YIELDS THE CHARACTER THAT WILL BE DISPLAYED WHEN THE CODE IS RECEIVED BY THE TERMINAL.

- -

TITLE: RCM/ENCLARRAYS/ENCLVECFROMMAT.1

TYPE: FUNCTION

SUMMARY: ENCLOSED VECTOR FROM SIMPLE MATRIX.

TIMESTAMP: 1983-03-01 06:45:40

CATEGORIES: 4 ENCLOSED ARRAYS

DESCRIPTION:
R←ENCLVECFROMMAT ⍵

ENCLOSED VECTOR FROM SIMPLE MATRIX. TRAILING BLANKS OR ZEROES (1↑0⍴⍵) REMOVED FROM EACH ROW BEFORE ENCLOSING.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/ENCLARRAYS/VECFROMENCLVEC.1 RCM/ENCLARRAYS/VECFROMENCLVEC.1

TYPE: FUNCTION

SUMMARY: SIMPLE VECTOR FROM ENCLOSED VECTOR

TIMESTAMP: 1983-03-01 06:45:51

CATEGORIES: 4 ENCLOSED ARRAYS

DESCRIPTION:
R←VECFROMENCLVEC ω

SIMPLE VECTOR FROM ENCLOSED VECTOR.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/EVENTTRAP/ERRORLOG.1 RCM/EVENTTRAP/ERRORLOG.1

TYPE: FUNCTION

SUMMARY: AUTOMATICALLY LOGS IN A FILE ALL ERRORS NOT OTHERWISE TRAPPED

TIMESTAMP: 1983-03-01 06:47:27

CATEGORIES: 44 EVENT TRAPPING

DESCRIPTION:
ERRORLOG

AUTOMATICALLY LOGS IN A FILE ALL ERRORS NOT OTHERWISE TRAPPED

USED WITH <SETUP∆ERRORLOG>

GLOBAL VARS ACCESSED- ERRORLOG

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/EVENTTRAP/EVENTΔREPORT.1

TYPE: FUNCTION

SUMMARY: ANALYZES ⎕ER AND RETURNS (EVENT CODE),(INDEX OF BAD CHAR IN STMT),(⎕AV INDICES OF CHARS IN STMT).

TIMESTAMP: 1983-03-01 06:46:20

CATEGORIES: 44 EVENT TRAPPING

DESCRIPTION:
ER←EVENTΔREPORT

ANALYZES ⎕ER AND RETURNS NUMERIC ERROR REPORT

RESULT- INTEGER VECTOR: EVENT CODE, POINTER TO OFFENDING CHAR IN STATEMENT,
⎕AV INDICES OF CHARS IN STATEMENT

GLOBAL VARS ACCESSED- ⎕ER

THIS FUNCTION WORKS IN EITHER ORIGIN. THE RESULT VARIES APPROPRIATELY
ACCORDING TO THE ORIGIN.

TITLE: RCM/EVENTTRAP/INTERPRET.1

TYPE: FUNCTION

SUMMARY: ARGUMENT IS ⎕ER. EXPLICIT RESULT IS AN ERROR MESSAGE SUITABLE FOR A NAIVE (NON-PROGRAMMER) USER.

TIMESTAMP: 1983-03-01 06:46:31

CATEGORIES: 44 EVENT TRAPPING

DESCRIPTION:
MSG←INTERPRET QER

THE ARGUMENT IS TYPICALLY ⎕ER. THE RESULT IS AN ERROR MESSAGE SUITABLE FOR A NAIVE (NON-PROGRAMMER) USER.

CURRENT POSSIBLE VALUES FOR THE RESULT (<MSG>) ARE THE FOLLOWING.

'THERE IS NOT SUFFICIENT SPACE IN YOUR SEGMENT OF THE COMPUTER''S MEMORY TO DO WHAT YOUR PROGRAM ATTEMPTED TO DO' 'THERE IS NOT SUFFICIENT SPACE IN YOUR SEGMENT OF THE COMPUTER''S AUXILIARY STORAGE TO DO WHAT YOUR PROGRAM ATTEMPTED TO DO' 'YOUR SIGNON ACCOUNT DOES NOT HAVE THE RESOURCES NECESSARY TO DO WHAT YOUR PROGRAM ATTEMPTED TO DO' 'SOME PART OF YOUR PROGRAM WAS NOT WRITTEN CORRECTLY' 'SOME PART OF YOUR PROGRAM WHICH IS USING THE COMPUTER''S AUXILIARY STORAGE IS NOT WRITTEN CORRECTLY' 'SOME PART OF YOUR PROGRAM WHICH IS ATEMPTING TO COMMUNICATE WITH ANOTHER PROGRAM IS NOT WRITTEN CORRECTLY' 'THERE IS A PROBLEM WITH THE MAIN COMPUTER SYSTEM YOU ARE USING' 'YOUR PROGRAM HAS ATTEMPTED TO DO SOMETHING WHICH WOULD VIOLATE SECURITY' 'YOU HAVE ATTEMPTED TO STOP THE EXECUTION OF YOUR PROGRAM' 'INVALID ERROR CODE' (IN 1+⎕FI QER[⎕IO;])

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/EVENTTRAP/SETUP∆ERRORLOG.1* *RCM/EVENTTRAP/SETUP∆ERRORLOG.1*

TYPE: *FUNCTION*

SUMMARY: *SETS UP WS AND FILE SO THAT ALL UNTRAPPED EVENTS MAY BE AUTOMATICALLY TRAPPED AND LOGGED IN A FILE*

TIMESTAMP: 1983-03-01 06:46:43

CATEGORIES: 44 *EVENT TRAPPING*

DESCRIPTION:
SETUP∆ERRORLOG NAME

SETS UP WS AND FILE SO THAT ALL UNTRAPPED ERRORS AND INTERRUPTS MAY BE AUTOMATICALLY TRAPPED AND LOGGED IN A FILE

USED WITH FUNCTION RCM/EVENTTRAP/ERRORLOG

ARGUMENT- CHAR VECTOR NAME OF FILE WHERE ERRORS ARE TO BE LOGGED

FILES ACCESSED- SPECIFIED BY USER
ASSUMES THAT FILE HAS ALREADY BEEN CREATED

GLOBAL VARS ACCESSED- □TRAP,ERRORLOG

SUBROUTINES- INTERPRET

THIS FUNCTION IS □IO-INDEPENDENT.

TITLE: *RCM/FILES/COMPS∆REPORT.2*

TYPE: *FUNCTION*

SUMMARY: *PREPARES A REPORT ON SPECIFIED COMPONENTS OF A FILE*

TIMESTAMP: 1984-01-28 22:04:13

CATEGORIES: 9 *FILES*
13 *FILE TOOLS*

DESCRIPTION:
TABLE←LMT COMPS∆REPORT TIE

PREPARES A REPORT ON SPECIFIED COMPONENTS OF A FILE

RIGHT ARGUMENT- INTEGER SCALAR OR 1 *OR* 2 *ELEMENT INTEGER VECTOR CONTAINING TIE NUMBER, AND PASSNUMBER IF NECESSARY*

LEFT ARGUMENT- FIRST AND LAST COMPONENTS TO BE REPORTED ON. DEFAULT SETTING IS ALL COMPONENTS.

RESULT- CHAR MATRIX TABLE CONTAINING COMPONENT NUMBER, AUTHOR ACCT, TIME WRITTEN, DATA TYPE, SIZE IN BYTES, RANK, AND SHAPE. FOR A PACKAGE, RANK AND SHAPE ARE REPLACED BY A LIST OF THE NAMES IN THE PACKAGE.

AS ORIGINALLY SUBMITTED, THIS FUNCTION DID NOT WORK. IT WAS CORRECTED BY ROHAN JAYASEKERA, I.P. SHARP ASSOCIATES, OTTAWA.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/FILES/DECODE∆TIME.1*

TYPE: *FUNCTION*

SUMMARY: *DECODES SYSTEM WRITE DATE/TIME*

TIMESTAMP: 1983-03-01 04:20:45

CATEGORIES: 9 *FILES*

DESCRIPTION:
TS←DECODE∆TIME TIME

DECODES SYSTEM WRITE DATE/TIME

ARGUMENT- NUMERIC SCALAR OR 1 *ELEMENT VECTOR:* ¯1↑⎕RDCI *TIE,COMPONENT,PASSNUMBER OR* ¯1↑2 ⎕WS 4

RESULT- 7 *ELEMENT NUMERIC VECTOR- YEAR,MONTH,DAY,HOUR,MINUTE,SECOND,SECOND*÷60

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/FILES/FAPPEND.1

TYPE: FUNCTION

SUMMARY: COVER FUNCTION FOR ⎕APPEND- RESIZES FILE IF NECESSARY

TIMESTAMP: 1983-03-01 04:19:02

CATEGORIES: 9 FILES
43 FILE PRIMITIVE SIMULATION

DESCRIPTION:
DATA FAPPEND TIE

COVER FUNCTION FOR ⎕APPEND- RESIZES FILE IF NECESSARY

LEFT ARGUMENT- DATA ITEM TO BE APPENDED

RIGHT ARGUMENT- INTEGER SCALAR OR 1 OR 2 ELEMENT INTEGER VECTOR- FILE TIE NUMBER, AND PASSNUMBER IF NECESSARY

NOTE- IF FILE RESERVATION IS EXCEEDED, PROGRAM WILL ERROR OUT, SINCE THIS ERROR REQUIRES USER INTERVENTION TO RESOLVE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/FILES/FAPPENDR.1

TYPE: FUNCTION

SUMMARY: COVER FUNCTION FOR ⎕APPENDR- RESIZES FILE IF NECESSARY

TIMESTAMP: 1983-03-01 04:19:57

CATEGORIES: 9 FILES
43 FILE PRIMITIVE SIMULATION

DESCRIPTION:
POINTER←DATA FAPPENDR TIE

COVER FUNCTION FOR ⎕APPENDR- RESIZES FILE IF NECESSARY

LEFT ARGUMENT- DATA ITEM TO BE APPENDED

RIGHT ARGUMENT- INTEGER SCALAR OR 1 OR 2 ELEMENT INTEGER VECTOR- FILE TIE NUMBER, AND PASSNUMBER IF NECESSARY

NOTE- IF FILE RESERVATION IS EXCEEDED, PROGRAM WILL ERROR OUT, SINCE THIS ERROR REQUIRES USER INTERVENTION TO RESOLVE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/FILES/FCOMPARE.2

TYPE: FUNCTION

SUMMARY: COMPARES 2 FILES THAT HAVE THE SAME NUMBER OF COMPONENTS

TIMESTAMP: 1984-01-31 23:34:05

CATEGORIES: 9 FILES
13 FILE TOOLS

DESCRIPTION:
LFILE FCOMPARE RFILE

COMPARES 2 FILES WHICH HAVE THE SAME NUMBER OF COMPONENTS, SOME OF WHICH MAY HAVE BEEN ⎕REPLACED WITH DIFFERENT VALUES

RIGHT ARGUMENT- FILE TIE NUMBER OF ONE FILE

LEFT ARGUMENT- FILE TIE NUMBER OF THE OTHER FILE

THE RESULT IS DISPLAYED, NOT RETURNED EXPLICITLY. IT DETAILS ANY DIFFERENCES BETWEEN CORRESPONDING COMPONENTS, STATING IF ONE IS A PACKAGE AND THE OTHER IS NOT, OR IF THEY ARE BOTH ARRAYS DIFFERING IN TYPE, SHAPE, OR DATA ELEMENTS, OR IF THEY ARE DIFFERING PACKAGES.

SEE ALSO RCM/FILES/FMATCH, WHICH DOES A MORE GENERAL COMPARISON BETWEEN TWO FILES--ESSENTIALLY LEFTFILE∘.≠RIGHTFILE.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

THIS FUNCTION, AS ORIGINALLY SUBMITTED, DID NOT WORK. IT HAS BEEN REVISED BY CHRIS D. BURKE AND DOUGLAS J. KEENAN.

TITLE: RCM/FILES/FCOMPRESS.1 RCM/FILES/FCOMPRESS.1

TYPE: FUNCTION

SUMMARY: ANALOG TO PRIMITIVE / FOR FILES

TIMESTAMP: 1983-03-01 04:18:49

CATEGORIES: 9 FILES

DESCRIPTION:
OUTFILE←CONTROL FCOMPRESS INFILE

ANALOG TO PRIMITIVE / FOR FILES

RIGHT ARGUMENT- INPUT FILE TIE NUMBER

LEFT ARGUMENT- BOOLEAN VECTOR SPECIFYING COMPONENTS TO KEEP OR DELETE

RESULT- OUTPUT FILE TIE NUMBER, OF THE NEWLY CREATED FILE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/FILES/FCOPY.1 RCM/FILES/FCOPY.1

TYPE: FUNCTION

SUMMARY: COPIES PART OR ALL OF FILE AS SPECIFIED

TIMESTAMP: 1983-03-01 04:17:38

CATEGORIES: 9 FILES
13 FILE TOOLS

DESCRIPTION:
OUTFILE←OUTFILE FCOPY INFILE

COPIES PART OR ALL OF FILE AS SPECIFIED

RIGHT ARGUMENT- INTEGER VECTOR OF 1-5 ELEMENTS:

[1] INPUT FILE TIE NUMBER, [2] FIRST COMPONENT TO BE COPIED,
[3] LAST COMPONENT TO BE COPIED, [4] COMPONENT INCREMENT,
[5] INPUT FILE PASSNUMBER

LEFT ARGUMENT- TIE NUMBER OR NAME OF OUTPUT FILE

RESULT- TIE NUMBER OF OUTPUT FILE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/FILES/FDROP.1*

TYPE: *FUNCTION*

SUMMARY: *ANALOG TO PRIMITIVE ↓ FOR FILES*

TIMESTAMP: 1983-03-01 04:15:45

CATEGORIES: 9 *FILES*
13 *FILE TOOLS*

DESCRIPTION:
OUTFILE←CONTROL FDROP INFILE

ANALOG TO PRIMITIVE ↓ FOR FILES

RIGHT ARGUMENT-INPUT FILE TIE NUMBER

LEFT ARGUMENT- INTEGER SPECIFYING NUMBER OF COMPONENTS TO DROP

RESULT- OUTPUT FILE TIE NUMBER, OF THE NEWLY CREATED FILE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: *RCM/FILES/FERASE.1*

TYPE: *FUNCTION*

SUMMARY: *ERASES A FILE SPECIFIED BY ARGUMENT*

TIMESTAMP: 1983-03-01 04:17:10

CATEGORIES: 9 *FILES*

DESCRIPTION:
FERASE FILE

ERASES A FILE SPECIFIED BY ARGUMENT

ARGUMENT- INTEGER SCALAR TIE NUMBER OR CHAR VECTOR FILE NAME

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/FILES/FEXPAND.1*

TYPE: *FUNCTION*

SUMMARY: *ANALOG TO PRIMITIVE \ FOR FILES*

TIMESTAMP: 1983-03-01 04:20:25

CATEGORIES: 9 *FILES*

DESCRIPTION:
OUTFILE←CONTROL FEXPAND INFILE

ANALOG TO PRIMITIVE \ FOR FILES

RIGHT ARGUMENT- INPUT FILE TIE NUMBER

LEFT ARGUMENT- BOOLEAN VECTOR SPECIFYING WHERE TO ADD OR COPY COMPONENTS

RESULT- OUTPUT FILE TIE NUMBER, OF THE NEWLY CREATED FILE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: *RCM/FILES/FEXTENT.3*

TYPE: *FUNCTION*

SUMMARY: *RETURNS TYPE, SPACE, RANK, AND SHAPE OF SPECIFIED COMPONENTS.*

TIMESTAMP: 1984-01-25 21:15:35

CATEGORIES: 9 *FILES*
13 *FILE TOOLS*

DESCRIPTION:
EXTENT←LMT FEXTENT TIE

RETURNS TYPE, SPACE, RANK, AND SHAPE OF SPECIFIED COMPONENTS.

RIGHT ARGUMENT- INTEGER SCALAR OR 1 OR 2 ELEMENT INTEGER VECTOR. FILE TIE NUMBER, AND PASSNUMBER IF NECESSARY

LEFT ARGUMENT - 2 ELEMENT INTEGER VECTOR. LOWEST AND HIGHEST COMPONENT NUMBERS FOR WHICH EXTENT IS TO BE OBTAINED. DEFAULT IS ALL COMPONENTS.

RESULT- TYPE(0=PACKAGE,1=BOOLEAN,2=CHAR,3=INTEGER,4=FLOATING POINT,5=COMPLEX,6=ENCLOSED), BYTES USED,RANK,SHAPE(PADDED WITH 0'S IF NECESSARY)

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/FILES/FILES∆REPORT.1

TYPE: FUNCTION

SUMMARY: PRODUCES A REPORT ON THE FILES BELONGING TO ONE OR MORE ACCOUNTS

TIMESTAMP: 1983-03-01 04:18:35

CATEGORIES: 9 FILES
13 FILE TOOLS

DESCRIPTION:
TABLE←FILES∆REPORT LIB

PRODUCES A REPORT ON THE FILES BELONGING TO ONE OR MORE ACCOUNTS

ARGUMENT- INTEGER SCALAR OR VECTOR OF ACCOUNTS AND LIBRARY NUMBERS
RESULT - CHAR MATRIX TABLE CONTAINING FILE NAME, SIZE AND ACCESS MATRIX FOR EACH FILE

AS ORIGINALLY SUBMITTED, THIS FUNCTION DID NOT WORK. IT WAS CORRECTED BY MICHEL BOUCHARD, IPSA/OTTAWA (MRAB). IT WILL, HOWEVER, STILL SUSPEND ON A 'FILE TIED' ERROR IF ANY FILE THAT IS TO APPEAR IN THE REPORT IS ALREADY TIED WHEN THE FN IS CALLED. ALSO, IT ASSUMES YOU HAVE ⎕RDAC ACCESS TO ALL THE FILES IN <LIB>.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/FILES/FINDEX.1

TYPE: FUNCTION

SUMMARY: ANALOG TO PRIMITIVE [] FOR FILES

TIMESTAMP: 1983-03-01 04:17:20

CATEGORIES: 9 FILES

DESCRIPTION:
OUTFILE←CONTROL FINDEX INFILE

ANALOG TO PRIMITIVE [] FOR FILES

RIGHT ARGUMENT- INPUT FILE TIE NUMBER

LEFT ARGUMENT- INTEGER VECTOR SPECIFYING COMPONENTS TO SELECT

RESULT- OUTPUT FILE TIE NUMBER, OF THE NEWLY CREATED FILE

THIS FUNCTION WORKS ONLY IN ORIGIN 1.

TITLE: RCM/FILES/FINSERT.1 RCM/FILES/FINSERT.1

TYPE: FUNCTION

SUMMARY: INSERTS DATA ITEM INTO COPY OF INPUT FILE AT SPECIFIED LOCATION

TIMESTAMP: 1983-03-01 04:19:29

CATEGORIES: 9 FILES

DESCRIPTION:
OUTFILE←DATA FINSERT TIE

INSERTS DATA ITEM INTO COPY OF INPUT FILE AT SPECIFIED LOCATION

RIGHT ARGUMENT- 2 OR 3 ELEMENT NUMERIC VECTOR. TIE NUMBER, COMPONENT NUMBER AFTER WHICH DATA IS TO BE INSERTED, AND PASSNUMBER IF NECESSARY

LEFT ARGUMENT- DATA VALUE TO BE INSERTED IN FILE

RESULT- FILE TIE NUMBER OF COPY OF THE NEWLY CREATED FILE FILE WITH DATA INSERTED

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/FILES/FLIB.1 RCM/FILES/FLIB.1

TYPE: FUNCTION

SUMMARY: COVER FUNCTION FOR ⎕LIB- RETURNS ALPHABETIZED FILE LIBRARY LIST

TIMESTAMP: 1983-03-01 04:16:42

CATEGORIES: 9 FILES

DESCRIPTION:
LIB←FLIB ACCT

COVER FUNCTION FOR ⎕LIB- RETURNS ALPHABETIZED FILE LIBRARY LIST

ARGUMENT-INTEGER SCALAR OR 1 ELEMENT VECTOR ACCOUNT NUMBER

RESULT- N×22 CHAR MATRIX OF FILE NAMES

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/FILES/FMATCH.1

TYPE: FUNCTION

SUMMARY: MATCHES COMPS OF 2 FILES WHERE ONE FILE HAS HAD COMPS INSERTED OR THE OTHER HAS HAD COMPS DELETED

TIMESTAMP: 1983-03-01 04:19:43

CATEGORIES: 9 FILES
13 FILE TOOLS

DESCRIPTION:
LFILE FMATCH RFILE

MATCHES COMPONENTS OF 2 FILES WHERE THE LEFT FILE HAS HAD COMPONENTS INSERTED OR THE RIGHT FILE HAS HAD COMPONENTS DELETED

RIGHT ARGUMENT- TIE NUMBER OF FILE WITH FEWER COMPONENTS

LEFT ARGUMENT- TIE NUMBER OF FILE WITH MORE COMPONENTS

THE RESULT IS DISPLAYED, NOT REURNED EXPLICITLY. IT IS SIMILAR TO LEFTFILE ∘.= RIGHTFILE.

IF ONE FILE IS JUST LIKE THE OTHER FILE, EXCEPT THAT SOME COMPONENTS HAVE BEEN REPLACED, RCM/FILES/FCOMPARE IS MORE EFFICIENT AND GIVES MORE INFORMATION.

THIS FUNCTION IS □IO-INDEPENDENT.

AS ORIGINALLY SUBMITTED, THIS FUNCTION DID NOT WORK IF EITHER FILE CONTAINED A PACKAGE. IT WAS REVISED BY DOUGLAS J. KEENAN.

TITLE: RCM/FILES/FMERGE.1

TYPE: FUNCTION

SUMMARY: MERGES SEVERAL FILES INTO A SINGLE FILE

TIMESTAMP: 1983-03-01 04:16:03

CATEGORIES: 9 FILES
12 FILE ORGANIZATION
13 FILE TOOLS

DESCRIPTION:
OUTFILE←FMERGE CONTROL

MERGES SEVERAL FILES INTO A SINGLE FILE

ARGUMENT- INTEGER VECTOR SPECIFIYING WHICH FILE THE NEXT COMPONENT SHOULD BE READ FROM ALL ELEMENTS MUST BE VALID TIE NUMBERS

THE NUMBER OF ELEMENTS SHOULD EQUAL THE TOTAL NUMBER OF COMPONENTS IN ALL THE FILES

WARNING: THIS FUNCTION USES A DIRECTORY TO MINIMIZE CPU TIME USED
IF YOUR TIE NUMBERS ARE LARGE (>10000) THIS PROGRAM MAY WS FULL

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/FILES/FOPEN.2

TYPE: FUNCTION

SUMMARY: SHARE TIES A FILE, AND CREATES IT IF NECESSARY

TIMESTAMP: 1984-03-22 02:41:53

CATEGORIES: 9 FILES
43 FILE PRIMITIVE SIMULATION

DESCRIPTION:
TIE←SIZE FOPEN NAME

SHARE TIES A FILE, AND CREATES IT IF NECESSARY

RIGHT ARGUMENT- CHAR VECTOR NAME OF FILE

LEFT ARGUMENT- OPTIONAL: BYTES TO RESERVE FOR FILE. IF SPECIFIED, FILE WILL BE CLEARED OF ALL EXISTING COMPONENTS

RESULT- INTEGER SCALAR SPECIFYING NUMBER FILE WAS TIED TO

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/FILES/FPRINT.1*

TYPE: *FUNCTION*

SUMMARY: *PRINTS CHARACTERISTICS AND/OR VALUES OF COMPONENTS IN A FILE*

TIMESTAMP: 1983-03-01 04:20:34

CATEGORIES: 9 *FILES*

DESCRIPTION:
CONTROL FPRINT FILE

PRINTS CHARACTERISTICS AND VALUES OF COMPONENTS IN A FILE

RIGHT ARGUMENT- INTEGER SCALAR FILE TIE NUMBER TO BE PROCESSED

LEFT ARGUMENT- A FOUR-ELEMENT BOOLEAN VECTOR

[1] *CONTROLS PRINTING OF VALUE.* [2] *CONTROLS PRINTING OF COMPONENT NUMBER*
[3] *CONTROL PRINTING DATA TYPE.* [4] *CONTROLS PRINTING SHAPE*

AS ORIGINALLY SUBMITTED, THIS FUNCTION DID NOT WORK. IT WAS COMPLETED AND CORRECTED BY DOUGLAS J. KEENAN.

THIS FUNCTION IS □*IO-INDEPENDENT.*

TITLE: RCM/FILES/FREPLACE.1

TYPE: FUNCTION

SUMMARY: COVER FUNCTION FOR ⎕REPLACE- RESIZES FILE IF NECESSARY

TIMESTAMP: 1983-03-01 04:19:15

CATEGORIES: 9 FILES
43 FILE PRIMITIVE SIMULATION

DESCRIPTION:
DATA FREPLACE TIE

COVER FUNCTION FOR ⎕REPLACE- RESIZES FILE IF NECESSARY

RIGHT ARGUMENT- 2 OR 3 ELEMENT INTEGER VECTOR- FILE TIE NUMBER, COMPONENT NUMBER, AND IF NECESSARY, PASSNUMBER

LEFT ARGUMENT- DATA ITEM TO BE WRITE OVER EXISTING COMPONENT

NOTE- IF FILE RESERVATION IS EXCEEDED, PROGRAM WILL ERROR OUT, SINCE THIS ERROR REQUIRES USER INTERVENTION TO RESOLVE

AS ORIGINALLY SUBMITTED, THIS FUNCTION DID NO WORK. IT WAS CORRECTED BY DOUGLAS J. KEENAN.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/FILES/FRESIZE.1

TYPE: FUNCTION

SUMMARY: COVER FUNCTION FOR ⎕RESIZE- IF ASKING TOO MUCH, GETS AS MUCH AS FILE RES ALLOWS

TIMESTAMP: 1983-03-01 04:17:01

CATEGORIES: 9 FILES
43 FILE PRIMITIVE SIMULATION

DESCRIPTION:
SIZE←SIZE FRESIZE TIE

COVER FUNCTION FOR ⎕RESIZE- IF ASKING TOO MUCH, GETS AS MUCH AS FILE RES ALLOWS

RIGHT ARGUMENT- INTEGER SCALAR OR 1 OR 2 ELEMENT INTEGER VECTOR. FILE TIE NUMBER, AND PASSNUMBER IF NECESSARY

LEFT ARGUMENT- INTEGER SCALAR SPECIFYING NEW FILE SIZE

RESULT- SIZE FILE WAS ACTUALLY RESIZED TO

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/FILES/FREVERSE.1

TYPE: FUNCTION

SUMMARY: ANALOG TO PRIMITIVE ⌽ FOR FILES

TIMESTAMP: 1983-03-01 04:17:47

CATEGORIES: 9 FILES

DESCRIPTION:
OUTFILE←FREVERSE INFILE

ANALOG TO PRIMITIVE ⌽ FOR FILES

ARGUMENT-INPUT FILE TIE NUMBER

RESULT- OUTPUT FILE TIE NUMBER, OF THE NEWLY CREATED FILE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/FILES/FROTATE.1

TYPE: FUNCTION

SUMMARY: ANALOG TO PRIMITIVE ⌽ FOR FILES

TIMESTAMP: 1983-03-01 04:20:07

CATEGORIES: 9 FILES

DESCRIPTION:
OUTFILE←CONTROL FROTATE INFILE

ANALOG TO PRIMITIVE ⌽ FOR FILES

RIGHT ARGUMENT- INPUT FILE TIE NUMBER

LEFT ARGUMENT- INTEGER SPECIFYING NUMBER OF COMPONENTS TO ROTATE

RESULT- OUTPUT FILE TIE NUMBER, OF THE NEWLY CREATED FILE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/FILES/FSPLIT.1

TYPE: FUNCTION

SUMMARY: SPLITS THE COMPONENTS OF A FILE INTO SEVERAL FILES

TIMESTAMP: 1983-03-01 04:16:52

CATEGORIES: 9 FILES

DESCRIPTION:
CONTROL FSPLIT INFILE

SPLITS THE COMPONENTS OF A FILE INTO SEVERAL FILES

RIGHT ARGUMENT- INPUT FILE TIE NUMBER

LEFT ARGUMENT- INTEGER VECTOR DESIGNATING WHICH FILE COMPONENT SHOULD BE APPENDED TO. ALL ELEMENTS MUST BE VALID TIE NUMBERS. MUST BE 1 ELEMENT PER COMPONENT

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

RCM/FILES/FTAKE.1

TITLE: RCM/FILES/FTAKE.1

TYPE: FUNCTION

SUMMARY: ANALOG TO PRIMITIVE ↑ FOR FILES

TIMESTAMP: 1983-03-01 04:20:54

CATEGORIES: 9 FILES

DESCRIPTION:
OUTFILE←CONTROL FTAKE INFILE

ANALOG TO PRIMITIVE ↑ FOR FILES

RIGHT ARGUMENT- INPUT FILE TIE NUMBER
LEFT ARGUMENT- INTEGER SPECIFYING NUMBER OF COMPONENTS TO TAKE

RESULT- OUTPUT FILE TIE NUMBER, OF THE NEWLY CREATED FILE

THE LEFT ARGUMENT MAY BE EITHER POSITIVE OR NEGATIVE. OVERTAKING IS NOT DONE--THE FUNCTION WILL HALT ON A FILE INDEX ERROR IF |CONTROL IS TOO LARGE.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/FILES/FTS.1

TYPE: FUNCTION

SUMMARY: RETURNS TIME STAMPS FOR DATES WHEN SPECIFIED COMPONENTS WERE WRITTEN

TIMESTAMP: 1983-03-01 04:23:08

CATEGORIES: 9 FILES

DESCRIPTION:
STAMP←LMT FTS TIE

RETURNS TIME STAMPS FOR DATES WHEN SPECIFIED COMPONENTS WERE WRITTEN

RIGHT ARGUMENT- 1 OR 2 ELEMENT INTEGER VECTOR. FILE TIE NUMBER, AND PASSNUMBER IF NECESSARY

LEFT ARGUMENT- 2 ELEMENT INTEGER VECTOR. LOWEST AND HIGHEST COMPONENT NUMBERS FOR WHICH TIMESTAMPS ARE TO BE OBTAINED. DEFAULT IS ALL COMPONENTS.

RESULT- N×7 INTEGER MATRIX OF TIMESTAMPTS WHEN SPECIFIED COMPONENTS WERE WRITTEN: YEAR,MONTH,DAY,HOUR,MINUTE,SECOND,SECOND÷60

THIS FUNCTION IS □IO-INDEPENDENT.

REVIEWS:
[1983-09-01 01:33:44]

THE FUNCTION APPEARS TO BE INCORRECT. IT JUST TOLD ME THAT A COMPONENT I □REPLACE'D IN AUGUST WAS □REPLACE'D IN MAY. PCB/RDCIDECODE GAVE THE CORRECT RESULT. --DOUGLAS J. KEENAN

TITLE: RCM/FILES/FWRITE.1

TYPE: *FUNCTION*

SUMMARY: *WRITES TO SPECIFIED COMPONENT. IF COMP DOESN'T EXIST, CREATES IT AND ANY NECESSARY PRECEDING COMPS.*

TIMESTAMP: 1983-03-01 04:15:53

CATEGORIES: 9 *FILES*
43 *FILE PRIMITIVE SIMULATION*

DESCRIPTION:
DATA FWRITE TIE

WRITES DATA TO SPECIFIED COMPONENT. IF COMPONENT DOESN'T EXIST, IT CREATES IT AND ANY NECESSARILY PRECEDING COMPONENTS

RIGHT ARGUMENT- 2 OR 3 ELEMENT INTEGER VECTOR. TIE NUMBER, COMPONENT NUMBER, AND OPTIONAL PASSNUMBER

LEFT ARGUMENT- ANY APL DATA VALUE-- TO BE WRITTEN TO FILE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/SHAREDVARS/MONITOR.1

TYPE: *FUNCTION*

SUMMARY: *NTASK MONITOR WHICH PROCESSES REQUESTS FROM OTHER TASKS*

TIMESTAMP: 1983-03-01 06:58:44

CATEGORIES: 14 *COMMUNICATION BETWEEN TASKS*
18 *N-TASKS*

DESCRIPTION:
MONITOR CLONEID

NTASK MONITOR WHICH PROCESSES REQUESTS FROM OTHER TASKS

ARGUMENT- INTEGER SCALAR CLONE ID

IT IS ASSUMED THEIR IS A MONADIC FUNCTION <PROCESS> THAT TAKES NEW VALUES OF A SHARED VARIABLE, PROCESSES IT, AND RETURNS AN EXPLICIT RESULT TO BE ASSIGNED TO THE SHARED VARIABLE. THE FUNCTION RCM/SHAREDVARS/NEWOFFERS IS USED TO ACCEPT ALL OFFERS MADE TO THIS NTASK; THAT IS, THE MONITOR WILL ATTEMPT TO PROCESS ANY VARIALBES THAT ARE OFFERED SPECIFICALLY TO IT.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/SHAREDVARS/NEWOFFERS.1

TYPE: FUNCTION

SUMMARY: DETECTS OFFERS BY NEW PROCESSORS, AND SETS UP SHARES WITH THEM

TIMESTAMP: 1983-03-01 06:57:52

CATEGORIES: 14 COMMUNICATION BETWEEN TASKS
15 SHARED VARIABLES

DESCRIPTION:
FLAG←NEWOFFERS

DETECTS OFFERS BY NEW PROCESSORS, AND SETS UP SHARES WITH THEM

COMMON VARS ACCESSED- SHAREDVARS,PROCIDS

THE TWO VARIABLES ARE MATRICES, THE FORMER CHAR AND THE LATTER NUMERIC. IF THIS FUNCTION IS CALLED BY THE FUNCTION RCM/SHAREDVARS/MONITOR, THEY WILL BE INITIALIZED TO 0 0⍴'' AND 0 2 ⍴0. THIS FUNCTION MAY CATENATE ROWS ON THE BOTTOM OF THEM. THEIR PURPOSE IS JUST WHAT YOU THINK.

THIS FUNCTION MAY BE PROFITABLY USED WITH RCM/SHAREDVARS/OLDOFFERS.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/SHAREDVARS/PORT.1

TYPE: FUNCTION

SUMMARY: ALLOWS)PORT TO BE DONE UNDER PROGRAM CONTROL

TIMESTAMP: 1983-03-01 06:48:36

CATEGORIES: 14 COMMUNICATION BETWEEN TASKS

DESCRIPTION:
REPORT←TEXT PORT SIGNON

ALLOWS)PORT TO BE DONE UNDER PROGRAM CONTROL

RIGHT ARGUMENT- CHAR VECTOR WITH YOUR ACCOUNT,COLON,LOCK. EX. '1234567:SECRET'

LEFT ARGUMENT- CHAR VECTOR PARAMETER TO)PORT. EX. '1234567'

RESULT- CHAR VECTOR OUTPUT FROM)PORT COMMAND

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/SHAREDVARS/SETUP∆INTERFACE.1

TYPE: FUNCTION

SUMMARY: SETS UP SPECIFIED INTERFACE BETWEEN TWO PROCESSORS RUNNING

TIMESTAMP: 1983-03-01 06:53:17

CATEGORIES: 14 COMMUNICATION BETWEEN TASKS
15 SHARED VARIABLES

DESCRIPTION:
NAME SETUP∆INTERFACE ID

SETS UP SPECIFIED INTERFACE BETWEEN TWO PROCESSORS RUNNING

ON THE SAME ACCT. PROGRAM STOPS WHEN COUPLING IS COMPLETE

RIGHT ARGUMENT- [1]=INTERFACE TYPE: 4→FULL DUPLEX; 3,2→HALF DUPLEX; 1→SIMPLEX
[2]= 0 OR 1, SPECIFYING CLONE ID.

FOR SIMPLEX, CLONE ID OF RECEIVING PROCESSOR SHOULD BE 0, SENDING PROCESSOR 1

LEFT ARGUMENT- CHAR VECTOR NAME OF INTERFACE VARIABLE.
LEFT ARGUMENT MAY BE OMITTED; DEFAULT VALUE IS 'INTERFACE'.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/SHAREDVARS/STASK∆NUMS.1

TYPE: FUNCTION

SUMMARY: CONVERTS THE CHARACTER OUTPUT OF AN STASK TO A NUMERIC ARRAY.

TIMESTAMP: 1983-03-01 07:00:54

CATEGORIES: 17 S-TASKS

DESCRIPTION:
R←STASK∆NUMS V

THIS FUNCTION TAKES THE RESULT OF RCM/SHAREDVARS/STASK∆OUTPUT AND RETURNS A

NUMERIC ARRAY SUCH THAT THE FIRST ELEMENT OF THE SHAPE ≠ 1.

USES THE FUNCTION, <PPLUSRA>

THIS FUNCTION IS VERY USEFUL WHEN IT IS KNOWN THAT THE OUTPUT FROM AN STASK IS
A NUMERIC ARRAY.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/SHAREDVARS/STASK∆OUTPUT.1

TYPE: FUNCTION

SUMMARY: RETURNS AN ARRAY OF OUTPUT FROM STASK

TIMESTAMP: 1983-03-01 06:49:03

CATEGORIES: 17 S-TASKS

DESCRIPTION:
OUTPUT←STASK∆OUTPUT VAR

RETURNS AN ARRAY OF OUTPUT FROM STASK

ARGUMENT- CHAR VECTOR NAME OF CONTROL VARIABLE
VARIABLE MUST ALREADY HAVE BEEN SHARED WITH PROCESSOR 1, AND HAVE ACCESS SET TO 1 1 1 1

RESULT- CHAR ARRAY WITH EMBEDDED CARRIAGE RETURNS WHICH IS A VISUAL REPRESENTATION OF THE RESULT OF AN APL STATEMENT OR SYSTEM COMMAND WHICH WAS PREVIOUSLY ASSIGNED TO THE CONTROL VAR

IF 1=ρρRESULT, NEXT INPUT IS IMMEX
IF 2=ρρRESULT, NEXT INPUT IS ⍞
IF 3=ρρRESULT, NEXT INPUT IS ⎕ARBIN

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/SHAREDVARS/TRANSMIT1.1

TYPE: FUNCTION

SUMMARY: TRANSMITS DATA TO PARTNER USING SIMPLEX MECHANISM

TIMESTAMP: 1983-03-01 06:53:50

CATEGORIES: 14 COMMUNICATION BETWEEN TASKS
15 SHARED VARIABLES

DESCRIPTION:
DATA TRANSMIT1 PARTNER

TRANSMITS DATA TO PARTNER USING SIMPLEX MECHANISM

RIGHT ARGUMENT- INTEGER SCALAR ACCOUNT NUMBER OF PARTNER, OR 2 ELEMENT VECTOR OF ACCOUNT AND CLONE ID

LEFT ARGUMENT- DATA TO BE TRANSMITTTED TO PARTNER

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/SHAREDVARS/TRANSMIT2.1*

TYPE: *FUNCTION*

SUMMARY: *TRANSMITS DATA TO PARTNER USING HALF-DUPLEX MECHANISM*

TIMESTAMP: 1983-03-01 06:54:05

CATEGORIES: 14 *COMMUNICATION BETWEEN TASKS*
15 *SHARED VARIABLES*

DESCRIPTION:
DATA TRANSMIT2 PARTNER

TRANSMITS DATA TO PARTNER USING HALF-DUPLEX MECHANISM

RIGHT ARGUMENT- INTEGER SCALAR ACCOUNT OF SHARING PARTNER, OR 2 ELEMENT VECTOR OF ACCOUNT AND CLONE ID

LEFT ARGUMENT- DATA TO BE TRANSMITTED TO PARTNER

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: *RCM/TERMFNS/AJ510/CHARSET.1*

TYPE: *FUNCTION*

SUMMARY: *CHANGES AJ510 CHARACTER SET. ARGUMENT-* 1→*ASCII*, 2→*APL*, 3→*GRAPHICS*

TIMESTAMP: 1983-02-28 08:20:58

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
CHARSET TYPE

AJ510
ARGUMENT- 1→*ASCII*, 2→*APL*, 3→*GRAPHICS*

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ510/FEATURE.1* *RCM/TERMFNS/AJ510/FEATURE.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ510. ARGUMENT- 0→END FEATURE, 1→UNDERLINE, 2→BLINK, 3→INVERSE VIDEO, 4→LOW INTENSITY*

TIMESTAMP: 1983-02-28 08:20:33

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
FEATURE TYPE

*AJ*510
ARGUMENT- 0→*END FEATURE*, 1→*UNDERLINE*, 2→*BLINK*, 3→*INVERSE VIDEO*, 4→*LOW INTENSITY*

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: *RCM/TERMFNS/AJ510/MOVE.1* *RCM/TERMFNS/AJ510/MOVE.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ510. MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)*

TIMESTAMP: 1983-02-28 08:20:50

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*
23 *TABS*

DESCRIPTION:
DIRECTION MOVE DISTANCE

*AJ*510
MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)
RIGHT ARGUMENT- INTEGER VECTOR OF 1 OR MORE ELEMENTS. SPECIFIES DISTANCE TO BE MOVED
LEFT ARGUMENT - CHAR VECTOR OF 1 OR MORE ELEMENTS. SPECIFIES DIRECTION OF MOVE. 'L'→LEFT,'R'→RIGHT,'U'→UP,'D'→DOWN

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ510/RESET∆TERM.1

TYPE: FUNCTION

SUMMARY: FOR AJ510. RESETS TERMINAL; SAME AS ESC) ρ

TIMESTAMP: 1983-02-28 08:21:06

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
RESET∆TERM

AJ510
RESETS THE TERMINAL; SAME AS ESC) ρ
THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/AJ510/SET∆HTABS.1

TYPE: FUNCTION

SUMMARY: FOR AJ510. SETS HORIZONTAL TAB STOPS

TIMESTAMP: 1983-02-28 08:20:40

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
SET∆HTABS N

AJ510
SETS HORIZONTAL TAB STOPS
ARGUMENT- INTEGER SCALAR CAUSES TABS TO BE SET AT INTERVALS OF N SPACES FROM LEFT MARGIN
INTEGER VECTOR CAUSES TABS TO BE SET AT ABSOLUTE POSITIONS SPECIFIED (FROM LEFT MARGIN)

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ510/STATUS.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ510.* *ARGUMENT- 1 OR 0, SPECIFYING STATUS LINE ON OR OFF*

TIMESTAMP: 1983-02-28 08:21:17

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
STATUS SWITCH

AJ510
ARGUMENT- 1 OR 0, SPECIFYING STATUS LINE ON OR OFF

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ510/UCALPHA.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ510.* *ARGUMENT- 0 OR 1, MEANING TURN UPPER CASE ALPHA OFF OR ON*

TIMESTAMP: 1983-02-28 08:21:25

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
UCALPHA SWITCH

AJ510
ARGUMENT- 0 OR 1, MEANING TURN UPPER CASE ALPHA OFF OR ON

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ832/AUTO∆LINEFEED.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ832. SETS VERTICAL PITCH TO 3 LINES/INCH*

TIMESTAMP: 1983-02-28 08:22:18

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
AUTO∆LINEFEED

AJ832
SETS VERTICAL PITCH TO 3 LINES/INCH

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: *RCM/TERMFNS/AJ832/BOLD∆FACE.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ832. TURNS ON OR OFF PRINT ENHANCEMENT MODE*

TIMESTAMP: 1983-02-28 08:22:26

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
BOLD∆FACE CONTROL

AJ832
TURNS ON OR OFF PRINT ENHANCEMENT MODE
ARGUMENT- BOOLEAN OR INTEGER SCALAR OR 1 ELEMENT VECTOR. 1→ *BOLDVACE ON,*
0→*BOLDFACE OFF*

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ832/CLEAR∆HTABS.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. CLEARS ALL HORIZONTAL TAB SETTINGS

TIMESTAMP: 1983-02-28 08:22:35

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
CLEAR∆HTABS

AJ832
CLEARS ALL HORIZONTAL TAB SETTINGS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/AJ832/CLEAR∆VTABS.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. CLEARS ALL VERTICAL TAB SETTINGS

TIMESTAMP: 1983-02-28 08:22:43

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
CLEAR∆VTABS

AJ832
CLEARS ALL VERTICAL TAB SETTINGS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ832/CONTROL∆ENABLE.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ832. ENABLES OR DISABLES CONTROL CODE INTERPRETATION*

TIMESTAMP: 1983-02-28 08:22:56

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
CONTROL∆ENABLE CONTROL

AJ832
ENABLES OR DISABLES CONTROL CODE INTERPRETATION
ARGUMENT- BOOLEAN OR INTEGER SCALAR OR 1 ELEMENT VECTOR. 1→CONTROL CODES INTERPRETED, 0→CONTROL CODES PRINTED

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: *RCM/TERMFNS/AJ832/GOTO.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ832. ABSOLUTE TAB TO SPECIFIED LOCATION*

TIMESTAMP: 1983-02-28 08:23:08

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*
23 *TABS*

DESCRIPTION:
AXIS GOTO POSITION

AJ832
ABSOLUTE TAB TO SPECIFIED LOCATION
RIGHT ARGUMENT- INTEGER VECTOR OF 1 OR 2 ELEMENTS. SPECIFIES LOCATION ON AXIS DESIGNATED BY LEFT ARGUMENT
LEFT ARGUMENT- CHAR VECTOR OF 1 OR 2 ELEMENTS. 'X' OR 'Y' OR BOTH. SPECIFIES AXIS OF MOVE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ832/LINE∆FEED.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. CAUSES VERTICAL MOVEMENT AS SPECIFIED

TIMESTAMP: 1983-02-28 08:23:21

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
LINE∆FEED N

AJ832
CAUSES VERTICAL MOVEMENT AS SPECIFIED
ARGUMENT- NUMERIC SCALAR SPECIFYING NUMBER OF FULL AND HALF LINEFEEDS
SIGN SPECIFIES DIRECTION(-UP),(+DOWN). HALF LF'S SPECIFIED BY USING
DECIMAL FRACTION, IE. LINEFEED 1.5

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/AJ832/MOVE.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)

TIMESTAMP: 1983-02-28 08:22:09

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
DIRECTION MOVE DISTANCE

AJ832
MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)
RIGHT ARGUMENT- INTEGER VECTOR OF 1 OR MORE ELEMENTS. SPECIFIES DISTANCE
(1/60 HORIZONTAL, 1/48 VERTICAL) TO BE MOVED
LEFT ARGUMENT - CHAR VECTOR OF 1 OR MORE ELEMENTS. SPECIFIES DIRECTION OF
MOVE. 'L'→LEFT,'R'→RIGHT,'U'→UP,'D'→DOWN

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ832/PLOT∆MODE.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. TURNS ON OR OFF STANDARD PLOT MODE

TIMESTAMP: 1983-02-28 08:23:33

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
PLOT∆MODE CONTROL

AJ832
TURNS ON OR OFF STANDARD PLOT MODE
ARGUMENT- BOOLEAN OR INTEGER SCALAR OR 1 ELEMENT VECTOR. 1→PLOT ON, 0→PLOT OFF

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/AJ832/PLOT∆SPACING.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. SETS HORIZONTAL OR VERTICAL SPACING FOR PLOT MODE.

TIMESTAMP: 1983-02-28 08:23:46

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
AXIS PLOT∆SPACING N

AJ832
RIGHT ARGUMENT- INTEGER SCALAR SPECIFYING MOVE DISTANCE IN PLOT MODE. HORIZONTAL 1/60 INCH, VERTICAL 1/48 INCH INCREMENTS
LEFT ARGUMENT- CHAR SCALAR OR VECTOR WHERE FIRST ELEMENT SPECIFIES AXIS. 'H'→HORIZONTAL,'V'→VERTICAL

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ832/PRINT.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ832. PRINTS TEXT. USES ⎕ARBOUT.*

TIMESTAMP: 1983-02-28 08:23:58

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
PRINT TEXT

AJ832
PRINTS THE TEXT AT THE TERMINAL
ARGUMENT- CHAR SCALAR OR VECTOR OF CHARS TO BE DISPLAYED. NO APL OVERSTRIKES

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: *RCM/TERMFNS/AJ832/PRINT∆ENABLE.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ832. ENABLE OR DISABLES TERMINAL PRINTING*

TIMESTAMP: 1983-02-28 08:24:07

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
PRINT∆ENABLE CONTROL

AJ832
ENABLE OR DISABLES TERMINAL PRINTING
ARGUMENT- BOOLEAN OR INTEGER SCALAR OR 1 ELEMENT VECTOR. 1→PRINT ON, 0→PRINT OFF

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ832/RELEASE∆MARGIN.1 RCM/TERMFNS/AJ832/RELEASE∆MARGIN.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. RELEASE MARGINS

TIMESTAMP: 1983-02-28 08:24:16

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
RELEASE∆MARGIN

AJ832
RELEASE MARGINS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/AJ832/RESET∆MARGINS.1 RCM/TERMFNS/AJ832/RESET∆MARGINS.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. LEFT MARGIN RESTORED TO 0, RIGHT MARGIN TO 131

TIMESTAMP: 1983-02-28 08:24:25

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
RESET∆MARGINS

AJ832
LEFT MARGIN RESTORED TO 0, RIGHT MARGIN TO 131

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ832/RESET∆PITCH.1 RCM/TERMFNS/AJ832/RESET∆PITCH.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. RESTORES HORIZONTAL PITCH TO SWITCH SETTING AND VERTICAL PITCH TO 6 LPI

TIMESTAMP: 1983-02-28 08:24:33

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
RESET∆PITCH

AJ832
RESTORES HORIZONTAL PITCH TO SWITCH SETTING AND VERTICAL PITCH TO 6 LPI

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/AJ832/RESET∆TERM.1 RCM/TERMFNS/AJ832/RESET∆TERM.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. RESTORES SETTING AS WHEN POWERED UP

TIMESTAMP: 1983-02-28 08:24:41

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
RESET∆TERM

AJ832
RESTORES SETTING AS WHEN POWERED UP

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ832/SELECT∆PITCH.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. SELECTS 1 OF 4 POSSIBLE PITCH SETTINGS: 1) 10H/6V, 2) 12H/6V, 3) 10H/8V, 4) 12H/8V

TIMESTAMP: 1983-02-28 08:21:52

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
SELECT∆PITCH N

AJ832
SELECTS 1 OF 4 POSSIBLE PITCH SETTINGS: 1) 10H/6V, 2) 12H/6V, 3) 10H/8V, 4)
12H/8V
ARGUMENT- INTEGER SCALAR 1,2,3, OR 4

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/AJ832/SET∆HTABS.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. SETS HORIZONTAL TAB STOPS

TIMESTAMP: 1983-02-28 08:21:35

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
SET∆HTABS N

AJ832
SETS HORIZONTAL TAB STOPS
ARGUMENT- INTEGER SCALAR CAUSES TABS TO BE SET AT INTERVALS OF N SPACES FROM
LEFT MARGIN
INTEGER VECTOR CAUSES TABS TO BE SET AT ABSOLUTE POSITIONS
SPECIFIED (FROM LEFT MARGIN)

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ832/SET∆MARGIN.1 RCM/TERMFNS/AJ832/SET∆MARGIN.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED

TIMESTAMP: 1983-02-28 08:24:54

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
CONTROL SET∆MARGIN POSITION

AJ832
SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RIGHT ARGUMENT- INTEGER SCALAR SPECIFYING POSITION FROM LEFT MARGIN
LEFT ARGUMENT- CHAR SCALAR SPECIFYING MARGIN EDGE. 'R'→RIGHT, 'L'→LEFT

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/AJ832/SET∆PAGINATION.1 RCM/TERMFNS/AJ832/SET∆PAGINATION.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. SETS NUMBER OF LINES TO BE SKIPPED AT BOTTOM OF PAGE

TIMESTAMP: 1983-02-28 08:25:07

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
SET∆PAGINATION N

AJ832
SETS NUMBER OF LINES TO BE SKIPPED AT BOTTOM OF PAGE
ARGUMENT- INTEGER SCALAR OR 1 ELEMENT VECTOR SPECIFYING NUMBER OF LINES TO SKIP

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ832/SET∆PITCH.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. SETS INCREMENT FOR HORIZONTAL OR VERTICAL MOVEMENT

TIMESTAMP: 1983-02-28 08:21:44

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
CONTROL SET∆PITCH N

AJ832
SETS INCREMENT FOR HORIZONTAL OR VERTICAL MOVEMENT
RIGHT ARGUMENT- INTEGER SPECIFYING NUMBER OF INCREMENTS PER INCH, AND SIGN SPECIFIES DIRECTION (CARTESIAN SYSTEM)
LEFT ARGUMENT- CHAR SCALAR OR VECTOR WHERE FIRST ELEMENT SPECIFIES AXIS. 'H'→HORIZONTAL, 'V'→VERTICAL

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/AJ832/SET∆TOF.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. SETS TOP-OF-FORM

TIMESTAMP: 1983-02-28 08:25:21

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
SET∆TOF

AJ832
SETS TOP-OF-FORM

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ832/SET∆TOF∆LENGTH.1* *RCM/TERMFNS/AJ832/SET∆TOF∆LENGTH.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ832. SETS TOP-OF-FORM AND FORM LENGTH*

TIMESTAMP: 1983-02-28 08:25:33

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
SET∆TOF∆LENGTH N

```
  AJ832
  SETS TOP-OF-FORM AND FORM LENGTH
<N> IS THE NEW FORM LENGTH.
THIS FUNCTION IS ⎕IO-INDEPENDENT.
```

- -

TITLE: *RCM/TERMFNS/AJ832/SET∆VTABS.1* *RCM/TERMFNS/AJ832/SET∆VTABS.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ832. SETS VERTICAL TAB STOPS*

TIMESTAMP: 1983-02-28 08:25:45

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*
23 *TABS*

DESCRIPTION:
SET∆VTABS N

```
  AJ832
  SETS VERTICAL TAB STOPS
  ARGUMENT- INTEGER SCALAR CAUSES TABS TO BE SET AT INTERVALS OF N LINES FROM
TOP OF FORM
           INTEGER VECTOR CAUSES TABS TO BE SET AT ABSOLUTE LINES SPECIFIED

THIS FUNCTION IS ⎕IO-INDEPENDENT.
```

TITLE: *RCM/TERMFNS/AJ832/SUBSCRIPT.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ832. PRINTS TEXT AS A SUBSCRIPT*

TIMESTAMP: 1983-02-28 08:25:59

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
SUBSCRIPT TEXT

AJ832
PRINTS TEXT AS A SUBSCRIPT
ARGUMENT- CHAR SCALAR OR VECTOR TO BE PRINTED- NO OVERSTRUCK APL CHARS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ832/SUPERSCRIPT.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ832. PRINTS TEXT AS SUPERSCRIPT*

TIMESTAMP: 1983-02-28 08:26:11

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
SUPERSCRIPT TEXT

AJ832
PRINTS TEXT AS SUPERSCRIPT
ARGUMENT- CHAR SCALAR OR VECTOR TO BE PRINTED- NO OVERSTRUCK APL CHARS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ832/TWELVE∆PITCH.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. SETS HORIZONTAL PITCH TO 12 CHARS/INCH

TIMESTAMP: 1983-02-28 08:22:00

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
TWELVE∆PITCH

AJ832
SETS HORIZONTAL PITCH TO 12 CHARS/INCH

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ832/ULTRAPLOT∆MODE.1

TYPE: FUNCTION

SUMMARY: FOR AJ832. TURNS ULTRAPLOT MODE ON OR OFF.

TIMESTAMP: 1983-02-28 08:26:24

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
CHAR ULTRAPLOT∆MODE CONTROL

AJ832
<u>RIGHT</u> <u>ARGUMENT</u>- BOOLEAN OR INTEGER SCALAR OR 1 ELEMENT VECTOR. 1→ULTRAPLOT ON, 0→ULTRAPLOT OFF
<u>LEFT</u> <u>ARGUMENT</u>- ULTRAPLOT PRINT CHAR. DEFAULT IS PERIOD. CAN'T BE AN OVERSTRUCK APL CHAR

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ832/TRANSLATE.1

TYPE: ARRAY

SUMMARY: FOR AJ832. A TABLE OF THE CHARACTERS PRINTED BY THE AJ832 UPON RECEIPT OF 7 BIT ASCII CODES.

TIMESTAMP: 1983-02-28 08:26:37

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
A 128 ELEMENT CHARACTER VECTOR. THE ORIGIN-0 POSITION OF A CHARACTER IN THE TABLE INDICATES THE ASCII TRANSMISSION CODE THAT IS SENT TO THE AJ832 TO CAUSE THE IT TO PRINT THE CHARACTER. THAT IS, IF <X> IS A 7 BIT ASCII CODE SENT TO THE TERMINAL, TRANSLATE[⎕IO+2⊥X] YIELDS THE CHARACTER THAT WILL BE PRINTED WHEN THE CODE IS RECEIVED BY THE TERMINAL.
THIS UTILITY IS USED BY SEVERAL OF THE UTILITIES IN RCM/ARBIO/*.

- -

TITLE: RCM/TERMFNS/AJ860/CLEAR∆HTABS.1 RCM/TERMFNS/AJ860/CLEAR∆HTABS.1

TYPE: FUNCTION

SUMMARY: FOR AJ860. CLEARS ALL HORIZONTAL TAB SETTINGS

TIMESTAMP: 1983-02-28 08:27:05

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
CLEAR∆HTABS

AJ860
CLEARS ALL HORIZONTAL TAB SETTINGS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ860/CLEAR∆VTABS.1

TYPE: FUNCTION

SUMMARY: FOR AJ860. CLEARS ALL VERTICAL TAB SETTINGS

TIMESTAMP: 1983-02-28 08:27:21

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
CLEAR∆VTABS

AJ860
CLEARS ALL VERTICAL TAB SETTINGS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/AJ860/EXPAND∆CHARS.1

TYPE: FUNCTION

SUMMARY: FOR AJ860. ENABLE OR DISABLE EXPANDED CHAR SET

TIMESTAMP: 1983-02-28 08:27:36

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
EXPAND∆CHARS CONTROL

AJ860
ENABLE OR DISABLE EXPANDED CHAR SET
ARGUMENT- BOOLEAN OR INTEGER SCALAR OR 1 ELEMENT VECTOR. 1→EXPANDED CHARS ON,
0→EXPANDED CHARS OFF

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ860/GOTO.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ860. ABSOLUTE TAB TO SPECIFIED LOCATION*

TIMESTAMP: 1983-02-28 08:27:49

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*
23 *TABS*

DESCRIPTION:
AXIS GOTO POSITION

AJ860
ABSOLUTE TAB TO SPECIFIED LOCATION
RIGHT ARGUMENT- INTEGER VECTOR OF 1 OR 2 ELEMENTS. SPECIFIES LOCATION ON AXIS DESIGNATED BY LEFT ARGUMENT
LEFT ARGUMENT- CHAR VECTOR OF 1 OR 2 ELEMENTS. 'X' OR 'Y' OR BOTH. SPECIFIES AXIS OF MOVE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ860/LINE∆FEED.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ860. GENERATES FULL AND HALF LINEFEEDS, UP OR DOWN.*

TIMESTAMP: 1983-02-28 08:28:04

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
LINE∆FEED N

AJ860
CAUSES VERTICAL MOVEMENT AS SPECIFIED
ARGUMENT- NUMERIC SCALAR SPECIFYING NUMBER OF FULL AND HALF LINEFEEDS
SIGN SPECIFIES DIRECTION(-UP),(+DOWN). HALF LF'S SPECIFIED BY USING DECIMAL FRACTION, IE. LINEFEED 1.5

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ860/PRINT.1

TYPE: FUNCTION

SUMMARY: FOR AJ860. PRINTS TEXT. USES ⎕ARBOUT.

TIMESTAMP: 1983-02-28 08:28:18

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
PRINT TEXT

AJ860
PRINTS THE TEXT AT THE TERMINAL. USES ⎕ARBOUT.
ARGUMENT- CHAR SCALAR OR VECTOR OF CHARS TO BE DISPLAYED. NO APL OVERSTRIKES

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/AJ860/PRINT∆ENABLE.1

TYPE: FUNCTION

SUMMARY: FOR AJ860. ENABLE OR DISABLES TERMINAL PRINTING

TIMESTAMP: 1983-02-28 08:28:33

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
PRINT∆ENABLE CONTROL

AJ860
ENABLE OR DISABLES TERMINAL PRINTING
ARGUMENT- BOOLEAN OR INTEGER SCALAR OR 1 ELEMENT VECTOR. 1→PRINT ON, 0→PRINT OFF

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ860/RESET∆MARGINS.1

RCM/TERMFNS/AJ860/RESET∆MARGINS.1

TYPE: FUNCTION

SUMMARY: FOR AJ860. LEFT MARGIN RESTORED TO 0, RIGHT MARGIN TO 131

TIMESTAMP: 1983-02-28 08:28:47

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
RESET∆MARGINS

AJ860
LEFT MARGIN RESTORED TO 0, RIGHT MARGIN TO 131

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/AJ860/RESET∆TERM.1

RCM/TERMFNS/AJ860/RESET∆TERM.1

TYPE: FUNCTION

SUMMARY: FOR AJ860. RESTORES TERMINAL SETTINGS TO THE STATE THEY WERE IN WHEN THE TERMINAL WAS POWERED UP.

TIMESTAMP: 1983-02-28 08:29:00

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
RESET∆TERM

AJ860
RESTORES TERMINAL SETTINGS TO THE STATE THEY WERE IN WHEN THE TERMINAL WAS POWERED UP.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ860/SET∆HTABS.1 RCM/TERMFNS/AJ860/SET∆HTABS.1

TYPE: FUNCTION

SUMMARY: FOR AJ860. SETS HORIZONTAL TABS

TIMESTAMP: 1983-02-28 08:26:52

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
SET∆HTABS N

AJ860
SETS HORIZONTAL TABS
ARGUMENT- INTEGER SCALAR CAUSES TABS TO BE SET AT INTERVALS OF N SPACES FROM LEFT MARGIN
INTEGER VECTOR CAUSES TABS TO BE SET AT THE ABSOLUTE POSITIONS SPECIFIED

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ860/SET∆MARGIN.1 RCM/TERMFNS/AJ860/SET∆MARGIN.1

TYPE: FUNCTION

SUMMARY: FOR AJ860. SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED

TIMESTAMP: 1983-02-28 08:29:14

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
CONTROL SET∆MARGIN POSITION

AJ860
SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RIGHT ARGUMENT- INTEGER SCALAR SPECIFYING POSITION FROM LEFT MARGIN
LEFT ARGUMENT- CHAR SCALAR SPECIFYING MARGIN EDGE. 'R'→RIGHT, 'L'→LEFT

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ860/SET∆PAGINATION.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ860. SETS NUMBER OF LINES TO BE SKIPPED AT BOTTOM OF PAGE*

TIMESTAMP: 1983-02-28 08:29:27

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
SET∆PAGINATION N

AJ860
SETS NUMBER OF LINES TO BE SKIPPED AT BOTTOM OF PAGE
ARGUMENT- INTEGER SCALAR OR 1 ELEMENT VECTOR SPECIFYING NUMBER OF LINES TO SKIP

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: *RCM/TERMFNS/AJ860/SET∆TOF.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ860. SETS TOP-OF-FORM*

TIMESTAMP: 1983-02-28 08:29:40

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
SET∆TOF

AJ860
SETS TOP-OF-FORM

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ860/SET∆TOF∆LENGTH.1 RCM/TERMFNS/AJ860/SET∆TOF∆LENGTH.1

TYPE: FUNCTION

SUMMARY: FOR AJ860. SETS TOP-OF-FORM AND FORM LENGTH

TIMESTAMP: 1983-02-28 08:29:53

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
SET∆TOF∆LENGTH N

AJ860
SETS TOP-OF-FORM AND FORM LENGTH
<N> IS THE REQUIRED FORM LENGTH.
THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ860/SET∆VTABS.1 RCM/TERMFNS/AJ860/SET∆VTABS.1

TYPE: FUNCTION

SUMMARY: FOR AJ860. SETS VERTICAL TAB STOPS

TIMESTAMP: 1983-02-28 08:30:08

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
SET∆VTABS N

AJ860
SETS VERTICAL TAB STOPS
ARGUMENT- INTEGER SCALAR CAUSES TABS TO BE SET AT INTERVALS OF N LINES FROM TOP OF FORM
INTEGER VECTOR CAUSES TABS TO BE SET AT ABSOLUTE LINES SPECIFIED

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ860/SUBSCRIPT.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ860. PRINTS TEXT AS A SUBSCRIPT*

TIMESTAMP: 1983-02-28 08:30:21

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
SUBSCRIPT TEXT

AJ860
PRINTS TEXT AS A SUBSCRIPT
ARGUMENT- CHAR SCALAR OR VECTOR TO BE PRINTED- NO OVERSTRUCK APL CHARS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/AJ860/SUPERSCRIPT.1*

TYPE: *FUNCTION*

SUMMARY: *FOR AJ860. PRINTS TEXT AS SUPERSCRIPT*

TIMESTAMP: 1983-02-28 08:30:34

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
SUPERSCRIPT TEXT

AJ860
PRINTS TEXT AS SUPERSCRIPT
ARGUMENT- CHAR SCALAR OR VECTOR TO BE PRINTED- NO OVERSTRUCK APL CHARS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/AJ860/TRANSLATE.1 RCM/TERMFNS/AJ860/TRANSLATE.1

TYPE: ARRAY

SUMMARY: FOR AJ860. A TABLE OF THE CHARACTERS PRINTED BY THE AJ860 UPON RECEIPT OF 7 BIT ASCII CODES.

TIMESTAMP: 1983-02-28 08:30:49

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
A 128 ELEMENT CHARACTER VECTOR. THE ORIGIN-0 POSITION OF A CHARACTER IN THE TABLE INDICATES THE ASCII TRANSMISSION CODE THAT IS SENT TO THE AJ860 TO CAUSE THE IT TO PRINT THE CHARACTER. THAT IS, IF <X> IS A 7 BIT ASCII CODE SENT TO THE TERMINAL, TRANSLATE[⎕IO+2⍳X] YIELDS THE CHARACTER THAT WILL BE PRINTED WHEN THE CODE IS RECEIVED BY THE TERMINAL.
THIS UTILITY IS USED BY SEVERAL OF THE UTILITIES IN RCM/ARBIO/*.

- -

TITLE: RCM/TERMFNS/DIABLO1620/AUTO∆LINEFEED.1 RCM/TERMFNS/DIABLO1620/AUTO∆LINEFEED.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620. CAUSES TERMINAL TO PRINT WITH LINES DOUBLE-SPACED

TIMESTAMP: 1983-02-28 08:31:35

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
AUTO∆LINEFEED

DIABLO1620
CAUSES TERMINAL TO PRINT WITH LINES DOUBLE-SPACED

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/DIABLO1620/CLEAR∆HTABS.1*

TYPE: *FUNCTION*

SUMMARY: *FOR DIABLO 1620. CLEARS ALL HORIZONTAL TABS*

TIMESTAMP: 1983-02-28 08:31:51

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*
23 *TABS*

DESCRIPTION:
CLEAR∆HTABS

DIABLO1620
CLEARS ALL HORIZONTAL TABS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/DIABLO1620/GOTO.1*

TYPE: *FUNCTION*

SUMMARY: *FOR DIABLO 1620. ABSOLUTE TAB TO SPECIFIED LOCATION*

TIMESTAMP: 1983-02-28 08:32:06

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*
23 *TABS*

DESCRIPTION:
CONTROL GOTO N

DIABLO1620
ABSOLUTE TAB TO SPECIFIED LOCATION
RIGHT ARGUMENT- INTEGER VECTOR OF 1 OR 2 ELEMENTS, SPECIFYING LOCATION ON AXIS CHOSEN
LEFT ARGUMENT- CHAR VECTOR OF 1 OR 2 ELEMENTS, 'X' OR 'Y' OR COMBINATION. SPECIFIES AXIS OF MOVE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/DIABLO1620/LINE∆FEED.1 RCM/TERMFNS/DIABLO1620/LINE∆FEED.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620. CAUSES PLATEN TO MOVE AS SPECIFIED

TIMESTAMP: 1983-02-28 08:32:21

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
LINE∆FEED N

DIABLO1620
CAUSES PLATEN TO MOVE AS SPECIFIED
ARGUMENT- NUMERIC SCALAR OR 1-ELEMENT VECTOR SPECIFYING NUMBER OF FULL AND HALF LINEFEED'S
SIGN SPECIFIES DIRECTION (-UP),(+DOWN). HALF LF'S SPECIFIED BY USING DECIMAL FRACTION, IE. LINE∆FEED 1.5

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/DIABLO1620/PLOT∆MODE.1 RCM/TERMFNS/DIABLO1620/PLOT∆MODE.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620. ARGUMENT- BOOLEAN SCALAR OR 1-ELEMENT VECTOR. 1→GRAPHICS, 0→GRAPHICS OFF

TIMESTAMP: 1983-02-28 08:32:37

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
PLOT∆MODE SWITCH

DIABLO1620
ARGUMENT- BOOLEAN SCALAR OR 1-ELEMENT VECTOR. 1→GRAPHICS, 0→GRAPHICS OFF

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/DIABLO1620/PRINT.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620. PRINTS TEXT. USES ⎕ARBOUT.

TIMESTAMP: 1983-02-28 08:32:51

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
PRINT TEXT

DIABLO1620
PRINTS THE TEXT AT THE TERMINAL. USES ⎕ARBOUT.
ARGUMENT- CHAR SCALAR OR VECTOR OF CHARS TO BE DISPLAYED. NO APL OVERSTRIKES

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/DIABLO1620/PRINT∆COLOR.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620. SELECTS COLOR OF RIBBON

TIMESTAMP: 1983-02-28 08:31:20

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
PRINT∆COLOR SWITCH

DIABLO1620
SELECTS COLOR OF RIBBON
ARGUMENT- BOOLEAN OR INTEGER SCALAR. 0→LOWER HALF OF RIBBON,1→UPPER HALF OF
RIBBON

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/DIABLO1620/PRINT∆DIRECTION.1 RCM/TERMFNS/DIABLO1620/PRINT∆DIRECTION.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620; SETS PRINT DIRECTION (FORWARD OR BACKWARD).

TIMESTAMP: 1983-02-28 08:33:06

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
PRINT∆DIRECTION CONTROL

DIABLO1620
ARGUMENT- CHAR SCALAR OR VECTOR. FIRST ELEMENT SIGNIFIES DIRECTION.
'F'→FORWARD, 'B'→BACKWARDS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/DIABLO1620/SET∆HTABS.1 RCM/TERMFNS/DIABLO1620/SET∆HTABS.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620. SETS HORIZONTAL TABS

TIMESTAMP: 1983-02-28 08:31:05

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
SET∆HTABS N

DIABLO1620
SETS HORIZONTAL TABS
ARGUMENT- INTEGER SCALAR CAUSES TABS TO BE SET AT INTERVALS OF N SPACES FROM
LEFT MARGIN
INTEGER VECTOR CAUSES TABS TO BE SET AT THE ABSOLUTE POSITIONS
SPECIFIED

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/DIABLO1620/SET∆MARGIN.1 RCM/TERMFNS/DIABLO1620/SET∆MARGIN.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620. SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED

TIMESTAMP: 1983-02-28 08:33:23

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
CONTROL SET∆MARGIN N

DIABLO1620
SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RIGHT ARGUMENT- INTEGER SCALAR OR 1 ELEMENT VECTOR SPECIFYING POSITION
LEFT ARGUMENT- CHAR SCALAR OR VECTOR. FIRST ELEMENT SIGNIFIES EDGE.
'R'→RIGHT, 'L'→LEFT

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/DIABLO1620/SET∆PITCH.1 RCM/TERMFNS/DIABLO1620/SET∆PITCH.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620. SET INCREMENT FOR HORIZONTAL OR VERTICAL MOVEMENT

TIMESTAMP: 1983-02-28 08:33:40

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
CONTROL SET∆PITCH N

DIABLO1620
SET INCREMENT FOR HORIZONTAL OR VERTICAL MOVEMENT
RIGHT ARGUMENT- INTEGER SCALAR SPECIFYING PITCH- NUMBER OF INCREMENTS PER
INCH
LEFT ARGUMENT- CHAR SCALAR OR VECTOR. FIRST ELEMENT SPECIFIES AXIS.
'H'-HORIZONTAL, 'V'-VERTICAL

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/DIABLO1620/SET∆TOF.1 RCM/TERMFNS/DIABLO1620/SET∆TOF.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620. ALLOWS USER TO SET TOP OF FORM

TIMESTAMP: 1983-02-28 08:33:55

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
SET∆TOF

DIABLO1620
ALLOWS USER TO SET TOP OF FORM

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/DIABLO1620/SUBSCRIPT.1 RCM/TERMFNS/DIABLO1620/SUBSCRIPT.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620. PRINTS TEXT AS A SUBSCRIPT

TIMESTAMP: 1983-02-28 08:34:08

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
SUBSCRIPT TEXT

DIABLO1620
PRINTS TEXT AS A SUBSCRIPT
ARGUMENT- CHAR SCALAR OR VECTOR TO BE PRINTED- NO OVERSTRUCK APL CHARS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/DIABLO1620/SUPERSCRIPT.1 RCM/TERMFNS/DIABLO1620/SUPERSCRIPT.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620. PRINTS TEXT AS SUPERSCRIPT

TIMESTAMP: 1983-02-28 08:34:23

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
SUPERSCRIPT TEXT

DIABLO1620
PRINTS TEXT AS SUPERSCRIPT
ARGUMENT- CHAR SCALAR OR VECTOR TO BE PRINTED- NO OVERSTRUCK APL CHARS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/DIABLO1620/TWELVE∆PITCH.1 RCM/TERMFNS/DIABLO1620/TWELVE∆PITCH.1

TYPE: FUNCTION

SUMMARY: FOR DIABLO 1620. CAUSES TERMINAL TO PRINT 12 CHARS/INCH

TIMESTAMP: 1983-02-28 08:34:36

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
TWELVE∆PITCH

DIABLO1620
CAUSES TERMINAL TO PRINT 12 CHARS/INCH

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/DIABLO1620/TRANSLATE.1 RCM/TERMFNS/DIABLO1620/TRANSLATE.1

TYPE: ARRAY

SUMMARY: FOR DIABLO 1620. A TABLE OF THE CHARS PRINTED BY THE DIABLO1620 UPON RECEIPT OF 7 BIT ASCII CODES.

TIMESTAMP: 1983-02-28 08:35:43

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
A 128 ELEMENT CHARACTER VECTOR. THE ORIGIN-0 POSITION OF A CHARACTER IN THE TABLE INDICATES THE ASCII TRANSMISSION CODE THAT IS SENT TO THE DIABLO1620 TO CAUSE THE IT TO PRINT THE CHARACTER. THAT IS, IF <X> IS A 7 BIT ASCII CODE SENT TO THE TERMINAL, TRANSLATE[⎕IO+2⊥X] YIELDS THE CHARACTER THAT WILL BE PRINTED WHEN THE CODE IS RECEIVED BY THE TERMINAL.
THIS UTILITY IS USED BY SEVERAL OF THE UTILITIES IN RCM/ARBIO/*.

- -

TITLE: RCM/TERMFNS/HP2641/CLEAR∆HTABS.1 RCM/TERMFNS/HP2641/CLEAR∆HTABS.1

TYPE: FUNCTION

SUMMARY: FOR HP2641. CLEARS ALL HORIZONTAL TABS.

TIMESTAMP: 1983-02-28 08:35:57

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS
31 FULL-SCREEN HANDLING

DESCRIPTION:
CLEAR∆HTABS

HP2641
CLEARS ALL HORIZONTAL TABS.
THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/HP2641/CONTROL∆CODES.1 RCM/TERMFNS/HP2641/CONTROL∆CODES.1

TYPE: FUNCTION

SUMMARY: FOR HP2641. ALLOWS CONTROL CODES TO DISPLAYED INSTEAD OF EXECUTED

TIMESTAMP: 1983-02-28 08:36:11

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
31 FULL-SCREEN HANDLING

DESCRIPTION:
CONTROL∆CODES SWITCH

HP2641
ALLOWS CONTROL CODES TO DISPLAYED INSTEAD OF EXECUTED
ARGUMENT- 1→CONTROL FUNCTIONS DISABLED AND DISPLAYED
0→CONTROL FUNCTIONS ENABLED(NORMAL STATE)

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/HP2641/CURSOR.1 RCM/TERMFNS/HP2641/CURSOR.1

TYPE: FUNCTION

SUMMARY: FOR HP2641. MOVES CURSOR UP,DOWN,RIGHT OR LEFT N POSITIONS

TIMESTAMP: 1983-02-28 08:36:25

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
31 FULL-SCREEN HANDLING

DESCRIPTION:
CONTROL CURSOR N

HP2641
MOVES CURSOR UP,DOWN,RIGHT OR LEFT N POSITIONS
RIGHT ARGUMENT- INTEGER SCALAR OR 1-ELEMENT VECTOR SPECIFYING NUMBER OF POSITIONS
LEFT ARGUMENT- CHAR SCALAR OR VECTOR. FIRST ELEMENT SIGNIFIES DIRECTION.
'U'→UP,'D'→DOWN,'R'→RIGHT,'L'→LEFT

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/HP2641/DISPLAY.1 RCM/TERMFNS/HP2641/DISPLAY.1

TYPE: FUNCTION

SUMMARY: FOR HP2641. DISPLAYS APL FUNCTIONS USING SPECIAL FEATURES OF THE HP2641.

TIMESTAMP: 1983-02-28 08:36:41

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
31 FULL-SCREEN HANDLING

DESCRIPTION:
CONTROL DISPLAY FUNCTION

DISPLAYS APL FUNCTIONS USING SPECIAL FEATURES OF HP2641
RIGHT ARGUMENT- VECTOR REPRESENTATION OF AN APL FUNCTION(1 ⎕FD FORM) OR NAME OF UNLOCKED FUNCTION IN ACTIVE WS
LEFT ARGUMENT- CHAR VECTOR OR MATRIX TO CONTROL DISPLAY FORM

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/HP2641/GOTO.1 RCM/TERMFNS/HP2641/GOTO.1

TYPE: FUNCTION

SUMMARY: FOR HP2641. ABSOLUTE TAB TO SPECIFIED LOCATION

TIMESTAMP: 1983-02-28 08:36:59

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS
31 FULL-SCREEN HANDLING

DESCRIPTION:
CONTROL GOTO N

HP2641
ABSOLUTE TAB TO SPECIFIED LOCATION
RIGHT ARGUMENT- INTEGER VECTOR OF 2 ELEMENTS. SPECIFIES LOCATION ON AXIS DESIGNATED BY LEFT ARGUMENT.
LEFT ARGUMENT- CHAR VECTOR OF 2 ELEMENTS. 'XY' OR 'YX'. SPECIFIES AXIS OF MOVE.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/HP2641/MENU.1

TYPE: FUNCTION

SUMMARY: FOR HP2641. PRESENTS A MENU OF CHOICES TO THE USER AND RETURNS THE USER'S CHOICE

TIMESTAMP: 1983-02-28 08:37:14

CATEGORIES:
20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
31 FULL-SCREEN HANDLING

DESCRIPTION:
CHOICE←MENU OPTIONS

HP2641
PRESENTS A MENU OF CHOICES TO THE USER AND RETURNS THE USER'S CHOICE
ARGUMENT- CHAR MATRIX OF OPTION NAMES. 24 39∧.≥⍴OPTIONS
RESULT- INTEGER ROW INDEX OF OPTION CHOICE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/HP2641/PAGE.1

TYPE: FUNCTION

SUMMARY: FOR HP2641. DISPLAY THE NTH PREVIOUS OR NEXT PAGE

TIMESTAMP: 1983-02-28 08:37:28

CATEGORIES:
20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
31 FULL-SCREEN HANDLING

DESCRIPTION:
CONTROL PAGE N

HP2641
DISPLAY THE NTH PREVIOUS OR NEXT PAGE
RIGHT ARGUMENT- INTEGER SCALAR OR 1-ELEMENT VECTOR PAGE NUMBER
LEFT ARGUMENT- CHAR SCALAR RO VECTOR. FIRST ELEMENT SIGNIFIES DIRECTION.
'P'→PREVIOUS,'N'→NEXT

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/HP2641/PRINT.1 RCM/TERMFNS/HP2641/PRINT.1

TYPE: FUNCTION

SUMMARY: FOR HP2641. PRINTS TEXT. USES ⎕ARBOUT.

TIMESTAMP: 1983-02-28 08:37:43

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
31 FULL-SCREEN HANDLING

DESCRIPTION:
PRINT TEXT

HP2641
PRINTS TEXT. USES ⎕ARBOUT.
THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/HP2641/RESET∆TERM.1 RCM/TERMFNS/HP2641/RESET∆TERM.1

TYPE: FUNCTION

SUMMARY: FOR HP2641. RESETS TERMINAL TO POWER-ON STATE

TIMESTAMP: 1983-02-28 08:38:00

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
31 FULL-SCREEN HANDLING

DESCRIPTION:
RESET∆TERM

HP2641
RESETS TERMINAL TO POWER-ON STATE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/HP2641/SCROLL.1

TYPE: FUNCTION

SUMMARY: FOR HP2641. SCROLL DISPLAY UP OR DOWN N LINES

TIMESTAMP: 1983-02-28 08:38:15

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
31 FULL-SCREEN HANDLING

DESCRIPTION:
CONTROL SCROLL N

HP2641
SCROLL DISPLAY UP OR DOWN N LINES
RIGHT ARGUMENT- INTEGER SCALAR OR 1-ELEMENT VECTOR SPECIFYING NUMBER OF LINES
LEFT ARGUMENT- CHAR SCALAR OR VECTOR. FIRST ELEMENT SIGNIFIES DIRECTION.'U'→UP,'D'→DOWN

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/HP2641/SET∆HTABS.1

TYPE: FUNCTION

SUMMARY: FOR HP2641. SETS HORIZONTAL TABS

TIMESTAMP: 1983-02-28 08:38:30

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS
31 FULL-SCREEN HANDLING

DESCRIPTION:
SET∆HTABS N

HP2641
SETS HORIZONTAL TABS
ARGUMENT- INTEGER SCALAR CAUSES TABS TO BE SET AT INTERVALS OF N SPACES FROM THE LEFT MARGIN
INTEGER VECTOR CAUSES TABS TO BE SET AT THE ABSOLUTE POSITIONS SPECIFIED

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/HP2641/SET∆MARGIN.1

TYPE: FUNCTION

SUMMARY: FOR HP2641. SET RIGHT OR LEFT MARGIN AT POSITION SPECIFIED

TIMESTAMP: 1983-02-28 08:38:51

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
31 FULL-SCREEN HANDLING

DESCRIPTION:
CONTROL SET∆MARGIN N

HP2641
SET RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RIGHT ARGUMENT- INTEGER SCALAR OR 1-ELEMENT VECTOR SPECIFYING MARGIN POSITION
LEFT ARGUMENT- CHAR SCALAR OR VECTOR. FIRST ELEMENT SIGNIFIES EDGE.'R'→
RIGHT,'L'→LEFT

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/HP2641/TRANSLATE.1

TYPE: ARRAY

SUMMARY: FOR HP2641. A TABLE OF THE CHARACTERS PRINTED BY THE HP2641 UPON RECEIPT OF 7 BIT ASCII CODES.

TIMESTAMP: 1983-02-28 08:39:05

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
31 FULL-SCREEN HANDLING

DESCRIPTION:
A 128 ELEMENT CHARACTER VECTOR. THE ORIGIN-0 POSITION OF A CHARACTER IN THE
TABLE INDICATES THE ASCII TRANSMISSION CODE THAT IS SENT TO THE HP2641 TO CAUSE
THE IT TO PRINT THE CHARACTER. THAT IS, IF <X> IS A 7 BIT ASCII CODE SENT TO
THE TERMINAL, TRANSLATE[⎕IO+2⊥X] YIELDS THE CHARACTER THAT WILL BE PRINTED WHEN
THE CODE IS RECEIVED BY THE TERMINAL.
THIS UTILITY IS USED BY SEVERAL OF THE UTILITIES IN RCM/ARBIO/*.

TITLE: RCM/TERMFNS/TRENDATA4000A/CLEAR∆HTABS.1 RCM/TERMFNS/TRENDATA4000A/CLEAR∆HTABS.1

TYPE: FUNCTION

SUMMARY: FOR TRNDATA 4000A. CLEARS ALL HORIZONTAL TAB SETTINGS

TIMESTAMP: 1983-02-28 08:44:34

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
CLEAR∆HTABS

TRENDATA4000A
CLEARS ALL HORIZONTAL TAB SETTINGS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/TRENDATA4000A/CLEAR∆VTABS.1 RCM/TERMFNS/TRENDATA4000A/CLEAR∆VTABS.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. CLEARS ALL VERTICAL TAB SETTINGS

TIMESTAMP: 1983-02-28 08:44:46

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
CLEAR∆VTABS

TRENDATA4000A
CLEARS ALL VERTICAL TAB SETTINGS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/TRENDATA4000A/GOTO.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. ABSOLUTE TAB TO SPECIFIED LOCATION

TIMESTAMP: 1983-02-28 08:45:00

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
AXIS GOTO POSITION

TRENDATA4000A
ABSOLUTE TAB TO SPECIFIED LOCATION
RIGHT ARGUMENT- INTEGER VECTOR OF 1 OR 2 ELEMENTS. SPECIFIES LOCATION ON AXIS DESIGNATED BY LEFT ARGUMENT
LEFT ARGUMENT- CHAR VECTOR OF 1 OR 2 ELEMENTS. 'X' OR 'Y' OR BOTH. SPECIFIES AXIS OF MOVE

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/TRENDATA4000A/LINE∆FEED.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. GENERATES FULL AND HALF LINEFEEDS, UP OR DOWN.

TIMESTAMP: 1983-02-28 08:45:14

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
LINE∆FEED N

TRENDATA4000A
CAUSES VERTICAL MOVEMENT AS SPECIFIED
ARGUMENT- NUMERIC SCALAR SPECIFYING NUMBER OF FULL AND HALF LINEFEEDS
SIGN SPECIFIES DIRECTION(-UP),(+DOWN). HALF LF'S SPECIFIED BY USING DECIMAL FRACTION, IE. LINEFEED 1.5

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/TRENDATA4000A/MOVE.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)

TIMESTAMP: 1983-02-28 08:44:24

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
DIRECTION MOVE DISTANCE

TRENDATA4000A
MOVE TO POSITION SPECIFIED BY DISTANCE FROM CURRENT POSITION (RELATIVE TAB)
RIGHT ARGUMENT- INTEGER VECTOR OF 1 OR MORE ELEMENTS. SPECIFIES DISTANCE (1/60 HORIZONTAL, 1/48 VERTICAL) TO BE MOVED
LEFT ARGUMENT - CHAR VECTOR OF 1 OR MORE ELEMENTS. SPECIFIES DIRECTION OF MOVE. 'L'→LEFT,'R'→RIGHT,'U'→UP,'D'→DOWN

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/TRENDATA4000A/PLOT.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. MOVE OR DRAW IN PLOT MODE

TIMESTAMP: 1983-02-28 08:45:28

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
CONTROL PLOT MATRIX

TRENDATA4000A
MOVE OR DRAW IN PLOT MODE
RIGHT ARGUMENT- N×2 NUMERIC MATRIX. [;1]=0→MOVE, =1→DRAW. [;2] SPECIFIES DISTANCE
LEFT ARGUMENT - N ELEMENT CHAR VECTOR. SPECIFIES DIRECTION. 'U'→UP,'D'→DOWN,'R'→RIGHT,'L'→LEFT

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/TRENDATA4000A/PLOT∆MODE.1 RCM/TERMFNS/TRENDATA4000A/PLOT∆MODE.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. TURNS ON OR OFF STANDARD PLOT MODE

TIMESTAMP: 1983-02-28 08:45:42

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
PLOT∆MODE CONTROL

TRENDATA4000A
TURNS ON OR OFF STANDARD PLOT MODE
ARGUMENT- BOOLEAN OR INTEGER SCALAR OR 1 ELEMENT VECTOR. 1→PLOT ON, 0→PLOT OFF

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/TRENDATA4000A/PRINT.1 RCM/TERMFNS/TRENDATA4000A/PRINT.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. PRINTS TEXT. USES ⎕ARBOUT.

TIMESTAMP: 1983-02-28 08:46:01

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
PRINT TEXT

TRENDATA4000A
PRINTS THE TEXT AT THE TERMINAL. USES ⎕ARBOUT.
ARGUMENT- CHAR SCALAR OR VECTOR OF CHARS TO BE DISPLAYED. NO APL OVERSTRIKES

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/TRENDATA4000A/PRINT∆COLOR.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. SELECTS COLOR OF RIBBON

TIMESTAMP: 1983-02-28 08:44:12

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
PRINT∆COLOR SWITCH

TRENDATA4000A
SELECTS COLOR OF RIBBON
ARGUMENT- BOOLEAN OR INTEGER SCALAR. 0→LOWER HALF OF RIBBON,1→UPPER HALF OF RIBBON

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/TRENDATA4000A/PRINT∆ENABLE.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. ENABLE OR DISABLES TERMINAL PRINTING

TIMESTAMP: 1983-02-28 08:46:22

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
PRINT∆ENABLE CONTROL

TRENDATA4000A
ENABLE OR DISABLES TERMINAL PRINTING
ARGUMENT- BOOLEAN OR INTEGER SCALAR OR 1 ELEMENT VECTOR. 1→PRINT ON, 0→PRINT OFF

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/TRENDATA4000A/RESET∆MARGINS.1 RCM/TERMFNS/TRENDATA4000A/RESET∆MARGINS.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. LEFT MARGIN IS RESTORED TO 0, RIGHT MARGIN RESTORED TO 131

TIMESTAMP: 1983-02-28 08:46:37

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
RESET∆MARGINS

TRENDATA4000A
LEFT MARGIN IS RESTORED TO 0, RIGHT MARGIN RESTORED TO 131

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/TRENDATA4000A/SELECT∆PITCH.1 RCM/TERMFNS/TRENDATA4000A/SELECT∆PITCH.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. SELECTS 1 OF 4 POSSIBLE PITCH SETTINGS: 10V/6H, 12V/6H, 10V/8H, 12V/8H

TIMESTAMP: 1983-02-28 08:44:02

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
SELECT∆PITCH N

TRENDATA4000A
SELECTS 1 OF 4 POSSIBLE PITCH SETTINGS: 1) 10V/6H, 2) 12V/6H, 3) 10V/8H, 4)
12V/8H
ARGUMENT- INTEGER SCALAR 1,2,3, OR 4

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/TRENDATA4000A/SET∆HTABS.1 RCM/TERMFNS/TRENDATA4000A/SET∆HTABS.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. SETS HORIZONTAL TAB STOPS

TIMESTAMP: 1983-02-28 08:42:27

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
SET∆HTABS N

TRENDATA4000A
SETS HORIZONTAL TAB STOPS
ARGUMENT- INTEGER SCALAR CAUSES TABS TO BE SET AT INTERVALS OF N SPACES FROM LEFT MARGIN
INTEGER VECTOR CAUSES TABS TO BE SET AT INTERVALS OF N SPACES FROM LEFT MARGIN

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/TRENDATA4000A/SET∆MARGIN.1 RCM/TERMFNS/TRENDATA4000A/SET∆MARGIN.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED

TIMESTAMP: 1983-02-28 08:46:52

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
CONTROL SET∆MARGIN POSITION

TRENDATA4000A
SETS RIGHT OR LEFT MARGIN AT POSITION SPECIFIED
RIGHT ARGUMENT- INTEGER SCALAR SPECIFYING POSITION FROM LEFT MARGIN
LEFT ARGUMENT- CHAR SCALAR SPECIFYING MARGIN EDGE. 'R'→RIGHT, 'L'→LEFT

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: *RCM/TERMFNS/TRENDATA4000A/SET∆PAGINATION.1* *RCM/TERMFNS/TRENDATA4000A/SET∆PAGINATION.1*

TYPE: *FUNCTION*

SUMMARY: *FOR TRENDATA 4000A. SETS NUMBER OF LINES TO BE SKIPPED AT BOTTOM OF PAGE*

TIMESTAMP: 1983-02-28 08:47:04

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
SET∆PAGINATION N

TRENDATA4000A
SETS NUMBER OF LINES TO BE SKIPPED AT BOTTOM OF PAGE
ARGUMENT- INTEGER SCALAR OR 1 ELEMENT VECTOR SPECIFYING NUMBER OF LINES TO SKIP

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: *RCM/TERMFNS/TRENDATA4000A/SET∆TOF∆LENGTH.1* *RCM/TERMFNS/TRENDATA4000A/SET∆TOF∆LENGTH.1*

TYPE: *FUNCTION*

SUMMARY: *FOR TRENDATA 4000A. SETS TOP-OF-FORM AND FORM LENGTH*

TIMESTAMP: 1983-02-28 08:47:18

CATEGORIES: 20 *TERMINAL INPUT/OUTPUT*
22 *TERMINAL CONTROL*

DESCRIPTION:
SET∆TOF∆LENGTH N

TRENDATA4000A
SETS TOP-OF-FORM AND FORM LENGTH
ARGUMENT- INTEGER SCALAR SPECIFYING NUMBER OF LINES IN FORM

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/TRENDATA4000A/SET∆VTABS.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. SETS VERTICAL TAB STOPS

TIMESTAMP: 1983-02-28 08:47:33

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

DESCRIPTION:
SET∆VTABS N

TRENDATA4000A
SETS VERTICAL TAB STOPS
ARGUMENT- INTEGER SCALAR CAUSES TABS TO BE SET AT INTERVALS OF N LINES FROM TOP OF FORM
INTEGER VECTOR CAUSES TABS TO BE SET AT ABSOLUTE LINES SPECIFIED

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/TRENDATA4000A/SUBSCRIPT.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. PRINTS TEXT AS A SUBSCRIPT

TIMESTAMP: 1983-02-28 08:47:47

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
SUBSCRIPT TEXT

TRENDATA4000A
PRINTS TEXT AS A SUBSCRIPT
ARGUMENT- CHAR SCALAR OR VECTOR TO BE PRINTED- NO OVERSTRUCK APL CHARS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: RCM/TERMFNS/TRENDATA4000A/SUPERSCRIPT.1 RCM/TERMFNS/TRENDATA4000A/SUPERSCRIPT.1

TYPE: FUNCTION

SUMMARY: FOR TRENDATA 4000A. PRINTS TEXT AS SUPERSCRIPT

TIMESTAMP: 1983-02-28 08:48:02

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
SUPERSCRIPT TEXT

TRENDATA4000A
PRINTS TEXT AS SUPERSCRIPT
ARGUMENT- CHAR SCALAR OR VECTOR TO BE PRINTED- NO OVERSTRUCK APL CHARS

THIS FUNCTION IS ⎕IO-INDEPENDENT.

- -

TITLE: RCM/TERMFNS/TRENDATA4000A/TRANSLATE.1 RCM/TERMFNS/TRENDATA4000A/TRANSLATE.1

TYPE: ARRAY

SUMMARY: FOR TRENDATA 4000A. A TABLE OF THE CHARS PRINTED BY THE 4000A UPON RECEIPT OF 7 BIT ASCII CODES.

TIMESTAMP: 1983-02-28 08:48:16

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL

DESCRIPTION:
A 128 ELEMENT CHARACTER VECTOR. THE ORIGIN-0 POSITION OF A CHARACTER IN THE TABLE INDICATES THE ASCII TRANSMISSION CODE THAT IS SENT TO THE TRENDATA4000A TO CAUSE THE IT TO PRINT THE CHARACTER. THAT IS, IF <X> IS A 7 BIT ASCII CODE SENT TO THE TERMINAL, TRANSLATE[⎕IO+2⊥X] YIELDS THE CHARACTER THAT WILL BE PRINTED WHEN THE CODE IS RECEIVED BY THE TERMINAL.
THIS UTILITY IS USED BY SEVERAL OF THE UTILITIES IN RCM/ARBIO/*.

TITLE: RMILL/FCOMPCOPY.1 RMILL/FCOMPCOPY.1

TYPE: FUNCTION

SUMMARY: NEARLY WSFULL PROOF COPY ONE FILE COMPONENT TO ANOTHER

TIMESTAMP: 1983-12-22 19:09:57

CATEGORIES:
9 FILES
13 FILE TOOLS
43 FILE PRIMITIVE SIMULATION

DESCRIPTION:

```
PURPOSE - USED INSTEAD OF EITHER ¨(⎕READ T1,C1) ⎕REPLACE T2,C2¨ OR
          ¨(⎕READ T1,C1) ⎕APPEND T2¨. THE SYNTAX TO DO EITHER THESE
          IS:   (T1,C1) FCOMPCOPY T2,C2    ⍝ FOR THE REPLACE
          OR:   (T1,C1) FCOMPCOPY T2       ⍝ FOR THE APPEND
          HOWEVER, IF FCOMPCOPY HITS A WSFULL IT RUNS A SPLITWS NTASK
          WHICH EXPUNGES ALL OBJECTS IN THE WS NOT BEING USED AND THEN
          TRIES AGAIN. ANY ERRORS HIT IN THE NTASK ARE COMMUNICATED BACK
          TO THE TASK THAT CALLED FCOMPCOPY VIA SHARED VARIABLES. FILE
          FULL IS ALSO TRAPPED AND THE NEW FILE IS RESIZED IF POSSIBLE.
..............BOB MILLER (RMILL)
```

TITLE: RMILL/ROLLAPPENDR.1 RMILL/ROLLAPPENDR.1

TYPE: FUNCTION

SUMMARY: ⎕APPENDR FOR A ROLLING FILE

TIMESTAMP: 1983-12-20 19:05:36

CATEGORIES:
9 FILES
13 FILE TOOLS

DESCRIPTION:

```
SYNTAX: R←DATA ROLLAPPENDR PARMS        ⍝PARMS: TIE NUMBER, FILE SIZE, PASSNO
ACTION: IF NECESSARY, ⎕DROPS COMPONENTS OFF OF FILE TIED TO TIE NUMBER, AND
        APPENDS DATA TO THE FILE. ENOUGH COMPONENTS ARE DROPPED OFF THE FILE
        SO THAT IT HAS A MAXIMUM OF PARMS[1+⎕IO] COMPONENTS AFTER THE APPEND.
        PARMS MUST BE AT LEAST SHAPE 2 - THE FILE PASSNUMBER IS OPTIONAL.
        RETURNS THE COMPONENT NUMBER CONTAINING <DATA>.
........BOB MILLER (RMILL)
```

```
TITLE:       RMILL/SPINWSDOC.3                                                    RMILL/SPINWSDOC.3

TYPE:        FUNCTION

SUMMARY:     SUBMIT A WSDOC OF THE OBJECTS IN A PACKAGE

TIMESTAMP:   1983-11-04 18:38:47

CATEGORIES:   8  PACKAGES
             35  WORKSPACE TOOLS  ⍝ E.G. WORKSPACE CROSS REFERENCE

DESCRIPTION:
PURPOSE - RUN A WSDOC OF THE ELEMENTS OF A PACKAGE IN A LOCAL ENVIRONMENT.
SYNTAX  - SPINWSDOC PKG     <PKG> A SAPL PACKAGE
BEHAVIOR- RUNS AN NTASK INTO WHICH IS PDEF'D THE OBJECTS IN PKG - ALL OTHER
          OBJECTS ARE SHADOWED. THE NTASK THEN DOES A WSTOFILE AND ⎕LOAD'S
          STATE. IT IS ASSUMED THAT YOU HAVE PROFILED THE STATE WORKSPACE.
          MEANWHILE, THE TTASK WAITS FOR THE NTASK TO COMPLETE. WHEN THE
          NTASK COMPLETES, THE WSDOC SOURCE FILE IS TIED, WITH THE TIE
          NUMBER IN ⎕SP.
AUTHOR -  BOB MILLER/ROCHESTER

*** EXECUTE   DETAILS UDESCRIBE 'RMILL/SPINWSDOC.3'   FOR MORE INFORMATION

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -


TITLE:       RMILL/TIED.1                                                          RMILL/TIED.1

TYPE:        FUNCTION

SUMMARY:     CHECKS TO SEE WHETHER A TIED FILE IS ⎕TIE'D OR ⎕STIE'D.

TIMESTAMP:   1984-01-24 15:05:10

CATEGORIES:   9  FILES
             11  CONCURRENT USE OF FILES  ⍝ FILE SHARING
             13  FILE TOOLS

DESCRIPTION:
SYNTAX: R←TIED TNO    <TNO>↔SINGLE TIE NUMBER
                      <R>↔1 IF THE FILE TIED TO TNO IS ⎕TIE'D
                          0 IF THE FILE TIED TO TNO IS ⎕STIE'D
                         ¯1 IF UNABLE TO PROCESS
        ∘ IF TNO ISN'T IN ⎕NUMS A FILE TIE ERROR IS ⎕SIGNAL'D.
METHOD: <TIED> STARTS A SPLIT-WS NTASK WHICH TRAPS EVENT NUMBER 24 (FILE TIED)
        AND TRIES TO ⎕STIE THE FILE TIED TO <TNO>. IF IT IS SUCCESSFUL, A
        VARIABLE SHARED WITH THE CALLING TASK IS SET TO 0. IF A FILE TIED IS
        GENERATED BY THE ⎕STIE ATTEMPT, THE SHARED VARIABLE IS SET TO ONE,
        AND IF ANYTHING ELSE GOES WRONG, THE SHARED VARIABLE IS SET TO ¯1.
        ⎕TRAP IN THE NTASK CONTAINS '∘ 2001 D CLEAR', SO THAT NORMALLY NO
        CRASH WORKSPACE IS SAVED.
....... CONTACT RMILL WITH PROBLEMS.
```

TITLE: *ROHAN/CENTRE.1*

TYPE: *FUNCTION*

SUMMARY: *CENTRES A VECTOR OF TEXT (SUCH AS A HEADING FOR A REPORT).*

TIMESTAMP: 1984-01-26 01:08:35

CATEGORIES:
25 *FORMATTING*
27 *TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS*
30 *REPORT FORMATTING*

DESCRIPTION:

RESULT←[WIDTH] CENTRE VECTOR:

CENTRES THE GIVEN <VECTOR> OF TEXT IN A FIELD OF THE SPECIFIED <WIDTH>, BY CATENATING LEADING AND TRAILING BLANKS AS REQUIRED. THE GIVEN VECTOR SHOULD NOT ALREADY HAVE ANY LEADING OR TRAILING BLANKS, NOR SHOULD IT CONTAIN ANY BACKSPACES. THE DEFAULT WIDTH IS ⎕PW.

IF THE GIVEN VECTOR IS ALREADY LONGER THAN THE WIDTH WILL ALLOW, A DOMAIN ERROR WILL OCCUR WITHIN THE FUNCTION.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

TITLE: ROHAN/COLUMNIZE.2

TYPE: FUNCTION

SUMMARY: GIVEN A NAMELIST (OR SIMILAR MATRIX), PUTS THE NAMES IN COLUMNS ACROSS THE PAGE.

TIMESTAMP: 1983-05-26 22:13:20

CATEGORIES: 25 FORMATTING
30 REPORT FORMATTING

CHANGES:
NOW PUTS A SPACE BETWEEN COLUMNS, SO THAT YOU DON'T HAVE TO CATENATE A SPACE TO THE ARGUMENT.

DESCRIPTION:
RESULT←WIDTH COLUMNIZE MATRIX (<WIDTH> IS OPTIONAL):
<MATRIX> IS A MATRIX OF CHARACTERS, WITH EACH ROW CONTAINING ONE NAME OR OTHER UNIT OF INFORMATION. <RESULT> HAS THE MATRIX SPLIT UP INTO COLUMNS WHICH ARE PLACED SIDE BY SIDE INTO A NEW MATRIX OF WIDTH <WIDTH> (THE DEFAULT WIDTH IS ⎕PW), WITH AT LEAST ONE BLANK BETWEEN COLUMNS. REQUIRES WIDTH≥¯1↑⍴MATRIX (OTHERWISE A DOMAIN ERROR WILL OCCUR INSIDE THE FUNCTION).

EXAMPLES:

```
      22 COLUMNIZE M←11 6⍴'MARY
                          HAD
                          A
                          LITTLE
                          LAMB
                          WHOSE
                          FLEECE
                          WAS
                          WHITE
                          AS
                          SNOW  '
MARY   LAMB   WHITE
HAD    WHOSE  AS
A      FLEECE SNOW
LITTLE WAS
      32 COLUMNIZE M
MARY    LITTLE  FLEECE  AS
HAD     LAMB    WAS     SNOW
A       WHOSE   WHITE
```

THE NUMBER OF COLUMNS IS ⌊(WIDTH+1)÷1+¯1↑⍴MATRIX; THE NUMBER OF ROWS IS ⌈(1↑⍴MATRIX)÷NUMBER∆OF∆COLUMNS. ⍴RESULT ↔ NUMBER∆OF∆ROWS,WIDTH.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

TITLE: ROHAN/COST.2

TYPE: FUNCTION

SUMMARY: RETURNS A TABLE SIMILAR TO THAT DISPLAYED AT SIGNOFF, WITH THE ACTUAL DOLLAR COST ALSO GIVEN.

TIMESTAMP: 1983-01-21 21:11:45

CATEGORIES: 36 MEASURING USAGE AND CHARGES

CHANGES:
RATES ARE EASIER TO CHANGE IN THIS VERSION (SUGGESTION BY J. HENRI SCHUELER).
THERE ARE OTHER SMALL IMPROVEMENTS.

DESCRIPTION:
Z←COST: RETURNS A TABLE SIMILAR TO THAT DISPLAYED AT SIGNOFF, WITH THE ACTUAL DOLLAR COST ALSO GIVEN. BASED ON RATES OF $1.00 PER CONNECT HOUR, $0.45 PER CPU UNIT, AND $0.70 PER KILOCHAR, BUT THESE RATES (SET ON LINE 4 OF THE FUNCTION) ARE EASY TO CHANGE.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

- -

TITLE: ROHAN/DAY.2

TYPE: FUNCTION

SUMMARY: RETURNS THE DAY OF THE WEEK ('MONDAY', 'TUESDAY', ETC.) THAT A DATE FALLS ON.

TIMESTAMP: 1983-04-05 22:36:23

CATEGORIES: 37 TIMES AND DATES

CHANGES:
SLIGHTLY MODIFIED TO TAKE ADVANTAGE OF THE NEW PRIMITIVE FUNCTION 'LINK'.

DESCRIPTION:
Z←DAY DATE: RETURNS THE DAY OF THE WEEK ('MONDAY', 'TUESDAY', ETC.) THAT THE GIVEN DATE FALLS ON. THE DATE MUST BE IN 3↑⎕TS FORMAT, E.G. 1982 11 21 . THE RESULT IS A VECTOR OF CHARACTERS.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS ADAPTED BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

TITLE: ROHAN/EASYTIE.2 ROHAN/EASYTIE.2

TYPE: FUNCTION

SUMMARY: TIES A FILE AND RETURNS THE TIE NUMBER, WITH PROVISION FOR LATER UNTYING A NEWLY-TIED FILE.

TIMESTAMP: 1983-12-02 18:50:08

CATEGORIES: 9 FILES
43 FILE PRIMITIVE SIMULATION

CHANGES:
THE CODE HAS BEEN SLIGHTLY IMPROVED.

DESCRIPTION:
SYNTAX: TIENUM←EASYTIE FILENAME. THE FILENAME MUST BE IN THE USUAL FORM REQUIRED BY ⎕TIE AND ⎕STIE.

<EASYTIE> TIES THE SPECIFIED FILE TO THE SMALLEST NUMBER NOT ALREADY IN USE, AND RETURNS THIS TIE NUMBER AS A SCALAR. IF THE FILE WAS ALREADY TIED THEN THE FUNCTION JUST RETURNS THE TIE NUMBER AS A SINGLE-ELEMENT VECTOR.

THIS FUNCTION IS INTENDED FOR USE WITH ROHAN/EASYUNTIE, WHICH WILL UNTIE A FILE ONLY IF <EASYTIE> ACTUALLY HAD TO TIE IT (AND NOT SIMPLY RETURN AN EXISTING TIE NUMBER). THUS <EASYUNTIE> WILL RESTORE THE TIE STATE OF A FILE TO WHAT IT HAD BEEN BEFORE <EASYTIE> WAS EXECUTED. FOR EXAMPLE:

```
      ∇ FOO;CTIE;UTIE
   [1]  CTIE←EASYTIE '1234567 CATALOGUE' ⋄ UTIE←EASYTIE 'UPDATES'
   [2]  (⎕READ UTIE,1) ⎕REPLACE CTIE,1
   [3]  EASYUNTIE CTIE ⋄ EASYUNTIE UTIE ⍝ NOTE THAT   EASYUNTIE CTIE,UTIE
   [4]                                  ⍝ WILL NOT WORK!
      ∇
```

<FOO> WILL LEAVE ⎕NUMS AND ⎕NAMES AS THEY WERE JUST BEFORE IT WAS EXECUTED.

ROHAN/EASYTIE OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

- -

TITLE: ROHAN/EASYUNTIE.1 ROHAN/EASYUNTIE.1

TYPE: FUNCTION

SUMMARY: UNTIES A FILE TIED BY ROHAN/EASYTIE.

TIMESTAMP: 1983-08-29 20:43:18

CATEGORIES: 9 FILES
43 FILE PRIMITIVE SIMULATION

DESCRIPTION:
UNTIES A FILE TIED BY ROHAN/EASYTIE. SEE THE DESCRIPTION OF ROHAN/EASYTIE FOR DETAILS.

TITLE: ROHAN/EDITMAT.1 ROHAN/EDITMAT.1

TYPE: FUNCTION

SUMMARY: CHARACTER-MATRIX EDITOR WHICH BEHAVES LIKE 'CH' IN MAILBOX.

TIMESTAMP: 1983-10-20 22:37:48

CATEGORIES: 24 EDITING

DESCRIPTION:
REVISED←EDITMAT MATRIX: ALLOWS THE USER TO EDIT THE CHARACTER-MATRIX ARGUMENT; THE RESULT IS THE REVISED MATRIX. EDITING IS AS IN THE <CH> FUNCTION IN WORKSPACE 4 CH AND CLOSELY RESEMBLES THE MAILBOX 'CH' EDITOR (FOR FULL DETAILS SEE <DESCRIBE> IN WS 4 CH). SEPARATION BETWEEN ROWS IS SHOWN BY A SPECIAL CHARACTER SELECTED BY EDITMAT; EDITMAT WILL SAY WHAT THIS CHARACTER IS BEFORE EDITING BEGINS.

TRAILING BLANKS IN EACH ROW OF THE ARGUMENT ARE THROWN AWAY BEFORE EDITING BEGINS. WHEN EDITING IS OVER, EACH ROW IS EXTENDED WITH AS MANY BLANKS AS REQUIRED TO HAVE ALL ROWS THE SAME LENGTH (SO THAT THE ROWS CAN BE PUT TOGETHER TO FORM A SINGLE MATRIX).

EDITMAT USES <CH> FROM WORKSPACE 4 CH; YOU MUST ')COPY 4 CH CH' BEFORE USING EDITMAT.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

- -

TITLE: ROHAN/EDITVEC.1 ROHAN/EDITVEC.1

TYPE: FUNCTION

SUMMARY: CHARACTER-VECTOR EDITOR LIKE THAT IN MAILBOX. HANDLES NEWLINES, BACKSPACES, ETC.

TIMESTAMP: 1983-10-20 22:37:25

CATEGORIES: 24 EDITING

DESCRIPTION:
REVISED←EDITVEC VECTOR: ALLOWS THE USER TO EDIT THE CHARACTER-VECTOR ARGUMENT; THE RESULT IS THE REVISED TEXT. EDITING IS AS IN THE <CH> FUNCTION IN WORKSPACE 4 CH AND CLOSELY RESEMBLES THE MAILBOX 'CH' EDITOR (FOR FULL DETAILS SEE <DESCRIBE> IN WS 4 CH). TERMINAL-CONTROL CHARACTERS ARE REPRESENTED BY ESCAPE SEQUENCES: NEWLINE BY '¨S', BACKSPACE BY '¨B', LINEFEED BY '¨L', IDLE BY '¨D', NULL BY '¨N', AND '¨' BY '¨¨' (ALL EXACTLY AS IN MAILBOX).

EDITVEC USES <CH> FROM WORKSPACE 4 CH; YOU MUST ')COPY 4 CH CH' BEFORE USING EDITVEC.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

TITLE: *ROHAN/FILES.2*

TYPE: *FUNCTION*

SUMMARY: *RETURNS A FORMATTED LIST OF THE FILES IN THE SPECIFIED LIBRARY.*

TIMESTAMP: 1983-02-03 16:15:16

CATEGORIES: 9 *FILES*
25 *FORMATTING*

CHANGES:
MODIFIED FROM VERSION 1 TO (A) BE ⎕IO-INDEPENDENT AND (B) TAKE AN OPTIONAL LEFT ARGUMENT OF THE DESIRED RESULT WIDTH.

DESCRIPTION:
Z←WIDTH FILES LIBRARY (<WIDTH> IS OPTIONAL):
RETURNS A FORMATTED LIST OF THE FILES IN THE SPECIFIED LIBRARY. THE LIBRARY NUMBER MUST BE AN INTEGER SCALAR. THE RESULT IS A MATRIX OF CHARACTERS, OF WIDTH <WIDTH> (THE DEFAULT WIDTH IS ⎕PW).

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

- -

TITLE: *ROHAN/FILESPACE.1*

TYPE: *FUNCTION*

SUMMARY: *RETURNS NAMES AND ⎕SIZES OF ALL FILES IN THE SPECIFIED LIBRARY, IN A CHARACTER MATRIX WITH TOTAL.*

TIMESTAMP: 1983-02-14 19:04:56

CATEGORIES: 9 *FILES*

DESCRIPTION:
RESULT←FILESPACE LIBRARY: RETURNS THE NAMES AND ⎕SIZES OF ALL FILES IN THE SPECIFIED LIBRARY, IN A CHARACTER MATRIX WITH TOTAL. USES ZEROS FOR ⎕SIZES OF FILES WHICH REQUIRE PASSNUMBER ACCESS.

SIDE-EFFECT: *UNTIES ALL TIED FILES.*

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

TITLE: ROHAN/FORMAT∆NL.1 ROHAN/FORMAT∆NL.1

TYPE: FUNCTION

SUMMARY: FORMATS A MATRIX NAMELIST JUST LIKE)FNS,)VARS, AND)GRPS DO. RESULT: A MATRIX OF SPECIFIED WIDTH.

TIMESTAMP: 1983-01-21 22:05:10

CATEGORIES: 25 FORMATTING

DESCRIPTION:
Z←WIDTH FORMAT∆NL NAMELIST (<WIDTH> IS OPTIONAL):
FORMATS A MATRIX NAMELIST JUST LIKE)FNS,)VARS, AND)GRPS DO. THE RESULT IS AN N BY <WIDTH> MATRIX; THE DEFAULT WIDTH IS ⎕PW.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

- -

TITLE: ROHAN/FORMAT∆TIMESTAMP.1 ROHAN/FORMAT∆TIMESTAMP.1

TYPE: FUNCTION

SUMMARY: PUTS THE GIVEN ⎕TS-STYLE TIMESTAMP INTO THE FORMAT <HH.MM.SS WWW DD MMM YYYY>.

TIMESTAMP: 1982-12-09 02:51:02

CATEGORIES: 37 TIMES AND DATES

DESCRIPTION:
RESULT←FORMAT∆TIMESTAMP TS: PUTS THE GIVEN ⎕TS-STYLE TIMESTAMP INTO THE FORMAT <HH.MM.SS WWW DD MMM YYYY> (THE SAME FORMAT USED BY THE SHARP APL MESSAGE PROCESSING FACILITY, 'MAILBOX'). EXAMPLE:

```
      FORMAT∆TIMESTAMP ⎕←⎕TS
1982 12 9 2 45 46 136
 2.45.46  THU  9 DEC 1982
```

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

TITLE: ROHAN/GETINPUT.1

TYPE: FUNCTION

SUMMARY: GETS A LINE OF CHAR INPUT, WITH AN EASY WAY TO TRY AGAIN IF THE INPUT IS LATER FOUND TO BE INVALID.

TIMESTAMP: 1983-01-07 22:05:40

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
21 TERMINAL INPUT

DESCRIPTION:
INPUT←GETINPUT PROMPT: GETS CHARACTER INPUT (PROMPTING WITH <PROMPT>). IF THE INPUT IS LATER FOUND TO BE INVALID, THE SUBUTILITY ROHAN/TRYAGAIN CAN PRINT AN ERROR MESSAGE AND RETURN CONTROL TO THE LINE WHICH CALLED <GETINPUT>. FOR EXAMPLE:

```
      ∇ FOO;INPUTSTACK;NUMS;STUFF
        . . .
  [7]   STUFF←GETINPUT 'ENTER WIDGET SERIAL NUMBERS: '
  [8]   →TRYAGAIN (~∧/⎕VI STUFF)/'NUMBERS ONLY'
  [9]   NUMS←⎕FI STUFF
  [10]  →TRYAGAIN (NUMS∨.≠⌊NUMS)/'NON-NEGATIVE INTEGERS ONLY'
        . . .
      ∇
```

BOTH FUNCTIONS USE THE GLOBAL VARIABLE <INPUTSTACK>, WHICH <GETINPUT> CREATES IF NECESSARY. YOU MAY WISH TO LOCALIZE <INPUTSTACK> AS IN THE EXAMPLE ABOVE.

BOTH FUNCTIONS OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. THEY WERE WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. THEY HAVE BEEN TESTED.

```
TITLE:       ROHAN/IF.1

TYPE:        FUNCTION

SUMMARY:     USAGE:  →IF CONDITION ⋄ ACTION∆IF∆CONDITION∆TRUE

TIMESTAMP:   1982-11-23 02:27:11

CATEGORIES:  26  EXECUTION CONTROL
             28  DEFINED FUNCTIONS  ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
USAGE IS:
      →IF CONDITION ⋄ ACTION∆IF∆CONDITION∆TRUE

FOR EXAMPLE:
    ∇ TEST
[1]   ⍝ THIS LINE IS ALWAYS EXECUTED
[2]   →IF (LTS ⎕TS)[4]<7 ⋄ 'INSOMNIA, EH?' ⋄ 'TRY COUNTING SHEEP.'
[3]   ⍝ THIS LINE IS ALWAYS EXECUTED
    ∇

NOTE THAT (AS IN LINE [2] ABOVE) THE ACTION TO BE TAKEN CAN OCCUPY MORE THAN
ONE STATEMENT.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO.  IT WAS WRITTEN BY
ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES.  IT HAS BEEN THOROUGHLY TESTED AND
HAS PROVEN RELIABLE IN ACTUAL USE.

THERE IS ALSO A COMPLEMENTARY UTILITY 'ROHAN/UNLESS', WHICH IS USED AS FOLLOWS:
      →UNLESS CONDITION ⋄ ACTION∆UNLESS∆CONDITION∆TRUE
```

- -


```
TITLE:       ROHAN/MAKE∆NL.1

TYPE:        FUNCTION

SUMMARY:     TAKES A CHARACTER-ARRAY ARGUMENT OF NAMES, OF ANY RANK, AND RETURNS A LEFT-JUSTIFIED MATRIX NAMELIST

TIMESTAMP:   1983-02-15 15:00:16

CATEGORIES:  2  STRUCTURAL TRANSFORMATIONS  ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.

DESCRIPTION:
RESULT←MAKE∆NL ARRAY:  TAKES A CHARACTER-ARRAY ARGUMENT OF NAMES, OF ANY RANK,
AND RETURNS A LEFT-JUSTIFIED MATRIX NAMELIST.  FOR EXAMPLE:

      MAKE∆NL '    FNSHOW   SESHOW BY   AND      '
FNSHOW
SESHOW
BY
AND

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO.  IT WAS WRITTEN BY
ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES.  IT HAS BEEN THOROUGHLY TESTED AND
HAS PROVEN RELIABLE IN ACTUAL USE.
```

TITLE: ROHAN/MAKE△STD△FILENAME.1

TYPE: FUNCTION

SUMMARY: PUTS A FILENAME INTO THE STANDARD 22-ELEMENT FORM RETURNED BY ⎕LIB AND ⎕NAMES.

TIMESTAMP: 1982-12-14 01:40:19

CATEGORIES: 9 FILES

DESCRIPTION:
RESULT←MAKE△STD△FILENAME A: PUTS THE GIVEN FILENAME (WHICH MAY OR MAY NOT CONTAIN A LIBRARY NUMBER) INTO THE STANDARD 22-ELEMENT FORM RETURNED BY ⎕LIB AND ⎕NAMES. THE ARGUMENT MUST BE A VECTOR (OR SCALAR) OF CHARACTERS; THE RESULT IS A 22-ELEMENT VECTOR OF CHARACTERS.

EXAMPLES:
```
      '→',(MAKE△STD△FILENAME '444 MAILSYSTEM'),'←'
→       444 MAILSYSTEM ←

      '→',(MAKE△STD△FILENAME '  444       MAILSYSTEM         '),'←'
→       444 MAILSYSTEM ←

      '→',(MAKE△STD△FILENAME 'MYFILE'),'←' ⍝ ASSUMING THAT 1↑⎕AI IS 1234567
→   1234567 MYFILE     ←
```

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

TITLE: ROHAN/NΔEXECUTE.2

TYPE: FUNCTION

SUMMARY: EXECUTES A GIVEN STATEMENT IN AN N-TASK.

TIMESTAMP: 1983-02-22 20:10:51

CATEGORIES: 18 N-TASKS

CHANGES:
CODE SLIGHTLY CLEANED UP AS SUGGESTED BY J. HENRI SCHUELER

DESCRIPTION:
RESULT←LIMITS NΔEXECUTE STATEMENT: THE STATEMENT (OR SEQUENCE OF STATEMENTS SEPARATED BY DIAMONDS) GIVEN AS THE RIGHT ARGUMENT IS EXECUTED BY AN N-TASK (USUALLY TO SAVE MONEY). THE RESULT OF THE FUNCTION IS THE RESULT OF THE ⎕RUN WHICH STARTED UP THE N-TASK, I.E. A 2-ELEMENT VECTOR. ANY RESULT OF THE EXECUTED STATEMENT IS LOST.

THE OPTIONAL LEFT ARGUMENT IS A 2-ELEMENT VECTOR GIVING CPU AND CONNECT LIMITS FOR THE N-TASK (IN UNITS AND SECONDS RESPECTIVELY); THE DEFAULT IS 1000 0 (I.E. MAXIMUM 1000 CPU UNITS AND NO LIMIT ON CONNECT TIME).

IF THE N-TASK CRASHES, ITS ACTIVE WS IS SAVED UNDER THE NAME 'NΔERROR'. OTHERWISE NO WORKSPACE IS SAVED.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. THE VALUES OF ALL SYSTEM VARIABLES (EXCEPT ⎕LX) GIVEN TO THE N-TASK ARE UNCHANGED FROM THOSE EXISTING WHEN <NΔEXECUTE> WAS INVOKED. ⎕LX IS LOCALLY SET TO AN EMPTY VECTOR, SO THAT IF THE N-TASK CRASHES YOU CAN SAFELY)LOAD NΔERROR .

THIS FUNCTION WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

TITLE: *ROHAN/PARTITION.1*

TYPE: *FUNCTION*

SUMMARY: *PARTITIONS <ARRAY> ALONG ITS LAST AXIS, INTO A VECTOR OF ENCLOSURES.*

TIMESTAMP: 1982-11-30 04:31:38

CATEGORIES: 3 *PARTITIONED ARRAY HANDLING*
4 *ENCLOSED ARRAYS*

DESCRIPTION:
Z←BOOLEAN PARTITION ARRAY: PARTITIONS <ARRAY> ALONG ITS LAST AXIS, INTO A VECTOR OF ENCLOSURES.

EACH 1 IN <BOOLEAN> DESIGNATES THE BEGINNING OF A PART OF <ARRAY>; <BOOLEAN> IS A VECTOR WITH ρBOOLEAN ↔ ¯1↑ρARRAY.

THE RESULT IS A VECTOR OF ENCLOSURES, WITH ONE PART OF <ARRAY> PER ENCLOSURE; ρRESULT ↔ +/BOOLEAN.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

EXAMPLE:

```
      ⎕←M←2 9ρ'ABCDEFGHI123456789'
ABCDEFGHI
123456789
      ρZ←1 0 0 1 0 1 0 0 0 PARTITION M
3
      >Z[1]
ABC
123
      >Z[2]
DE
45
      >Z[3]
FGHI
6789
      ρ⍤> Z
2 3
2 2
2 4
```

ROHAN/PARTITION∆VECTOR.1

TITLE: ROHAN/PARTITION∆VECTOR.1

TYPE: FUNCTION

SUMMARY: FASTER THAN ROHAN/PARTITION, BUT ONLY WORKS ON VECTORS.

TIMESTAMP: 1983-10-13 22:51:37

CATEGORIES: 3 PARTITIONED ARRAY HANDLING
4 ENCLOSED ARRAYS

DESCRIPTION:
THIS FUNCTION IS JUST LIKE ROHAN/PARTITION BUT COSTS LESS TO USE. HOWEVER, IT ONLY WORKS ON VECTORS.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

ROHAN/QSAVE.1

TITLE: ROHAN/QSAVE.1

TYPE: FUNCTION

SUMMARY: SAVES THE ACTIVE WORKSPACE UNDER A GIVEN NAME.

TIMESTAMP: 1982-11-21 23:23:15

CATEGORIES: 1 MISCELLANEOUS

DESCRIPTION:
SYNTAX: SIGNON QSAVE WSNAME.

SAVES THE ACTIVE WORKSPACE UNDER THE NAME GIVEN BY THE RIGHT ARGUMENT. THE OPTIONAL LEFT ARGUMENT SPECIFIES THE ACCOUNT NUMBER AND PASSWORD TO BE USED; THE DEFAULT IS THE CURRENT ACCOUNT AND PASSWORD. (SEE THE DEFINITION OF THE FUNCTION FOR DETAILS ABOUT THIS.) <QSAVE> AND ANY OTHER FUNCTIONS THROUGH WHICH IT WAS CALLED ARE CLEARED FROM THE)SI BEFORE SAVING.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

TITLE: ROHAN/SIMPLE.2

TYPE: FUNCTION

SUMMARY: DETERMINES WHETHER AN ARRAY IS SIMPLE.

TIMESTAMP: 1983-05-12 13:33:55

CATEGORIES: 4 ENCLOSED ARRAYS

CHANGES:
REWRITTEN TO USE THE NEW PRIMITIVE FUNCTION ⊃.

DESCRIPTION:
RESULT←SIMPLE ARRAY: DETERMINES WHETHER THE GIVEN ARRAY IS SIMPLE (I.E. DOES NOT CONTAIN ANY ENCLOSURES). NOTE THAT BY DEFINITION AN EMPTY ARRAY IS ALWAYS SIMPLE. FOR INSTANCE:

```
      SIMPLE 3 4⍴⍳12
1
      SIMPLE ⎕AV
1
      SIMPLE X←(<'ABCD'),<2 2⍴3.4
0
      SIMPLE 0⍴X
1
```

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

- -

TITLE: ROHAN/SIZES.2

TYPE: FUNCTION

SUMMARY: RETURNS A FORMATTED TABLE OF THE NAMES AND SIZES OF THE OBJECTS SPECIFIED IN THE ARGUMENT.

TIMESTAMP: 1983-12-21 16:36:32

CATEGORIES: 34 MEASURING TIME AND SPACE REQUIREMENTS

CHANGES:
NOW GIVES THE NAMES IN DECREASING ORDER OF SIZE.

DESCRIPTION:
Z←WIDTH SIZES NAMELIST (<WIDTH> IS OPTIONAL): THE RIGHT ARGUMENT IS A MATRIX NAMELIST OF OBJECTS IN THE ACTIVE WORKSPACE; THE RESULT IS A FORMATTED MATRIX OF THE OBJECTS' NAMES AND SIZES, IN DECREASING ORDER OF SIZE. THE OPTIONAL LEFT ARGUMENT SPECIFIES THE MAXIMUM WIDTH OF THE RESULT; THE DEFAULT VALUE IS ⎕PW.

YOU MAY FIND THIS FUNCTION VERY USEFUL WHEN YOU ARE FACED WITH 'WS FULL' PROBLEMS. IT WILL GIVE YOU SOME IDEA OF WHERE THE SPACE IN YOUR WS IS GOING.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

TITLE: *ROHAN/SUBTOTAL.1*

TYPE: *FUNCTION*

SUMMARY: *INSERTS FIRST-AXIS SUBTOTALS INTO AN ARRAY.*

TIMESTAMP: 1982-12-09 02:31:10

CATEGORIES:
3 *PARTITIONED ARRAY HANDLING*
5 *NUMERIC CALCULATION*
38 *BUSINESS AND FINANCE*

DESCRIPTION:
RESULT←BOOLEAN SUBTOTAL ARRAY: RETURNS <ARRAY> WITH FIRST-AXIS SUBTOTALS INSERTED. EACH 1 IN VECTOR <BOOLEAN> DESIGNATES THE BEGINNING OF A NEW SECTION FOR SUBTOTALLING.

EXAMPLE:

```
      1 0 1 1 0 0 0 SUBTOTAL 7 3 ⍴⍳21
1  2  3
4  5  6
5  7  9
7  8  9
7  8  9
10 11 12
13 14 15
16 17 18
19 20 21
58 62 66
```

⍴BOOLEAN ↔ 1↑⍴ARRAY; 1↑⍴RESULT ↔ (⍴BOOLEAN)++/BOOLEAN. NOTE THAT <BOOLEAN> MUST (UNLESS EMPTY) BEGIN WITH A 1.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

TITLE: ROHAN/SVHOLD.1

TYPE: FUNCTION

SUMMARY: HOLDS THE SPECIFIED SHARED VARIABLES, SOMEWHAT LIKE ⎕HOLD HOLDS FILES.

TIMESTAMP: 1983-10-20 00:06:08

CATEGORIES: 14 COMMUNICATION BETWEEN TASKS
15 SHARED VARIABLES

DESCRIPTION:
SYNTAX IS SVHOLD VARIABLES , WHERE <VARIABLES> IS A NAMELIST OF SHARED VARIABLES. THIS HOLDS THE SPECIFIED SHARED VARIABLES, LIKE ⎕HOLD DOES FOR FILES. USE ROHAN/SVRELEASE TO RELEASE THESE HOLDS; THE SYNTAX IS SVRELEASE VARIABLES .

NOTE WELL THAT VARIABLES HELD IN THIS MANNER SHOULD NOT BE USED TO TRANSFER DATA. THUS TO USE THIS SCHEME YOU SHOULD SHARE A NEW VARIABLE IN ADDITION TO THE ONES THAT YOU WOULD NORMALLY HAVE, AND USE SVHOLD AND SVRELEASE ONLY ON THIS EXTRA VARIABLE.

AS WITH ⎕HOLD, BOTH TASKS SHARING THE VARIABLE MUST USE SVHOLD AND SVRELEASE; ONE TASK IS NOT ENOUGH.

NOTE THAT THE ARGUMENT MUST BE AN ACCEPTABLE ARGUMENT TO ⎕SVC; A VECTOR OF NAMES SEPARATED BY SPACES WILL NOT WORK.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES, BUT THE IDEA IS THAT OF EUGENE MCDONNELL, ALSO OF I.P. SHARP. <SVHOLD> AND <SVRELEASE> HAVE BEEN THOROUGHLY TESTED AND HAVE PROVEN RELIABLE IN ACTUAL USE.

- -

TITLE: ROHAN/SVRELEASE.1

TYPE: FUNCTION

SUMMARY: TO BE USED WITH ROHAN/SVHOLD.

TIMESTAMP: 1983-10-20 00:06:34

CATEGORIES: 14 COMMUNICATION BETWEEN TASKS
15 SHARED VARIABLES

DESCRIPTION:
THIS FUNCTION IS INTENDED FOR USE WITH ROHAN/SVHOLD. SEE THE DESCRIPTION OF ROHAN/SVHOLD FOR DETAILS.

TITLE: ROHAN/TABSET.2

TYPE: FUNCTION

SUMMARY: SETS TABS, GIVEN THE TYPE OF TERMINAL AND THE DESIRED TAB SETTINGS.

TIMESTAMP: 1983-11-22 15:35:39

CATEGORIES: 20 TERMINAL INPUT/OUTPUT
22 TERMINAL CONTROL
23 TABS

CHANGES:
ERROR CHECKING AND REPORTING SLIGHTLY IMPROVED.

DESCRIPTION:
TERMTYPE TABSET HT: SETS ⎕HT TO <HT> AND SETS THE CORRESPONDING TABSTOPS ON THE TERMINAL, FOR TERMINAL TYPE <TERMTYPE>. EXAMPLE: 'AJ832' TABSET 5 .

TO SEE A LIST OF THE AVAILABLE TERMINAL TYPES, TYPE '' TABSET X , WHERE X CAN BE ANYTHING.

<HT> CAN BE ANY LEGAL VALUE FOR ⎕HT, I.E. IT CAN BE EITHER A SCALAR OR A VECTOR. IF IT IS A VECTOR, THE TERMINAL'S TABSTOPS ARE SET AT EXACTLY THOSE POSITIONS. IF IT IS A SCALAR, THE TABSTOPS WILL BE SET AS FAR AS ⎕PW, E.G. 'HDS108' TABSET 10 WITH ⎕PW=80 WILL SET TABSTOPS AT 10, 20, 30, 40, 50, 60, 70, AND 80.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

- -

TITLE: ROHAN/TIED.1

TYPE: FUNCTION

SUMMARY: SHOWS WHICH FILES ARE TIED TO WHAT NUMBERS, IN A READABLE FORMAT.

TIMESTAMP: 1983-02-15 15:41:05

CATEGORIES: 9 FILES
25 FORMATTING

DESCRIPTION:
SHOWS WHICH FILES ARE TIED TO WHAT NUMBERS, IN A READABLE FORMAT. FOR EXAMPLE:

```
      TIED
          PRIVATE      1
        86 MABRAV3     100
           MSK99       101
   1854339 STUFF       12
      ⍴TIED
4 33
```

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

TITLE: ROHAN/UNLESS.1

TYPE: FUNCTION

SUMMARY: USAGE: →UNLESS CONDITION ◇ ACTION∆UNLESS∆CONDITION∆TRUE

TIMESTAMP: 1983-02-14 18:06:16

CATEGORIES: 26 EXECUTION CONTROL
28 DEFINED FUNCTIONS ⍝ UTILITIES DEALING WITH DEFINED FUNCTIONS

DESCRIPTION:
USAGE IS:

```
      →UNLESS CONDITION ◇ ACTION∆UNLESS∆CONDITION∆TRUE
```

FOR EXAMPLE:

```
    ∇ TEST
[1]   ⍝ THIS LINE IS ALWAYS EXECUTED
[2]   →UNLESS ⎕WA≥5000 ◇ 'INSUFFICIENT WORKING SPACE' ◇ →0
[3]   ⍝ THIS LINE IS EXECUTED IF AND ONLY IF ⎕WA≥5000
    ∇
```

NOTE THAT (AS IN LINE [2] ABOVE) THE ACTION TO BE TAKEN CAN OCCUPY MORE THAN ONE STATEMENT.

THIS FUNCTION OPERATES CORRECTLY WITH EITHER VALUE OF ⎕IO. IT WAS WRITTEN BY ROHAN JAYASEKERA OF I.P. SHARP ASSOCIATES. IT HAS BEEN THOROUGHLY TESTED AND HAS PROVEN RELIABLE IN ACTUAL USE.

THERE IS ALSO A COMPLEMENTARY UTILITY 'ROHAN/IF', WHICH IS USED AS FOLLOWS:

```
      →IF CONDITION ◇ ACTION∆IF∆CONDITION∆TRUE
```

- -

SAUCE/ALLOCEQ.1

TITLE: SAUCE/ALLOCEQ.1

TYPE: FUNCTION

SUMMARY: ALLOCATES A NUMBER EQUITABLY SUBJECT TO LIMITS. E.G. 29 ALLOCEQ 3 20 500 7 ↔ 3 9.5 9.5 7.

TIMESTAMP: 1982-11-20 07:04:33

CATEGORIES: 5 NUMERIC CALCULATION
38 BUSINESS AND FINANCE

DESCRIPTION:
RESULT←TOTAL ALLOCEQ LIMITS

<TOTAL> IS A SINGLETON AND <LIMITS> IS A VECTOR. THE FUNCTION ALLOCATES <TOTAL> EQUITABLY SUBJECT TO <LIMITS>.

E.G. 29 ALLOCEQ 3 20 500 7 ↔ 3 9.5 9.5 7.

<ALLOCEQ> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/ALLOCFIFO.1

TYPE: FUNCTION

SUMMARY: ALLOCATES A NUMBER ON A FIFO BASIS, SUBJECT TO LIMITS. E.G. 9 ALLOCFIFO 4 6 11 ↔ 4 5 0.

TIMESTAMP: 1982-11-22 08:28:05

CATEGORIES:
5 NUMERIC CALCULATION
38 BUSINESS AND FINANCE

DESCRIPTION:
RESULT←TOTAL ALLOCFIFO LIMITS

<TOTAL> IS A SINGLETON AND <LIMITS> IS A VECTOR. THE FUNCTION ALLOCATES
<TOTAL> ON A FIRST-IN FIRST-OUT BASIS SUBJECT TO <LIMITS>.

E.G. 9 ALLOCFIFO 4 6 11 ↔ 4 5 0.

<ALLOCFIFO> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/CATENATEROWS.1

TYPE: FUNCTION

SUMMARY: CATENATES ONE VECTOR OR MATRIX UNDER ANOTHER, USING OVERTAKE AS NECESSARY.

TIMESTAMP: 1982-11-13 02:48:17

CATEGORIES: 2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.

DESCRIPTION:
MATRIX←TOP CATENATEROWS BOTTOM

CATENATES THE <TOP> ROW(S) AND THE BOTTOM ROW(S), WHICH MUST BOTH BE OF RANK
AT MOST 2.

EXAMPLES:

```
      ⍴⎕←'HI' CATENATEROWS 'THERE'
HI
THERE
2 5
      ⍴⎕←(2 3⍴'ABCDEF') CATENATEROWS 'GH'
ABC
DEF
GH
3 3
      ⍴⎕←(2 3⍴'ABCDEF') CATENATEROWS 3 4⍴'WXYZ'
ABC
DEF
WXYZ
WXYZ
WXYZ
5 4

      (2 1 ⍴ 4 3)≡4 CATENATEROWS 3
1
```

<CATENATEROWS> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/CYCLICMESH.1

TYPE: FUNCTION

SUMMARY: E.G. 4 CYCLICMESH 2 12⍴'AEI','BFJ','CGK','DHL' ↔ 2 12⍴'ABCDEFGHIJKL'.

TIMESTAMP: 1983-10-03 20:01:01

CATEGORIES: 2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.

DESCRIPTION:
RESULT←N CYCLICMESH ARRAY

E.G. 4 CYCLICMESH 2 12⍴'AEI','BFJ','CGK','DHL' ↔ 2 12⍴'ABCDEFGHIJKL'

<ARRAY> MAY BE OF ARBITRARY RANK; IT MAY CONTAIN NUMBERS, CHARACTERS, OR ENCLOSURES.

<CYCLICMESH> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

SAUCE/DESCRIBE.1

TITLE: SAUCE/DESCRIBE.1

TYPE: ARRAY

SUMMARY: AN OVERVIEW OF THE FUNCTIONS IN SAUCE/*.

TIMESTAMP: 1984-10-27 19:41:45

CATEGORIES: 1 MISCELLANEOUS

DESCRIPTION:
SAUCE/* (SHARP APL UTILITIES FOR CODING EASE) HOLDS A COLLECTION OF APL UTILITY FUNCTIONS--SMALL, GENERAL-PURPOSE FUNCTIONS THAT CAN BE USED AS COMPONENTS IN BUILDING AN APPLICATION SYSTEM. EACH PERFORMS A CONCEPTUALLY SIMPLE OPERATION (FOR INSTANCE, REMOVING EXTRA BLANKS FROM A VECTOR OF TEXT, OR PERFORMING A GRADE-UP ALONG ANY GIVEN AXIS OF AN ARRAY), AND IS SIMILAR TO AN APL PRIMITIVE FUNCTION IN THAT IT TAKES ONE OR TWO ARGUMENTS AND RETURNS A RESULT DEPENDENT ONLY ON THE ARGUMENT(S) AND PERHAPS ONE OR MORE SYSTEM VARIABLES. SAUCE FUNCTIONS, LIKE ALMOST ALL PRIMITIVE FUNCTIONS, ARE USEFUL IN A WIDE VARIETY OF APPLICATIONS, AND SAUCE CAN BE VIEWED AS EXTENDING THE REPERTOIRE OF PRIMITIVES AVAILABLE TO THE PROGRAMMER.

THE DEFINITION OF THIS UTILITY CONTAINS A DETAILED DESCRIPTION OF THE PRINCIPLES UNDERLYING SAUCE.

TITLE: SAUCE/DISTINCT.1

TYPE: FUNCTION

SUMMARY: REMOVES DUPLICATE ELEMENTS FROM A VECTOR. E.G. DISTINCT 3 5 3 4 ↔ 3 5 4.

TIMESTAMP: 1982-11-09 03:36:13

CATEGORIES: 40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
COMPRESSED←DISTINCT VECTOR

REMOVES DUPLICATE ELEMENTS FROM <VECTOR>. <VECTOR> MUST HAVE RANK ≤1; IT MAY CONTAIN CHARACTERS, NUMBERS, OR ENCLOSURES.

EXAMPLES:

```
      DISTINCT 3 5 3 4
3 5 4
      DISTINCT 'ABBBCBBD'
'ABCD'
```

<DISTINCT> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

REVIEWS:
[1982-11-12 01:01:26]

THE FUNCTION WORKS ON RANK>1 ARRAYS BY RAVELLING THEM.
IT IS SENSITIVE TO ⎕CT, AND SHOULD BE USED WITH CAUTION
ON LONG-INTEGER, REAL, OR COMPLEX NUMERIC ARRAYS.
THE ELEMENTS IN THE RESULT APPEAR IN THE SAME ORDER AS
THEY DID IN THE ARGUMENT.

J. HENRI SCHUELER

TITLE: SAUCE/DISTINCTROWS.1 SAUCE/DISTINCTROWS.1

TYPE: FUNCTION

SUMMARY: REMOVES DUPLICATE ROWS FROM MATRIX, E.G. DISTINCTROWS 4 2ρ'AB','CD','AB','AD' ↔ 3 2ρ'AB','CD','AD'.

TIMESTAMP: 1982-11-16 22:08:25

CATEGORIES: 40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
COMPRESSED←DISTINCTROWS MATRIX

REMOVES DUPLICATE ROWS FROM <MATRIX>. <MATRIX> MAY CONTAIN CHARACTERS OR NUMBERS; IT SHOULD NOT CONTAIN ENCLOSURES.

EXAMPLE

```
      DISTINCTROWS 6 2 ρ'AB','AB','CD','AB','CD','EF'
AB
CD
EF
```

<DISTINCTROWS> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: SAUCE/DISTROUND.1 SAUCE/DISTROUND.1

TYPE: FUNCTION

SUMMARY: DISTRIBUTIVE ROUNDING. E.G. 0.01 DISTROUND 2 6 ρ÷ 1 3 ↔ 2 6 ρ 1 0.34 1 0.33 1 0.33.

TIMESTAMP: 1982-12-01 07:54:07

CATEGORIES: 5 NUMERIC CALCULATION
38 BUSINESS AND FINANCE
39 STATISTICS AND PROBABILITY

DESCRIPTION:
ROUNDED←UNIT DISTROUND ARRAY

PERFORMS ROUNDING ENSURING THAT +/ARRAY ↔ +/ROUNDED, WHEN BOTH ARE EXPRESSED IN THE SAME <UNIT>S. SUBJECT TO THIS, +/|ARRAY-ROUNDED IS MINIMIZED.

E.G. 0.01 DISTROUND 2 6 ρ÷ 1 3 ↔ 2 6 ρ 1 0.34 1 0.33 1 0.33.

E.G. 0.01 DISTROUND 9ρ÷3 ↔ 0.34 0.34 0.34 0.33 0.33 0.33 0.33 0.33 0.33.

E.G. 0.01 DISTROUND 2 2 ρ÷3 ↔ 2 2 ρ 0.34 0.33 0.34 0.33.

<DISTROUND> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/ELEMREPLACE.1 SAUCE/ELEMREPLACE.1

TYPE: FUNCTION

SUMMARY: REPLACES, IN A VECTOR CONTAINING NO ENCLOSURES, ALL OCCURRENCES OF A SCALAR BY A (DIFFERENT) VECTOR.

TIMESTAMP: 1983-02-18 06:03:34

CATEGORIES: 24 EDITING
27 TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS
46 MODIFYING ARRAYS ⍝ INDEXED ASSIGNMENT, SUBSTRING REPLACEMENT, ETC.

DESCRIPTION:
RESULT←VECTOR ELEMREPLACE ELEMSTRING

REPLACES, IN <VECTOR>, ALL OCCURRENCES OF 1↑ELEMSTRING BY 1↓ELEMSTRING.
<VECTOR> MAY BE CHARACTER OR NUMERIC; IT SHOULD NOT CONTAIN ENCLOSURES.

E.G. 'ABAACDA' ELEMREPLACE 'A','ZAP' ↔ 'ZAPBZAPZAPCDZAP'.

E.G. 'ABAACDA' ELEMREPLACE 'A' ↔ 'BCD'.

E.G. 'ABAACDA' ELEMREPLACE '' ↔ 'ABAACDA'.

E.G. ' B CD ' ELEMREPLACE '' ↔ 'BCD'.

<ELEMREPLACE> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: SAUCE/EXTENDPARTS.1 SAUCE/EXTENDPARTS.1

TYPE: FUNCTION

SUMMARY: FOR EXTENDING PARTS. E.G. 1 1 0 0 1 0 0 EXTENDPARTS 1 0 0 0 1 1 0 ↔ 1 0 0 0 1 0 0 1 0 0 0.

TIMESTAMP: 1983-04-03 07:29:02

CATEGORIES: 3 PARTITIONED ARRAY HANDLING
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.
41 BOOLEAN ARRAYS

DESCRIPTION:
MASK←SELECTOR EXTENDPARTS BOOLEAN

EACH 1 IN <BOOLEAN> DESIGNATES THE BEGINNING OF A PART. THE RESULT IS AN
EXPANSION MASK THAT CAN BE USED TO EXTEND PARTS. <SELECTOR> DETERMINES HOW
MUCH EACH PART IS TO BE EXPANDED.

E.G. 1 1 0 0 1 0 0 EXTENDPARTS 1 0 0 0 1 1 0 ↔ 1 0 0 0 1 0 0 1 0 0 0.

E.G. 1 1 0 0 1 0 0 EXTENDPARTS 1 1 0 1 ↔ 1 1 0 0 0 1 0 0.

<EXTENDPARTS> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/FIRSTONES.1

TYPE: FUNCTION

SUMMARY: SETS TO 0 ALL BUT THE FIRST 1 IN EACH SEQUENCE OF 1'S IN A BOOLEAN VECTOR.

TIMESTAMP: 1982-12-08 08:48:18

CATEGORIES: 40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.
41 BOOLEAN ARRAYS

DESCRIPTION:
RESULT←FIRSTONES BOOLEAN

<BOOLEAN> MUST BE VECTOR OR SCALAR. ⍴RESULT ↔ ⍴,BOOLEAN, WITH ALL BUT THE FIRST 1 IN EACH SEQUENCEOF 1'S SET TO 0.

E.G. FIRSTONES 1 0 1 1 1 0 1 1 0 0 1 1 ↔ 1 0 1 0 0 0 1 0 0 0 1 0.

<FIRSTONES> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/FIRSTZEROES.1

TYPE: FUNCTION

SUMMARY: SETS TO 1 ALL BUT THE FIRST 0 IN EACH SEQUENCE OF 0'S IN A BOOLEAN VECTOR.

TIMESTAMP: 1982-12-08 08:40:40

CATEGORIES: 40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.
41 BOOLEAN ARRAYS

DESCRIPTION:
RESULT←FIRSTZEROES BOOLEAN

<BOOLEAN> MUST BE VECTOR OR SCALAR. ⍴RESULT ↔ ⍴,BOOLEAN, WITH ALL BUT THE FIRST 0 IN EACH SEQUENCE OF 0'S SET TO 1.

E.G. FIRSTZEROES 0 1 0 0 0 1 0 0 1 1 0 0 ↔ 0 1 0 1 1 1 0 1 1 1 0 1.

<FIRSTZEROES> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/FREQDIST.1 SAUCE/FREQDIST.1

TYPE: FUNCTION

SUMMARY: RETURNS THE FREQUENCY DISTRIBUTION, IN SPECIFIED CLASSES, OF NUMERIC DATA.

TIMESTAMP: 1985-01-02 00:16:47

CATEGORIES: 6 SORTING AND GRADING
39 STATISTICS AND PROBABILITY

DESCRIPTION:
DISTRIBUTION←CLASSES FREQDIST DATA

RETURNS THE FREQUENCY DISTRIBUTION OF <DATA> IN THE SPECIFIED <CLASSES>.
<CLASSES> SHOULD BE A VECTOR OF LOWER CLASS LIMITS. IT IS ASSUMED THAT
<CLASSES> IS IN ASCENDING ORDER.

EXAMPLES

```
      0 10 100 1000 FREQDIST 6 90002 8 17 71
2 2 0 1
      ¯1.5 500 7500 FREQDIST ¯3.4 500 3500 3500
0 3 0
```

<FREQDIST> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER
VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: SAUCE/FROM.1 SAUCE/FROM.1

TYPE: FUNCTION

SUMMARY: A VARIANT OF INDEXING. EACH ROW OF THE LEFT ARGUMENT SELECTS ONE ELEMENT FROM THE RIGHT ARGUMENT.

TIMESTAMP: 1983-02-18 06:13:19

CATEGORIES: 40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
ELEMENTS←INDICES FROM ARRAY

A VARIANT OF INDEXING. EACH ROW OF <INDICES> SELECTS ONE ELEMENT FROM <ARRAY>.
⍴INDICES ↔ (⍴ELEMENTS),⍴⍴ARRAY.

E.G. [⎕IO←1] (3 2 ⍴ 1 4 , 2 1 , 1 3) FROM 2 4 ⍴'ABCD','EFGH' ↔ 'DEC'.

E.G. ⍴(? 3 2 7 5 ⍴6) FROM 10 9 8 7 6 ⍴2 ↔ 3 2 7.

<FROM> HAS BEEN THOROUGHLY TESTED. THE LEFT ARGUMENT IS INTERPRETED
APPROPRIATELY ACCORDING TO THE VALUE OF ⎕IO. IF THE FUNCTION BECOMES
SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: *SAUCE/FROMTO.1*

TYPE: *FUNCTION*

SUMMARY: *E.G.* 1 157 ¯12 *FROMTO* 2 161 ¯10 ↔ 1 2 157 158 159 160 161 ¯12 ¯11 ¯10.

TIMESTAMP: 1982-12-08 08:45:42

CATEGORIES: 40 *SELECTING FROM ARRAYS* ⍝ *INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.*

DESCRIPTION:
RESULT←ORIGINS FROMTO ENDS

E.G. 1 157 ¯12 *FROMTO* 2 161 ¯10 ↔ 1 2 157 158 159 160 161 ¯12 ¯11 ¯10.

<FROMTO> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: *SAUCE/GEROWS.1*

TYPE: *FUNCTION*

SUMMARY: *DETERMINES WHICH ROWS OF A MATRIX ARE ≥ (IN THE SENSE OF ⍋) A VECTOR.*

TIMESTAMP: 1984-09-20 02:26:23

CATEGORIES:
6 *SORTING AND GRADING*
7 *SEARCHING* ⍝ *INCLUDING MEMBERSHIP AND INDEX-OF*
40 *SELECTING FROM ARRAYS* ⍝ *INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.*

DESCRIPTION:
BOOLEAN←MATRIX GEROWS VECTOR

THE RESULT IS A BOOLEAN VECTOR OF LENGTH 1↑⍴MATRIX. BOOLEAN[I]=1 IF AND ONLY IF MATRIX[I;] IS GREATER THAN OR EQUAL TO VECTOR (IN THE SENSE OF ⍋).

EXAMPLES

```
      MAT
2 7 5
5 2 1
2 7 9
4 5 8
      MAT GEROWS 2 7 7
0 1 1 1
      MAT GEROWS 2 8 ¯1.5
0 1 0 1
      MAT GEROWS 2 7 9
0 1 1 1
```

<GEROWS> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/GRADEDOWN.1

TYPE: FUNCTION

SUMMARY: APPLIES ⍒ OVER THE DESIGNATED AXIS OF AN ARRAY.

TIMESTAMP: 1983-10-03 19:57:36

CATEGORIES: 6 SORTING AND GRADING

DESCRIPTION:
INDICES←[AXIS] GRADEDOWN ARRAY

⍒ IS APPLIED OVER THE DESIGNATED <AXIS> OF <ARRAY>. THE DEFAULT <AXIS> IS THE LAST AXIS OF <ARRAY>.

```
E.G.      ⎕IO←1
          GRADEDOWN 1 3 3 ⍴ 1 3 2 , 6 5 4 , 8 9 7
    2 3 1
    1 2 3
    2 1 3
```

<GRADEDOWN> HAS BEEN THOROUGHLY TESTED. THE RESULT VARIES APPROPRIATELY ACCORDING TO THE VALUE OF ⎕IO. IF THE FUNCTION BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: SAUCE/GRADEUP.1

TYPE: FUNCTION

SUMMARY: APPLIES ⍋ OVER THE DESIGNATED AXIS OF AN ARRAY.

TIMESTAMP: 1983-10-03 19:54:44

CATEGORIES: 6 SORTING AND GRADING

DESCRIPTION:
INDICES←[AXIS] GRADEUP ARRAY

⍋ IS APPLIED OVER THE DESIGNATED <AXIS> OF <ARRAY>. THE DEFAULT <AXIS> IS THE LAST AXIS OF <ARRAY>.

```
E.G.      ⎕IO←1
          GRADEUP 1 3 3 ⍴ 1 3 2 , 6 5 4 , 8 9 7
    1 3 2
    3 2 1
    3 1 2
```

<GRADEUP> HAS BEEN THOROUGHLY TESTED. THE RESULT VARIES APPROPRIATELY ACCORDING TO THE VALUE OF ⎕IO. IF THE FUNCTION BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/GTROWS.1

TYPE: FUNCTION

SUMMARY: DETERMINES WHICH ROWS OF A MATRIX ARE > (IN THE SENSE OF ⍋) A VECTOR.

TIMESTAMP: 1984-09-20 02:26:43

CATEGORIES: 6 SORTING AND GRADING
7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
BOOLEAN←MATRIX GTROWS VECTOR

THE RESULT IS A BOOLEAN VECTOR OF LENGTH 1↑⍴MATRIX. BOOLEAN[I]=1 IF AND ONLY IF MATRIX[I;] IS GREATER THAN VECTOR (IN THE SENSE OF ⍋).

EXAMPLES

```
      MAT
2 7 5
5 2 1
2 7 9
4 5 8
      MAT GTROWS 2 7 7
0 1 1 1
      MAT GTROWS 2 8 ¯1.5
0 1 0 1
      MAT GTROWS 2 7 9
0 1 0 1
```

<GTROWS> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: SAUCE/INDEXGEN.1

TYPE: FUNCTION

SUMMARY: THE ARG SHOULD BE A VECTOR. RETURNS (⍳1↑VECTOR),(⍳1↑1↓VECTOR),(⍳1↑2↓VECTOR),...,⍳¯1↑VECTOR.

TIMESTAMP: 1982-12-13 10:55:56

CATEGORIES: 40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
RESULT←INDEXGEN VECTOR

RESULT←(⍳1↑VECTOR),(⍳1↑1↓VECTOR),(1↑2↓VECTOR),...,⍳¯1↑VECTOR.

<INDEXGEN> HAS BEEN THOROUGHLY TESTED. THE RESULT VARIES APPROPRIATELY ACCORDING TO THE VALUE OF ⎕IO. IF THE FUNCTION BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/INDEXOFROWS.2 SAUCE/INDEXOFROWS.2

TYPE: FUNCTION

SUMMARY: RETURNS THE 'INDEX OF' EACH ROW OF ONE MATRIX IN ANOTHER MATRIX.

TIMESTAMP: 1983-10-18 21:21:49

CATEGORIES: 7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

CHANGES:
CODE SLIGHTLY CLEANER, AS SUGGESTED BY J. HENRI SCHUELER.

DESCRIPTION:
INDICES←MATRIX INDEXOFROWS ROWS

<ROWS> MAY BE A VECTOR OR MATRIX OF ANY SHAPE. <INDICES> CONTAINS THE 'INDEX OF' EACH ROW OF <ROWS> IN <MATRIX>. ρINDICES ↔ ¯1↓ρROWS.

EXAMPLES:

```
      ⎕IO←1
      (4 2 ρ'AB','CD','EF','GH') INDEXOFROWS 3 2 ρ'EF','AB','YZ'
3 1 5
      ⎕IO←0
      (4 2 ρ'AB','CD','EF','GH') INDEXOFROWS 'CD'
1
```

BOTH <ROWS> AND <MATRIX> MAY CONTAIN CHARACTERS, NUMBERS, OR ENCLOSURES. IF EITHER <ROWS> OR <MATRIX> CONTAINS ENCLOSURES, (¯1↑ρMATRIX) ↔ ¯1↑1,ρROWS.

<INDEXOFROWS> HAS BEEN THOROUGHLY TESTED. THE RESULT VARIES APPROPRIATELY ACCORDING TO THE VALUE OF ⎕IO. IF THE FUNCTION BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: SAUCE/LASTONES.1 SAUCE/LASTONES.1

TYPE: FUNCTION

SUMMARY: SETS TO 0 ALL BUT THE LAST 1 IN EACH SEQUENCE OF 1'S IN A BOOLEAN VECTOR.

TIMESTAMP: 1982-12-08 08:50:06

CATEGORIES: 40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.
41 BOOLEAN ARRAYS

DESCRIPTION:
RESULT←LASTONES BOOLEAN

<BOOLEAN> MUST BE VECTOR OR SCALAR. ρRESULT ↔ ρ,BOOLEAN, WITH ALL BUT THE LAST 1 IN EACH SEQUENCE OF 1'S SET TO 0.

E.G. LASTONES 1 0 1 1 1 0 1 1 0 0 1 1 ↔ 1 0 0 0 1 0 0 1 0 0 0 1.

<LASTONES> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/LASTZEROES.1

TYPE: FUNCTION

SUMMARY: SETS TO 1 ALL BUT THE LAST 0 IN EACH SEQUENCE OF 0'S IN A BOOLEAN VECTOR.

TIMESTAMP: 1982-12-08 08:52:04

CATEGORIES: 40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.
41 BOOLEAN ARRAYS

DESCRIPTION:
RESULT←LASTZEROES BOOLEAN

<BOOLEAN> MUST BE VECTOR OR SCALAR. ⍴RESULT ←→ ⍴,BOOLEAN, WITH ALL BUT THE LAST 0 IN EACH SEQUENCE OF 0'S SET TO 1.

```
E.G. LASTZEROES 0 1 0 0 0 1 0 0 1 1 0 0 ←→ 0 1 1 1 0 1 1 0 1 1 1 0.
```

<LASTZEROES> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: SAUCE/LEFTJUSTIFY.1

TYPE: FUNCTION

SUMMARY: LEFT JUSTIFIES EACH ROW OF AN ARRAY.

TIMESTAMP: 1982-11-12 23:15:04

CATEGORIES: 2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.
25 FORMATTING
30 REPORT FORMATTING

DESCRIPTION:
RESULT←[ELEMENT] LEFTJUSTIFY ARRAY

THE DEFAULT <ELEMENT> IS 1↑0⍴ARRAY (I.E. 0 OR ''). THE FUNCTION LEFT JUSTIFIES EACH ROW OF <ARRAY> WITH RESPECT TO <ELEMENT>. NOTE: IT WILL WORK EVEN IF <ELEMENT> HAS MORE THAT ONE DISTINCT ELEMENT; IN THIS CASE, LEFT JUSTIFICATION IS DONE WITH RESPECT TO ALL THE ELEMENTS OF <ELEMENT>.

```
E.G.  LEFTJUSTIFY '  JOE    ←→  'JOE
                  MARY          MARY
                   JOHN         JOHN
                     ED'        ED  '

E.G.  ' ∘' LEFTJUSTIFY '∘∘∘JOE   ←→   'JOE∘∘∘
                        MARY∘∘         MARY∘∘
                         JOHN∘         JOHN∘
                        ∘  ED∘'        ED∘∘  '
```

<ARRAY> MAY CONTAIN CHARACTERS, NUMBERS, OR ENCLOSURES.

<LEFTJUSTIFY> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/MASKPARTS.1 SAUCE/MASKPARTS.1

TYPE: FUNCTION

SUMMARY: E.G. 1 0 1 0 MASKPARTS 1 0 0 1 0 1 0 1 0 ↔ 1 1 1 0 0 1 1 0 0.

TIMESTAMP: 1982-12-05 06:46:12

CATEGORIES:
3 PARTITIONED ARRAY HANDLING
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.
41 BOOLEAN ARRAYS

DESCRIPTION:
RESULT←MASK MASKPARTS BOOLEAN

EACH 1 IN <BOOLEAN> DESIGNATES THE BEGINNING OF A PART. THE FUNCTION MASKS EACH PART ACCORDING TO THE BOOLEAN <MASK>.

E.G. 1 0 1 0 MASKPARTS 1 0 0 0 1 0 1 0 1 0 ↔ 1 1 1 0 0 1 1 0 0.

<MASKPARTS> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: SAUCE/MEMBERROWS.1 SAUCE/MEMBERROWS.1

TYPE: FUNCTION

SUMMARY: RETURNS A BOOLEAN ARRAY INDICATING WHETHER EACH ROW OF ONE ARG IS A ROW OF THE OTHER ARG.

TIMESTAMP: 1983-03-02 02:19:18

CATEGORIES:
7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.
41 BOOLEAN ARRAYS

DESCRIPTION:
BOOLEAN←ROWS MEMBERROWS MATRIX

<ROWS> MAY BE AN ARRAY OF ANY SHAPE. THE RESULT IS A BOOLEAN ARRAY OF SHAPE ¯1↓⍴ROWS INDICATING WHICH ROWS OF <ROWS> ARE ROWS OF <MATRIX>. NEITHER <ROWS> NOR <MATRIX> MAY CONTAIN ENCLOSURES.

EXAMPLE
```
      'AB  MEMBERROWS 'XYZ   ↔  1 0
       XY'             AB
                       CDE'
```

<MEMBERROWS> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/MESH.1 SAUCE/MESH.1

TYPE: FUNCTION

SUMMARY: MESHES CATENATED ARRAYS. E.G. 1 2 3 3 2 1 MESH 2 6 ρ'AD','BE','CF' ↔ 2 6 ρ'ABCDEF'.

TIMESTAMP: 1983-05-11 05:26:23

CATEGORIES: 2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
RESULT←INDICES MESH ARRAY

MESHES CATENATED ARRAYS. E.G. 1 2 3 3 2 1 MESH 2 6 ρ'AD','BE','CF' ↔
2 6 ρ'ABCFED'. IT IS ASSUMED THAT ¯1↑1,ρARRAY ↔ ρ,INDICES.

E.G. 1 2 2 1 1 2 MESH 'ADB','ECF' ↔ 'AECDBF'.

E.G. 1 3 1 3 2 1 3 3 MESH 'ABC','M','WXYZ' ↔ 'AWBXMCYZ'.

<ARRAY> MAY BE CONTAIN CHARACTERS, NUMBERS, OR ENCLOSURES.

<MESH> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: SAUCE/NOTMEMBERROWS.1 SAUCE/NOTMEMBERROWS.1

TYPE: FUNCTION

SUMMARY: RETURNS A BOOLEAN ARRAY INDICATING WHETHER EACH ROW OF ONE ARG IS A NOT A ROW OF THE OTHER ARG.

TIMESTAMP: 1983-03-02 02:56:01

CATEGORIES: 7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.
41 BOOLEAN ARRAYS

DESCRIPTION:
BOOLEAN←ROWS MEMBERROWS MATRIX

<ROWS> MAY BE AN ARRAY OF ANY SHAPE. THE RESULT IS A BOOLEAN ARRAY OF SHAPE
¯1↓ρROWS INDICATING WHICH ROWS OF <ROWS> ARE NOT ROWS OF <MATRIX>. NEITHER
<ROWS> NOR <MATRIX> MAY CONTAIN ENCLOSURES.

EXAMPLE

```
      'AB  NOTMEMBERROWS 'XYZ   ↔  0 1
       XY'                AB
                          CDE'
```

<NOTMEMBERROWS> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/PANDREDUCE.1

TYPE: FUNCTION

SUMMARY: PARTITIONED ∧/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

TIMESTAMP: 1982-11-16 22:40:54

CATEGORIES:
3 PARTITIONED ARRAY HANDLING
41 BOOLEAN ARRAYS

DESCRIPTION:
RESULT←BOOLEAN PANDREDUCE ARRAY

PARTITIONED ∧/ARRAY. <BOOLEAN> IS A VECTOR WITH ⍴BOOLEAN ↔ ¯1↑⍴ARRAY
(ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES
THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES

```
      1 0 0 0 1 0 1 0 PANDREDUCE 1 1 0 1 1 1 0 0
  ↔   (∧/ 1 1 0 1),(∧/ 1 1),(∧/ 0 0)

      1 0 1 0 0 0 PANDREDUCE 2 6 ⍴0,11⍴1
0 1
1 1
```

IT IS ASSUMED THAT ∧/,ARRAY∈ 0 1 ↔ 1.

<PANDREDUCE> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: *SAUCE/PANDSCAN.1*

TYPE: *FUNCTION*

SUMMARY: *PARTITIONED ∧\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.*

TIMESTAMP: 1982-11-16 22:46:05

CATEGORIES:
3 *PARTITIONED ARRAY HANDLING*
41 *BOOLEAN ARRAYS*

DESCRIPTION:
RESULT←BOOLEAN PANDSCAN ARRAY

PARTITIONED ∧\ARRAY. <BOOLEAN> IS A VECTOR WITH ⍴BOOLEAN ↔ ¯1↑⍴ARRAY (ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLE

```
      1 0 0 0 1 0 1 0 PANDSCAN 1 1 0 1 1 0 0 1
  ↔  (∧\ 1 1 0 1),(∧\ 1 1),(∧\ 0 0)


      1 0 1 0 0 0 PANDSCAN 2 6 ⍴ 1 0 1 0 1 0 , 1 1 0 1 1 1
1 0 1 0 0 0
1 1 0 0 0 0
```

IT IS ASSUMED THAT ∧/,ARRAY∈ 0 1 ↔ 1.

<PANDSCAN> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/PEQSCAN.1

TYPE: FUNCTION

SUMMARY: PARTITIONED =\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

TIMESTAMP: 1982-12-08 07:56:19

CATEGORIES: 3 PARTITIONED ARRAY HANDLING
41 BOOLEAN ARRAYS

DESCRIPTION:
RESULT←BOOLEAN PEQSCAN ARRAY

PARTITIONED =\ARRAY. <BOOLEAN> IS A VECTOR WITH ρBOOLEAN ↔ ¯1↑ρARRAY
(ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES
THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES

```
      1 0 0 0 1 0 1 0 PEQSCAN 1 1 0 1 1 0 0 1
↔  (=\ 1 1 0 1),(=\ 1 0),(=\ 0 1)

      1 0 1 0 0 0 PEQSCAN 2 6 ρ 1 0 1 0 1 0 , 1 1 0 1 1 1
1 0 1 0 0 1
1 1 0 0 0 0
```

IT IS ASSUMED THAT ∧/,ARRAY∈ 0 1 ↔ 1.

<PEQSCAN> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: *SAUCE/PGRADEDOWN.1*

TYPE: *FUNCTION*

SUMMARY: *PARTITIONED ⍒ OVER THE LAST AXIS OF OF AN ARRAY.*

TIMESTAMP: 1983-02-18 05:22:32

CATEGORIES: 3 *PARTITIONED ARRAY HANDLING*
6 *SORTING AND GRADING*

DESCRIPTION:
RESULT←BOOLEAN PGRADEDOWN ARRAY

PARTITIONED ⍒ OVER THE LAST AXIS OF <ARRAY>. <BOOLEAN> IS A VECTOR WITH ⍴BOOLEAN ↔ ¯1↑⍴ARRAY (ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES:

```
      1 0 0 0 1 0 1 0 PGRADEDOWN 2 5 3 4 9 8 0.3 ¯1
↔  (⍒ 2 5 3 4),(⍒ 9 8),(⍒ 0.3 ¯1)

      1 0 1 0 0 0 PGRADEDOWN 2 6 ⍴ 1 3 2 6 4 8 , 0.2 12 ¯1 ¯2 ¯3 ¯4
↔  2 6 ⍴⎕IO+ 1 0 3 1 2 0 , 1 0 3 2 1 0
```

<PGRADEDOWN> HAS BEEN THOROUGHLY TESTED. THE RESULT VARIES APPROPRIATELY ACCORDING TO THE VALUE OF ⎕IO. IF THE FUNCTION BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: *SAUCE/PGRADEUP.1*

TYPE: *FUNCTION*

SUMMARY: *PARTITIONED ⍋ OVER THE LAST AXIS OF AN ARRAY.*

TIMESTAMP: 1983-02-18 05:28:21

CATEGORIES: 3 *PARTITIONED ARRAY HANDLING*
6 *SORTING AND GRADING*

DESCRIPTION:
RESULT←BOOLEAN PGRADEUP ARRAY

PARTITIONED ⍋ OVER THE LAST AXIS OF <ARRAY>. <BOOLEAN> IS A VECTOR WITH ⍴BOOLEAN ↔ ¯1↑⍴ARRAY (ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES:
```
      1 0 0 0 1 0 1 0 PGRADEUP 2 5 3 4 9 8 0.3 ¯1
  ↔   (⍋ 2 5 3 4),(⍋ 9 8),(⍋ 0.3 ¯1)

      1 0 1 0 0 0 PGRADEUP 2 6 ⍴ 1 3 2 6 4 8 , 0.2 12 ¯1 ¯2 ¯3 ¯4
  ↔   2 6 ⍴⎕IO+ 0 1 0 2 1 3 , 0 1 0 1 2 3
```

<PGRADEUP> HAS BEEN THOROUGHLY TESTED. THE RESULT VARIES APPROPRIATELY ACCORDING TO THE VALUE OF ⎕IO. IF THE FUNCTION BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: *SAUCE/PLENGTHS.2*

TYPE: *FUNCTION*

SUMMARY: *E.G. PLENGTHS* 1 1 0 0 1 0 0 0 0 1 0 ↔ 1 3 5 2.

TIMESTAMP: 1983-10-18 21:42:52

CATEGORIES: 3 *PARTITIONED ARRAY HANDLING*

CHANGES:
CODE SLIGHTLY IMPROVED, AS SUGGESTED BY J. HENRI SCHUELER.

DESCRIPTION:
LENGTHS←PLENGTHS BOOLEAN

E.G. PLENGTHS 1 1 0 0 1 0 0 0 0 1 0 ↔ 1 3 5 2.

<PLENGTHS> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/PLESCAN.1

TYPE: FUNCTION

SUMMARY: PARTITIONED ≤\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

TIMESTAMP: 1982-12-08 09:10:41

CATEGORIES: 3 PARTITIONED ARRAY HANDLING
41 BOOLEAN ARRAYS

DESCRIPTION:
RESULT←BOOLEAN PLESCAN ARRAY

PARTITIONED ≤\ARRAY. <BOOLEAN> IS A VECTOR WITH ⍴BOOLEAN ↔ ¯1↑⍴ARRAY
(ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES
THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES

```
  1 0 0 0 1 0 1 0 PLESCAN 1 1 0 1 1 0 0 1
  ↔  (≤\ 1 1 0 1),(≤\ 1 0),(≤\ 0 1)

      1 0 1 0 0 0 PLESCAN 2 6 ⍴ 1 0 1 0 1 0 , 1 1 0 1 1 1
1 0 1 0 1 1
1 1 0 1 1 1
```

IT IS ASSUMED THAT ∧/,ARRAY∈ 0 1 ↔ 1.

<PLESCAN> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/PLTSCAN.1

TYPE: FUNCTION

SUMMARY: PARTITIONED <\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

TIMESTAMP: 1982-12-06 09:02:41

CATEGORIES: 3 PARTITIONED ARRAY HANDLING
41 BOOLEAN ARRAYS

DESCRIPTION:
RESULT←BOOLEAN PLTSCAN ARRAY

PARTITIONED <\ARRAY. <BOOLEAN> IS A VECTOR WITH ρBOOLEAN ↔ ¯1↑ρARRAY
(ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES
THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES

```
      1 0 0 0 1 0 1 0 PLTSCAN 1 1 0 1 1 0 0 1
↔  (<\ 1 1 0 1),(<\ 1 0),(<\ 0 1)

      1 0 1 0 0 0 PLTSCAN 2 6 ρ 1 0 1 0 1 0 , 1 1 0 1 1 1
1 0 1 0 0 0
1 0 0 1 0 0
```

IT IS ASSUMED THAT ∧/,ARRAYϵ 0 1 ↔ 1.

<PLTSCAN> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: *SAUCE/PMAXREDUCE.1*

TYPE: *FUNCTION*

SUMMARY: *PARTITIONED ⌈/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.*

TIMESTAMP: 1982-11-18 03:17:21

CATEGORIES:
3 *PARTITIONED ARRAY HANDLING*
5 *NUMERIC CALCULATION*
7 *SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF*

DESCRIPTION:
RESULT←BOOLEAN PMAXREDUCE ARRAY

PARTITIONED ⌈/ARRAY. <BOOLEAN> IS A VECTOR WITH ⍴BOOLEAN ↔ ¯1↑⍴ARRAY (ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES

```
      1 0 0 0 1 0 1 0 PMAXREDUCE ¯1 2 8 3 9 0.5 12 14
  ↔  (⌈/ ¯1 2 8 3),(⌈/ 9 0.5),(⌈/ 12 14)


      1 0 1 0 0 0 PMAXREDUCE 2 6 ⍴ 1 2 6 4 5 3 , 0.5 ¯3 ¯5 ¯4 ¯6 ¯6
2    ¯6
0.5 ¯4
```

<PMAXREDUCE> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/PMAXSCAN.1

TYPE: FUNCTION

SUMMARY: PARTITIONED ⌈\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

TIMESTAMP: 1982-11-18 03:28:33

CATEGORIES:
3 PARTITIONED ARRAY HANDLING
5 NUMERIC CALCULATION
7 SEARCHING ⍷ INCLUDING MEMBERSHIP AND INDEX-OF

DESCRIPTION:
RESULT←BOOLEAN PMAXSCAN ARRAY

PARTITIONED ⌈\ARRAY. <BOOLEAN> IS A VECTOR WITH ⍴BOOLEAN ↔ ¯1↑⍴ARRAY
(ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES
THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES

```
      1 0 0 0 1 0 1 0 PMAXSCAN ¯1 2 8 3 9 0.5 12 14
↔  (⌈\ ¯1 2 8 3),(⌈\ 9 0.5),(⌈\ 12 14)


      1 0 1 0 0 0 PMAXSCAN 2 6 ⍴ 1 2 5 6 4 3 , 0.5 ¯3 ¯4 ¯2 ¯6 ¯6
1    2    5    ¯6    ¯6    ¯6
0.5  0.5 ¯4    ¯2    ¯2    ¯2
```

<PMAXSCAN> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: *SAUCE/PMINREDUCE.1*

TYPE: *FUNCTION*

SUMMARY: *PARTITIONED ⌊/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.*

TIMESTAMP: 1982-11-19 04:40:39

CATEGORIES:
3 *PARTITIONED ARRAY HANDLING*
5 *NUMERIC CALCULATION*
7 *SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF*

DESCRIPTION:
RESULT←BOOLEAN PMINREDUCE ARRAY

PARTITIONED ⌊/ARRAY. <BOOLEAN> IS A VECTOR WITH ⍴BOOLEAN ↔ ¯1↑⍴ARRAY (ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES

```
      1 0 0 0 1 0 1 0 PMINREDUCE ¯1 2 ¯8 3 9 0.5 12 14
  ↔  (⌊/ ¯1 2 ¯8 3),(⌊/ 9 0.5),(⌊/ 12 14)


      1 0 1 0 0 0 PMINREDUCE 2 6 ⍴ 1 2 6 4 5 3 , 0.5 ¯3 ¯5 ¯4 ¯6 ¯6
 ¯1  ¯3
 ¯3  ¯6
```

<PMINREDUCE> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/PMINSCAN.1

TYPE: FUNCTION

SUMMARY: PARTITIONED ⌊\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

TIMESTAMP: 1982-11-19 07:04:46

CATEGORIES:
3 PARTITIONED ARRAY HANDLING
5 NUMERIC CALCULATION
7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF

DESCRIPTION:
RESULT←BOOLEAN PMINSCAN ARRAY

PARTITIONED ⌊\ARRAY. <BOOLEAN> IS A VECTOR WITH ⍴BOOLEAN ↔ ¯1↑⍴ARRAY
(ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES
THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES

```
      1 0 0 0 1 0 1 0 PMINSCAN ¯1 2 ¯8 3 9 0.5 12 14
  ↔   (⌊\ ¯1 2 ¯8 3),(⌊\ 9 0.5),(⌊\ 12 14)

      1 0 1 0 0 0 PMINSCAN 2 6 ⍴ 1 2 6 4 5 3 , 0.5 ¯3 ¯5 ¯4 ¯6 ¯6
1    1    6    4    4    3
0.5 ¯3   ¯5   ¯5   ¯6   ¯6
```

<PMINSCAN> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/PNESCAN.1

TYPE: FUNCTION

SUMMARY: PARTITIONED ≠\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

TIMESTAMP: 1982-12-08 08:13:57

CATEGORIES: 3 PARTITIONED ARRAY HANDLING
41 BOOLEAN ARRAYS

DESCRIPTION:
RESULT←BOOLEAN PNESCAN ARRAY

PARTITIONED ≠\ARRAY. <BOOLEAN> IS A VECTOR WITH ρBOOLEAN ↔ ¯1↑ρARRAY
(ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES
THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES

```
      1 0 0 0 1 0 1 0 PNESCAN 1 1 0 1 1 0 0 1
  ↔   (≠\ 1 1 0 1),(≠\ 1 0),(≠\ 0 1)

      1 0 1 0 0 0 PNESCAN 2 6 ρ 1 0 1 0 1 0 , 1 1 0 1 1 1
1 1 1 1 0 0
1 0 0 1 0 1
```

IT IS ASSUMED THAT ∧/,ARRAY∈ 0 1 ↔ 1.

<PNESCAN> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/PORREDUCE.1

TYPE: FUNCTION

SUMMARY: PARTITIONED ∨/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

TIMESTAMP: 1982-11-16 22:50:48

CATEGORIES: 3 PARTITIONED ARRAY HANDLING
41 BOOLEAN ARRAYS

DESCRIPTION:
RESULT←BOOLEAN PORREDUCE ARRAY

PARTITIONED ∨/ARRAY. <BOOLEAN> IS A VECTOR WITH ρBOOLEAN ↔ ¯1↑ρARRAY
(ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES
THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLE

```
      1 0 0 0 1 0 1 0 PORREDUCE 0 1 0 0 1 1 0 0
↔     (∨/ 0 1 0 0),(∨/ 1 1),(∨/ 0 0)

      1 0 1 0 0 0 PORREDUCE 2 6 ρ1,(10ρ0),1
1 0
0 1
```

IT IS ASSUMED THAT ∧/,ARRAY∈ 0 1 ↔ 1.

<PORREDUCE> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/PORSCAN.1

TYPE: FUNCTION

SUMMARY: PARTITIONED ⍲\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

TIMESTAMP: 1982-11-16 22:55:24

CATEGORIES: 3 PARTITIONED ARRAY HANDLING
41 BOOLEAN ARRAYS

DESCRIPTION:
RESULT←BOOLEAN PORSCAN ARRAY

PARTITIONED ⍲\ARRAY. <BOOLEAN> IS A VECTOR WITH ⍴BOOLEAN ↔ ¯1↑⍴ARRAY
(ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES
THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLE

```
      1 0 0 0 1 0 1 0 PORSCAN 0 1 0 1 1 0 0 1
  ↔  (⍲\ 0 1 0 1),(⍲\ 1 0),(⍲\ 0 1)


      1 0 1 0 0 0 PORSCAN 2 6 ⍴ 0 1 0 1 0 1 , 0 0 0 0 1 0
0 1 0 1 1 1
0 0 0 0 1 1
```

IT IS ASSUMED THAT ∧/,ARRAY∈ 0 1 ↔ 1.

<PORSCAN> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: *SAUCE/PPLUSREDUCE.1*

TYPE: *FUNCTION*

SUMMARY: *PARTITIONED +/ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.*

TIMESTAMP: 1982-11-16 23:00:06

CATEGORIES: 3 *PARTITIONED ARRAY HANDLING*
5 *NUMERIC CALCULATION*

DESCRIPTION:
RESULT←BOOLEAN PPLUSREDUCE ARRAY

PARTITIONED +/ARRAY. <BOOLEAN> IS A VECTOR WITH ρBOOLEAN ↔ ¯1↑ρARRAY (ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES

```
      1 0 0 0 1 0 1 0 PPLUSREDUCE ¯1 2 8 3 9 0.5 12 14
↔  (+/ ¯1 2 8 3),(+/ 9 0.5),(+/ 12 14)

      1 0 1 0 0 0 PPLUSREDUCE 2 6 ρ 1 2 6 4 5 3 , 0.5 ¯3 ¯5 ¯4 ¯6 ¯6
 3    18
¯2.5 ¯21
```

<PPLUSREDUCE> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/PPLUSSCAN.1

TYPE: FUNCTION

SUMMARY: PARTITIONED +\ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

TIMESTAMP: 1982-11-16 22:34:28

CATEGORIES: 3 PARTITIONED ARRAY HANDLING
5 NUMERIC CALCULATION

DESCRIPTION:
RESULT←BOOLEAN PPLUSSCAN ARRAY

PARTITIONED +\ARRAY. <BOOLEAN> IS A VECTOR WITH ⍴BOOLEAN ↔ ¯1↑⍴ARRAY
(ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES
THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES

```
      1 0 0 0 1 0 1 0 PPLUSSCAN ¯1 2 8 3 9 0.5 12 14
  ↔  (+\ ¯1 2 8 3),(+\ 9 0.5),(+\ 12 14)

      1 0 1 0 0 0 PPLUSSCAN 2 6 ⍴ 1 2 6 4 5 3 , 0.5 ¯3 ¯5 ¯4 6 6
1    3   6   10   15   18
0.5 ¯2.5 ¯5  ¯9   ¯3    3
```

<PPLUSSCAN> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/PREVERSE.1 SAUCE/PREVERSE.1

TYPE: FUNCTION

SUMMARY: PARTITIONED ⌽ARRAY; EACH 1 IN THE BOOLEAN LEFT ARG DESIGNATES THE BEGINNING OF A PART IN THE ARRAY.

TIMESTAMP: 1982-11-24 03:29:52

CATEGORIES: 2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.
3 PARTITIONED ARRAY HANDLING

DESCRIPTION:
RESULT←BOOLEAN PREVERSE ARRAY

PARTITIONED ⌽ARRAY. <BOOLEAN> IS A VECTOR WITH ⍴BOOLEAN ↔ ¯1↑⍴ARRAY
(ALTHOUGH SCALAR EXTENSION MAY BE PERFORMED). EACH 1 IN <BOOLEAN> DESIGNATES
THE BEGINNING OF A PART OF <ARRAY>.

EXAMPLES

```
      1 0 0 0 1 0 1 0 PREVERSE 'ABCDXYMN'
  ↔   (⌽'ABCD'),(⌽'XY'),(⌽'MN')

      ⎕IO←1
      1 0 1 0 0 0 PREVERSE 2 6 ⍴⍳12
2  1  6  5  4  3
8  7 12 11 10  9
```

<ARRAY> MAY CONTAIN CHARACTERS, NUMBERS, OR ENCLOSURES.

<PREVERSE> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: SAUCE/PROINDEXOF.1 SAUCE/PROINDEXOF.1

TYPE: FUNCTION

SUMMARY: PROGRESSIVE DYADIC IOTA, E.G. 'ABA' PROINDEXOF 'ACAABA' ↔ ⎕IO+ 0 3 2 3 1 3.

TIMESTAMP: 1982-11-09 06:19:20

CATEGORIES: 7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
INDICES←SEQUENCE PROINDEXOF VECTOR

PROGRESSIVE DYADIC IOTA, E.G. 'ABA' PROINDEXOF 'ACAABA' ↔ ⎕IO+ 0 3 2 3 1 3.
THE ARGUMENTS MAY CONTAIN CHARACTERS, NUMBERS, OR ENCLOSURES.

<PROINDEXOF> HAS BEEN THOROUGHLY TESTED. THE RESULT VARIES APPROPRIATELY
ACCORDING TO THE VALUE OF ⎕IO. IF THE FUNCTION BECOMES SUSPENDED, IT CAN BE
RESTARTED BY →⎕LC.

TITLE: SAUCE/RECURRENCE.1

TYPE: FUNCTION

SUMMARY: RESULT[I]←LEFTARGUMENT[I]+RIGHTARGUMENT[I]×RESULT[I-1]; RESULT[⎕IO]←LEFTARGUMENT[⎕IO].

TIMESTAMP: 1982-12-08 08:22:59

CATEGORIES: 5 NUMERIC CALCULATION

DESCRIPTION:
SEQUENCE←ADDEND RECURRENCE MULTIPLIER

SEQUENCE[I]←ADDEND[I]+MULTIPLIER[I]×SEQUENCE[I-1]; SEQUENCE[⎕IO]←ADDEND[⎕IO].

IT IS ASSUMED THAT ~0∊MULTIPLIER.

<RECURRENCE> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: SAUCE/REMOVETRAIL.1

TYPE: FUNCTION

SUMMARY: REMOVES TRAILING COLUMNS OR ELEMENTS FROM AN ARRAY.

TIMESTAMP: 1983-03-02 15:43:07

CATEGORIES: 25 FORMATTING
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
COMPRESSED←[ELEMENTS] REMOVETRAIL ARRAY

REMOVES ALL TRAILING COLUMNS CONSISTING SOLELY OF <ELEMENTS> FROM <ARRAY>.
DEFAULT <ELEMENTS> IS 1↑0⍴ARRAY (I.E. 0 OR '').

EXAMPLES

```
      REMOVETRAIL '  AB C  D    '  ↔  '  AB C  D'

      '∘⎕' REMOVETRAIL 'ABC∘∘∘    ↔   'ABC
                        ⎕∘X∘∘⎕         ⎕∘X
                        ⎕⎕⎕⎕⎕⎕'        ⎕⎕⎕'
```

<REMOVETRAIL> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/RIGHTJUSTIFY.1

TYPE: FUNCTION

SUMMARY: RIGHT JUSTIFIES EACH ROW OF AN ARRAY.

TIMESTAMP: 1983-03-02 16:28:07

CATEGORIES:
- 2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.
- 25 FORMATTING
- 30 REPORT FORMATTING

DESCRIPTION:
RESULT←[ELEMENT] RIGHTJUSTIFY ARRAY

THE DEFAULT <ELEMENT> IS 1↑0ρARRAY (I.E. 0 OR ''). THE FUNCTION RIGHT JUSTIFIES EACH ROW OF <ARRAY> WITH RESPECT TO <ELEMENT>. NOTE: IT WILL WORK EVEN IF <ELEMENT> HAS MORE THAT ONE DISTINCT ELEMENT; IN THIS CASE, RIGHT JUSTIFICATION IS DONE WITH RESPECT TO ALL THE ELEMENTS OF <ELEMENT>.

```
E.G.  RIGHTJUSTIFY 'JOE      ↔  '  JOE
                     MARY         MARY
                    JOHN          JOHN
                    ED    '         ED'
```

```
E.G.  ' ∘' RIGHTJUSTIFY 'JOE∘∘∘   ↔  '∘∘∘JOE
                         ∘∘MARY       ∘∘MARY
                         ∘JOHN         ∘JOHN
                         ∘ED  ∘'        ∘∘ED'
```

<ARRAY> MAY CONTAIN CHARACTERS, NUMBERS, OR ENCLOSURES.

<RIGHTJUSTIFY> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

- -

TITLE: SAUCE/RUNLENGTHS.1 SAUCE/RUNLENGTHS.1

TYPE: FUNCTION

SUMMARY: E.G. RUNLENGTHS 1 9 9 4 4 4 9 ↔ 1 2 3 1.

TIMESTAMP: 1982-12-02 07:10:00

CATEGORIES:
- 7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF
- 39 STATISTICS AND PROBABILITY

DESCRIPTION:
LENGTHS←RUNLENGTHS VECTOR

E.G. RUNLENGTHS 1 9 9 4 4 4 9 ↔ 1 2 3 1. <VECTOR> MAY BE CHARACTER OR NUMERIC.

<RUNLENGTHS> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/SQUEEZE.1

TYPE: FUNCTION

SUMMARY: REMOVES ALL LEADING, TRAILING, AND REDUNDANT ELEMENTS (SPEC'D IN THE LEFT ARGUMENT) FROM A VECTOR.

TIMESTAMP: 1982-11-16 00:04:25

CATEGORIES: 25 FORMATTING
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
COMPRESSED←[ELEMENT] SQUEEZE VECTOR

DEFAULT <ELEMENT> IS 1↑0⍴VECTOR (I.E. 0 OR ''). THE RESULT IS LIKE <VECTOR>,
BUT WITH ALL LEADING, TRAILING, AND REDUNDANT <ELEMENT>S REMOVED.

EXAMPLE
SQUEEZE ' AB C DE F ' ↔ 'AB C DE F'

<VECTOR> MAY CONTAIN CHARACTERS OR NUMBERS; IF <ELEMENT> IS SPECIFIED,
<VECTOR> MAY ALSO CONTAIN ENCLOSURES.

<SQUEEZE> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH
EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/STRINGFIND.2

TYPE: FUNCTION

SUMMARY: FINDS ALL OCCURRENCES OF ONE VECTOR IN ANOTHER.

TIMESTAMP: 1984-09-14 22:42:02

CATEGORIES:
7 SEARCHING ⍝ INCLUDING MEMBERSHIP AND INDEX-OF
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
INDICES←VECTOR STRINGFIND SUBSTRING

RETURNS THE INDEX OF THE BEGINNING OF EACH OCCURRENCE OF <SUBSTRING> IN <VECTOR>.

EXAMPLES:

```
      ⎕IO←1
      'BANANA' STRINGFIND 'ANA'
2 4
      ⍴⎕←'BANANA' STRINGFIND 'NAN'
3
1
      'BANANA' STRINGFIND ''    ⍝      VECTOR STRINGFIND '' ↔ ⍳1+⍴VECTOR
1 2 3 4 5 6 7                   ⍝ AS SUGGESTED BY A. D. FALKOFF IN 'A NOTE ON
                                ⍝ MATCHING: WHERE DO YOU FIND THE MATCH TO AN
                                ⍝ EMPTY ARRAY?', APL79 CONFERENCE PROCEEDINGS.
      ⎕IO←0
      'BANANA' STRINGFIND 'ANA'
1 3
```

<STRING> MAY BE CHARACTER OR NUMERIC.

<STRINGFIND> HAS BEEN THOROUGHLY TESTED. THE RESULT VARIES APPROPRIATELY ACCORDING TO THE VALUE OF ⎕IO. IF THE FUNCTION BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: SAUCE/VTOM.1

TYPE: FUNCTION

SUMMARY: RETURNS A MAT WITH ONE STRING PER ROW, WHERE EACH STRING IN THE RIGHT ARG IS PRECEDED BY 1↑RIGHTARG.

TIMESTAMP: 1982-12-28 01:46:41

CATEGORIES:
2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.
25 FORMATTING
27 TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS
30 REPORT FORMATTING

DESCRIPTION:
MATRIX←[ELEMENT] VTOM VECTOR

(VECTOR TO MATRIX.) THE FUNCTION RETURNS A LEFT-JUSTIFIED MATRIX WITH ONE STRING PER ROW, WHERE EACH STRING IN <VECTOR> IS PRECEDED BY 1↑VECTOR. <ELEMENT> SPECIFIES THE FILL ELEMENT USED FOR JUSTIFICATION; THE DEFAULT <ELEMENT> IS 1↑0⍴VECTOR (I.E. 0 OR ' '). <VECTOR> MAY BE CONTAIN CHARACTERS, NUMBERS, OR ENCLOSURES.

EXAMPLES

```
      VTOM ' THE KITTEN RAN AWAY.'
THE
KITTEN
RAN
AWAY.

      '*' VTOM '∘HEY∘DIDDLE∘∘DIDDLE∘∘∘'

HEY***
DIDDLE
******
DIDDLE
******
******
******
```

SAUCE/VTOM IS FREQUENTLY USED WITH SAUCE/SQUEEZE, AS FOLLOWS.

```
      VTOM ' ',⎕←SQUEEZE ' JOHN   GEORGE ED     TED   '
JOHN GEORGE ED TED
JOHN
GEORGE
ED
TED
```

<VTOM> HAS BEEN THOROUGHLY TESTED. IT WILL OPERATE CORRECTLY WITH EITHER VALUE OF ⎕IO. IF IT BECOMES SUSPENDED, IT CAN BE RESTARTED BY →⎕LC.

TITLE: TS/GETOFFSET.1

TYPE: FUNCTION

SUMMARY: RETURNS AN OFFSET FROM UTC, IN SECONDS, FOR THE NODE SPECIFIED IN THE ARGUMENT.

TIMESTAMP: 1983-02-23 00:21:13

CATEGORIES: 37 TIMES AND DATES

DESCRIPTION:
OFFSET←[TS] GETOFFSET NODE

RETURNS AN OFFSET FROM UNIVERSAL COORDINATED TIME (UTC), IN SECONDS, FOR <NODE>. <NODE> MUST BE A SINGLE INTEGER SPECIFYING A NODE IN THE I.P. SHARP COMMUNICATIONS NETWORK. THE OFFSET IS BASED ON EITHER THE CURRENT TIME AND DATE, OR THE TIMESTAMP SPECIFIED EXPLICITLY IN THE OPTIONAL LEFT ARGUMENT, <TS>. <TS> MAY BE IN EITHER ⎕TS OR ⎕RDCI FORMAT.

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TS/LTS.2

TITLE: TS/LTS.2

TYPE: FUNCTION

SUMMARY: RETIRNS THE LOCAL TIMESTAMP WHEN GIVEN THE TIMESTAMP IN UTC.

TIMESTAMP: 1984-07-26 14:00:20

CATEGORIES: 37 TIMES AND DATES

DESCRIPTION:
TS←[OFFSET] LTS TS

RETURNS THE LOCAL TIMESTAMP WHEN GIVEN THE TIMESTAMP IN UNIVERSAL COORDINATED TIME (UTC). <TS>, THE UTC TIMESTAMP, MAY BE IN EITHER ⎕TS OR ⎕RDCI FORMAT. THE LOCAL OFFSET FROM UTC IS BASED ON EITHER THE CURRENT NODE OF THE I.P. SHARP COMMUNICATIONS NETWORK THAT THE USER IS CONNECTED TO, OR MAY BE SPECIFIED EXPLICITLY IN THE OPTIONAL LEFT ARGUMENT, <OFFSET>.

IF <OFFSET> IS PROVIDED, IT MUST BE A SINGLE INTEGER SPECIFYING THE OFFSET FROM UTC IN SECONDS. THE PERMISSIBLE RANGE IS ¯43200 ≤ OFFSET ≤ 43200 (43200 SECONDS = 12 HOURS).

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: TS/NODE.1

TYPE: FUNCTION

SUMMARY: RETURNS NODE NUMBERS AND LOCATIONS OF (SPECIFIED) NODES OR LOCATIONS ON THE IPSA NETWORK.

TIMESTAMP: 1983-02-23 00:22:24

CATEGORIES: 1 MISCELLANEOUS

DESCRIPTION:
TBL←NODE SPC

IF <SPC> IS A NUMERIC ARRAY, THE RESULT IS A (CHARACTER) TABLE OF NODE NUMBERS AND THEIR LOCATIONS. FOR EXAMPLE

```
      NODE 1 3 86
   1 TORONTO, ONTARIO, CANADA
   3 TORONTO, ONTARIO, CANADA
  86 NEW ORLEANS, LA, U.S.A.
```

<SPC> MAY ALSO BE A MATRIX NAMELIST CONTAINING PREFIXES OF LOCATIONS OR IT MAY BE A VECTOR (OR SCALAR) CONTAINING A SINGLE PREFIX. FOR EXAMPLE

```
     NODE 'NEW'
  17 NEWPORT BEACH, CA, U.S.A.
  27 NEW YORK, NY, U.S.A.
  66 NEW YORK, NY, U.S.A.
  86 NEW ORLEANS, LA, U.S.A.
 117 NEWPORT BEACH, CA, U.S.A.
```

THIS FUNCTION IS □IO-INDEPENDENT.

TITLE: TS/UTC.2 TS/UTC.2

TYPE: FUNCTION

SUMMARY: RETURNS A UTC TIMESTAMP FROM AN ARGUMENT IN LOCAL TIMESTAMP.

TIMESTAMP: 1984-07-26 13:55:32

CATEGORIES: 37 TIMES AND DATES

DESCRIPTION:
TS←[OFFSET] UTC TS

RETURNS A UNIVERSAL COORDINATED (UTC) TIMESTAMP WHEN GIVEN THE LOCAL TIMESTAMP. <TS>, THE LOCAL TIMESTAMP, MAY BE IN EITHER ⎕TS OR ⎕RDCI FORMAT. THE LOCAL OFFSET FROM UTC IS BASED ON EITHER THE CURRENT NODE OF THE I.P. SHARP COMMUNICATIONS NETWORK THAT THE USER IS CONNECTED TO, OR MAY BE SPECIFIED EXPLICITLY IN THE OPTIONAL LEFT ARGUMENT, <OFFSET>.

IF <OFFSET> IS PROVIDED, IT MUST BE A SINGLE INTEGER SPECIFYING THE OFFSET FROM UTC IN SECONDS. THE PERMISSIBLE RANGE IS ¯43200 ≤ OFFSET ≤ 43200 (43200 SECONDS = 12 HOURS).

THIS FUNCTION IS ⎕IO-INDEPENDENT.

TITLE: WGR/FTT.2 WGR/FTT.2

TYPE: FUNCTION

SUMMARY: FORMS FORMATTED REPRESENTATION OF TIMESTAMPS FROM ⎕RDCI FORM.ALLOWS VECTOR INPUT

TIMESTAMP: 1984-02-17 20:49:57

CATEGORIES: 9 FILES
13 FILE TOOLS
37 TIMES AND DATES

DESCRIPTION:
FORMS FORMATTED REPRESENTATION OF TIMESTAMPS FROM ⎕RDCI FORM. SIMILAR TO FUNCTION FTT IN WS 1 TS BUT ALLOWS VECTOR INPUT AND RETURNS MATRIX RESULT. ARGUMENT IS VECTOR OF TIMESTAMPS AS ENCODED NUMERIC VECTORS FROM ⎕RDCI[3] RESULT IS ARRAY OF TIMESTAMPS AS NUMERIC MATRIX. EACH TIMESTAMP HAS THE FORM OF ⎕TS.

TITLE: *WGR/PACKSHOW.1*

TYPE: *FUNCTION*

SUMMARY: *FUNCTION TO DISPLAY CONTENTS OF PACKAGE(INCLUDING FUNCTIONS AND OTHER PACKAGES) IN NEAT FORM*

TIMESTAMP: 1983-12-08 15:58:31

CATEGORIES:
 8 *PACKAGES*
 35 *WORKSPACE TOOLS ⍝ E.G. WORKSPACE CROSS REFERENCE*

DESCRIPTION:
PACKSHOW IS A SINGLE FUNCTION THAT DISPLAYS AS A HEIRARCHIAL STRUCTURE,THE CONTENTS OF A PACKAGE GIVEN AS ARGUMENT. THE RIGHT ARGUMENT IS A PACKAGE. THE OPTIONAL LEFT ARGUMENT IS A CHARACTER STRING USED TO NAME THE ROOT LEVEL OF PACKAGE.IF NO LEFT ARGUMENT IS GIVEN AND RIGHT ARGUMENT IS PACKAGE,NO ROOT NAME IS USED.
FUNCTIONS ARE EXTRACTED AND DISPLAYED IN 1 ⎕FD FORM ,VARIABLES ARE PRECEDED BY SHAPE VECTOR IN ASSIGNMENT FORM. UNDEFINED NAMES ARE REPRESENTED BY ?. LOCKED FUNCTIONS BY ⍫

- -

TITLE: *WGR/REPORT.1*

TYPE: *FUNCTION*

SUMMARY: *RETURNS A SUMMARY REPORT OF FILES IN A LIBRARY OR SET OF FILENAMES.*

TIMESTAMP: 1983-11-17 17:26:33

CATEGORIES:
 9 *FILES*
 12 *FILE ORGANIZATION*
 13 *FILE TOOLS*

DESCRIPTION:
REPORT RETURNS A CHARACTER MATRIX CONTAINING A DIRECTORY LISTING OF FILES USED IN IN A USER ACCOUNT OR IN A LIST OF FILENAMES IN ⎕NAMES FORMAT.
THE LEFT ARGUMENT IS OPTIONAL AND CONTAINS THE FILE LOCK NUMBER TO BE USED. A FILE KEY OF 0 IS ASSUMED IF ARGUMENT IS OMITTED.
THE RIGHT ARGUMENT IS EITHER A NUMERIC ACCOUNT NUMBER OR OR A MATRIX OF NAMES IN FORMAT RETURNED BY ⎕LIB OR ⎕NAMES. THE RESULT IS A TABLE LISTING FILENAME ⎕SIZE ,FORMATTED ⎕RDCI OF LAST COMPONENT.
IF ANY INFORMATION IS UNAVAILABLE(DUES TO ACCESS MATRIX RESTRICTIONS),IT IS LEFT BLANK.
THE TABLE HAS SUMMARY OF STORAGE USED AND RESERVED AT BOTTOM.

TITLE: *WGR/TTF.1*

TYPE: *FUNCTION*

SUMMARY: *FORMS ⎕RDCI[3] FORMAT ENCODED TIMESTAMP VECTOR FROM MATRIX OF ⎕TS FORM TIMESTAMPS*

TIMESTAMP: 1984-02-17 21:09:01

CATEGORIES:
9 *FILES*
13 *FILE TOOLS*
37 *TIMES AND DATES*

DESCRIPTION:
FORMS ⎕RDCI[3] FORMAT ENCODED TIMESTAMP VECTOR FROM MATRIX OF ⎕TS FORM TIMESTAMPS. SIMILAR TO FUNCTION TTF IN WS 1 TS BUT ALLOWS MATRIX INPUT AND RETURNS VECTOR RESULT. ARGUMENT IS VECTOR OF TIMESTAMPS AS NUMERIC ARRAYS FROM ⎕TS . RESULT IS VECTOR OF TIMESTAMPS AS ENCODED NUMERIC VECTOR.

- -

TITLE: *WHAM/TOKEN∆DECIMALS.1*

TYPE: *FUNCTION*

SUMMARY: *RETURN NO. DECIMALS IN 'NUMBERS' ALONG ROWS OF TEXT MATRIX*

TIMESTAMP: 1984-12-13 12:26:44

CATEGORIES:
20 *TERMINAL INPUT/OUTPUT*
21 *TERMINAL INPUT*

DESCRIPTION:
FUNCTION RETURNS NUMBER OF DECIMAL PLACES PER SINGLE NON-SCIENTIFIC NUMBER ON EACH ROW A TEXT ARRAY.

FUNCTION DOES NOT CHECK THAT ROWS CONTAIN VALID NUMBERS (WHAT ELSE IS ⎕FI FOR?), AND DEEMS DECIMALS TO BE ALL CHARACTERS BETWEEN THE FIRST '.' ON EACH ROW AND THE NEXT SPACE.

0 *IN THE RESULT MEANS THAT NO DECIMAL POINT EXISTED IN THE ASSOCIATED ROW,* ¯1 *MEANS THAT A DECIMAL POINT DID EXIST, BUT IT WAS IMMEDIATELY FOLLOWED BY SPACE OR END OF ROW - IT IS PRESUMED THAT WITH STRICT DECIMAL CHECKING THESE WILL PROBABLY BE INVALID.*

```
TITLE:        WHAM/TOKEN∆LEADZ.1                                          WHAM/TOKEN∆LEADZ.1

TYPE:         FUNCTION

SUMMARY:      RETURNS WHETHER SINGLE NUMBER ON EACH ROW OF TEXT ARRAY CONTAINS LEAD ZEROES

TIMESTAMP:    1984-12-13 12:38:17

CATEGORIES:   20  TERMINAL INPUT/OUTPUT
              21  TERMINAL INPUT

DESCRIPTION:
FUNCTION RETURNS WHETHER SINGLE SINGLE NON-SCIENTIFIC NUMBER ON EACH ROW OF
TEXT ARRAY RHA CONTAINS LEADING ZEROES.

RESULT IS A BOOLEAN ARRAY WITH SHAPE ¯1↓⍴RHA.

FUNCTION DOES NOT CHECK THAT EACH ROW CONTAINS A SINGLE VALID NUMBER (SEE
WHAM/TOKEN∆SINGLE AND ⎕FI). IT DEEMS LEADING ZEROES TO BE '0'S PRECEDED BY
NEITHER A DIGIT NOR A DECIMAL POINT, NOT FOLLOWED BY A SPACE, DECIMAL POINT
OR END OF ROW.

LEADING ZERO EXAMPLES ARE:  -02  000  12,000 ⍝ NOTE LAST

NON-LEADING ZERO EXAMPLES ARE: .00000 0 0.0 -0  2000.0000
```

- -

```
TITLE:        WHAM/TOKEN∆SINGLE.1                                         WHAM/TOKEN∆SINGLE.1

TYPE:         FUNCTION

SUMMARY:      RETURN WHETHER EACH ROW OF AN ARRAY CONTAINS A SINGLE TOKEN

TIMESTAMP:    1984-12-14 08:08:18

CATEGORIES:   20  TERMINAL INPUT/OUTPUT
              21  TERMINAL INPUT

DESCRIPTION:
RETURNS MASK INDICATING ROWS OF TEXT ARRAY RHA CONTAINING A SINGLE TOKEN, THAT
IS A SINGLE SEQUENCE OF NON-BLANKS, OPTIONALLY WITH BLANKS EITHER SIDE. NEITHER
ALL BLANKS NOR ZERO-LENGTH ROWS COUNT AS TOKENS, AS THE NON-BLANK SEQUENCE MUST
BE NON-EMPTY.

THE RESULT HAS SHAPE ¯1↓⍴RHA.

THE FUNCTION IS TYPICALLY USEFUL FOR ENSURING THAT ⎕FI ON THE FLATTENDED ARRAY
WILL PRODUCE A RESULT OF THE EXPECTED LENGTH; AS IN:

      ∇N←QFI MAT;B
[1]    B←TOKEN∆SINGLE MAT ⍝ ROWS WITH ONE TOKEN
[2]    N←B\⎕FI,' ',B⌿MAT ⍝ SEPARATE IN CASE TOKENS FILL ROW
      ∇
```

TITLE: YUDI/FILL.1

TYPE: FUNCTION

SUMMARY: FILLS A CHARACTER STRING WITH A DELIMITER FOR A SPECIFIED NUMBER OF TIMES

TIMESTAMP: 1983-11-29 14:54:35

CATEGORIES:
2 STRUCTURAL TRANSFORMATIONS ⍝ RESHAPING, CATENATING, TRANSPOSING, ETC.
27 TEXT PROCESSING ⍝ E.G. SPELLING CHECKERS
40 SELECTING FROM ARRAYS ⍝ INDEXING, TAKE, COMPRESSION, UNIQUE-ELEMENTS, ETC.

DESCRIPTION:
RESULT←<DEL△NO△POS> FILL <STRING>

<FILL> IS A CHARACTER STRING FILLER. IT CAN BE USED TO FILL A CHARACTER VECTOR AT POSITIONS <POS> WITH A DELIMITER <DEL> FOR AN INTEGER <NO> OF TIMES. THE LEFT ARGUMENT IS A 3 ELEMENT ENCLOSED ARRAY CONTAINING <DEL>, <NO>, AND <POS>. THE RIGHT ARGUMENT IS A CHARACTER VECTOR. THE RESULT IS <NO> CHARACTERS LONGER THAN THE ORIGINAL STRING. THIS FUNCTION IS ⎕IO INDEPENDENT AND ASSUMES ANYTHING OUT OF RANGE IS EITHER TAGGED ON TO THE BEGINNING OR THE END OF THE CHARACTER <STRING>.

```
 EG. (' '⊃3⊃(STRING=' ')/⍳⍴STRING) FILL STRING←'A STRING OF CHARACTERS TO FILL'
A    STRING    OF    CHARACTERS    TO    FILL
```

YUDI