

CLI MACROS

A collection of time-saving general purpose and model macros for AOS and AOS/VS users.



053-000070

Licensed Material

NOTICE

DATA GENERAL CORPORATION (DGC) HAS PREPARED THIS DOCUMENT FOR USE BY DGC PERSONNEL, LICENSEES, AND CUSTOMERS. THE INFORMATION CONTAINED HEREIN IS THE PROPERTY OF DGC AND SHALL NOT BE REPRODUCED IN WHOLE OR IN PART WITHOUT DGC PRIOR WRITTEN APPROVAL.

DGC reserves the right to make changes in specifications and other information contained in this document without prior notice, and the reader should in all cases consult DGC to determine whether any such changes have been made.

THE TERMS AND CONDITIONS GOVERNING THE SALE OF DGC HARDWARE PRODUCTS AND THE LICENSING OF DGC SOFTWARE CONSIST SOLELY OF THOSE SET FORTH IN THE WRITTEN CONTRACTS BETWEEN DGC AND ITS CUSTOMERS. NO REPRESENTATION OR OTHER AFFIRMATION OF FACT CONTAINED IN THIS DOCUMENT INCLUDING BUT NOT LIMITED TO STATEMENTS REGARDING CAPACITY, RESPONSE-TIME PERFORMANCE, SUITABILITY FOR USE OR PERFORMANCE OF PRODUCTS DESCRIBED HEREIN SHALL BE DEEMED TO BE A WARRANTY BY DGC FOR ANY PURPOSE, OR GIVE RISE TO ANY LIABILITY OF DGC WHATSOEVER.

CEO, DASHER, DATAPREP, ECLIPSE, ENTERPRISE, INFOS, MANAP, microNOVA, NOVA, PRESENT, PROXI, SUPERNOVA, SWAT, ECLIPSE MV/4000, ECLIPSE MV/6000, and ECLIPSE MV/8000 are U.S. registered trademarks of Data General Corporation. AZ-TEXT, COMPUCALC, DG/L, DESKTOP GENERATION, DATA GENERAL/One, ECLIPSE MV/10000, GW/4000, GDC/1000, GENAP, MV/UX, REV-UP, TRENDVIEW, DEFINE, SLATE, microECLIPSE, BusiPEN, BusiGEN, BusiTEXT, and XODIAC are U.S. trademarks of Data General Corporation.

Copyright © Data General Corporation, 1986
All Rights Reserved

CLI MACROS

A collection of time-saving general purpose and model macros for AOS and AOS/VS users.



053-000070

Licensed Material

CLI Macros at a Glance Table of Contents

???.CLI	1	DUMP_UTIL.CLI	59	PASCAL.MENU_4.CLI	111
?NET.CLI	3	ENABLE.CLI	60	PASCAL.MENU_5.CLI	112
?CEO.CLI	4	FDEL.CLI	61	PASSWORD.CLI	113
ARISTO.DOWN.CLI	5	FDIR.CLI	62	PCT_FULL.CLI	114
ARISTO.UP.CLI	7	FIND_PID.CLI	63	P_MAKER.CLI	115
BACKUP.CLI	10	FIND_USER.CLI	64	READ_UTIL.CLI	117
BACKUP.DAY.CLI	13	FORMFEED.CLI	65	REVISIONS.CLI	118
BACKUP.FULL.INC.CLI	14	FS.CLI	66	REVISIONS.PROGRAMS	120
BACKUP.SERIES.CLI	15	GET.CLI	67	REVISIONS.SCF	122
BACKUP1.CLI	16	GET.LISTS.CLI	68	REVISIONS.SMF	123
BACKUP2.CLI	23	GET.TIME.STAMP.CLI	69	REVISIONS.SORT.CLI	124
BE.CLI	24	GET_STRING.CLI	71	SETUP.CLI	125
BESTART.CLI	25	HASH.CLI	72	SHUTDOWN.CLI	126
BOMB.CLI	26	HOW.CLI	74	SOFF.CLI	127
BUILD_CEO_USERS.CLI	27	KILL.CLI	75	SPACE.LIST.CLI	128
CCHAR.CLI	28	LA.DOWN.CLI	78	SYSMAP.CLI	129
CCHAR.TXT	29	LA.UP.CLI	80	SYSTEM.CLI	133
CE.CLI	30	LDUS.CLI	86	SYSUP.CLI	134
CEO.AFTER.FULL.DUMP.CLI	31	LINE.PRINTER.CLI	87	SYSUP2.CLI	135
CEO.AFTER.INCR.DUMP.CLI	32	LIST.PROFILES.CLI	88	SYSUP3.CLI	136
CEO.BEFORE.FULL.DUMP.CLI	33	LOAD_UTIL.CLI	89	TESTABORT.CLI	137
CEO.BEFORE.INCR.DUMP.CLI	34	LOCATE.CLI	90	TT.CLI	138
CEO.LOGFILE.TIME.CLI	35	MAKER1.CLI	91	UDDBACKUP.CLI	140
CEO.YES.NO.CLI	36	MASS.PROFILE.CLI	93	USERS.CLI	142
CEO?.CLI	37	MASS.PROFILE1.CLI	95	USERS.IN	143
CERTS.CLI	39	MEGABYTES.CLI	96	WHERE.CLI	144
CHGSEA.CLI	40	MEGS.CLI	97	WHO?.CLI	145
CHG_PASSWORD.CLI	42	MM.CLI	98	YES_OR_NO.CLI	146
CK.CLI	45	MON.CLI	100		
COUNT_FILES.CLI	47	MOVE.ZELDA.CLI	103		
DC.DOWN.CLI	48	NOW.CLI	104		
DC.UP.CLI	50	ON.CLI	105		
DF.CLI	54	PASCAL.MENU.CLI	106		
DIVERT.CLI	55	PASCAL.MENU_1.CLI	108		
DO.CERTS.CLI	56	PASCAL.MENU_2.CLI	109		
DTSTAMP.CLI	58	PASCAL.MENU_3.CLI	110		

CLI Macros At A Glance

Macros are used to save you time and energy. Sometimes macros are created by the user for a specific purpose; other times macros are created that are general purpose in nature, applicable to many different situations. Enclosed in this document are examples of macros that can save you time and energy in creating your own macros.

The examples on the following pages illustrate many features of macros including:

Dummy Arguments

Common Pseudo-Macros

Conditional Pseudo-Macros

Arithmetic Pseudo-Macros

Interactive Macros

File Handling Through Macros

Menu Generation

The macros contained in this document were obtained from many sources. The authors created some of the macros for specific environments to perform specific functions. Therefore, use the macros as illustrations. Some modifications may be necessary to the macros contained in this document before they will run on your system.

Please note that all the macros contained in this document were written prior to AOS/VS REV 7. Some modifications may be needed to the macros to take full advantage of the changes, e.g., [!PIDS] should be substituted whenever active PID numbers need to be obtained. Consult the current system documentation for more details.

Look at the macros on the following pages. See how the authors have used the many different macro features. Then use the macros in this document as a reference in creating your own.

```
comment +-----+
comment |                ??? .CLI                |
comment +-----+
```

```
[!equal,large,comment]-----+
|
|   This macro displays the disk space utilization for each CPD under
|   the root directory.
|
+-----+-----[!end]
```

```
PUSH
SUPERUSER ON
WRITE [!ASCII 214 211]Disk Space Utilization - Root Directory
WRITE
DIR :
SP/V/2=I :
SP/V/2=I (A B C D E F G H I J K L M N O P Q R S T U V W X Y Z ? _ .)+
WRITE
@NULL [!READ [!ASCII 211]Please Press 'NEW-LINE' To Continue! ]
WRITE [!ASCII 214 211]Disk Space Utilization - Users 'A' Thru 'F'
WRITE
DIR :UDD
SP/V/2=I (A B C D E F)+
WRITE
@NULL [!READ [!ASCII 211]Please Press 'NEW-LINE' To Continue! ]
WRITE [!ASCII 214 211]Disk Space Utilization - Users 'G' Thru 'L'
WRITE
SP/V/2=I (G H I J K L)+
WRITE
@NULL [!READ [!ASCII 211]Please Press 'NEW-LINE' To Continue! ]
WRITE [!ASCII 214 211]Disk Space Utilization - Users 'M' Thru 'R'
WRITE
SP/V/2=I (M N O P Q R)+
WRITE
@NULL [!READ [!ASCII 211]Please Press 'NEW-LINE' To Continue! ]
WRITE [!ASCII 214 211]Disk Space Utilization - Users 'S' Thru 'Z'
WRITE
SP/V/2=I (S T U V W X Y Z 0 1 2 3 4 5 6 7 8 9 ? _ .)+
WRITE
@NULL [!READ [!ASCII 211]Please Press 'NEW-LINE' To Continue! ]
```

```
WRITE [!ASCII 214 211]Disk Space Utilization - User '[!USER]'  
WRITE  
DIR :UDD:[!USER]  
SP/V/2=I (A B C D E F G H I J K L M N O P Q R S T U V W X Y Z ? _ .)+  
WRITE  
@NULL [!READ [!ASCII 211]Please Press 'NEW-LINE' To Continue! ]  
WRITE [!ASCII 214]  
POP
```

```
comment +-----+
comment |           ?NET.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This submacro is called from CK.CLI. It lists all the XODIAC
| processes on the system.
|
+-----[!end]
```

```
[!NEQUAL,%1%,]
  WHO%0/% <%1%>:OP:NETOP
  WHO%0/% <%1%>:OP:X25[!NEQ,%1%,[!HOST]]_LMGR[!END]
  WHO%0/% <%1%>:OP:SVTA
  WHO%0/% <%1%>:OP:FTA
  WHO%0/% <%1%>:OP:RMA
[!ELSE]
  WHO%0/% OP:NETOP
  WHO%0/% OP:X25_LMGR
  WHO%0/% OP:SVTA
  WHO%0/% OP:FTA
  WHO%0/% OP:RMA
[!END]
```



```
comment +-----+
comment |                ?CEO.CLI                |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This submacro is called from CK.CLI. It lists all the CEO
| processes on the system.
|
+-----+ [!end]
```

```
PUSH
PROMPT POP
[!NEQUAL,%1%,]
  SQUEEZE ON
  WHO%0/% <%1%>:CEO_MGR:CEO <POA CSA QMA FSA>
  WHO%0/%/2=IGN <%1%>:CEO_MGR:CEO_LOG
[!ELSE]
  WHO%0/% CEO_MGR:CEO <POA CSA QMA FSA>
  WHO%0/%/2=IGN CEO_MGR:CEO_LOG
[!END]
```

```
comment +-----+
comment |          ARISTO.DOWN.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This non-interactive driver macro brings down the processes
| brought up by ARISTO.UP.CLI.
|
| This DOWN macro is system dependent. Submacros and programs are
| executed that may not be on every system. Devices are stopped that
| may not be on your system.
|
| Remember, this macro is an example of an DOWN macro and would be
| modified for your system.
|
+-----[!end]
```

```
[!EQUAL, ([!UNE,002,[!PID]]ERROR[!END]%0\BACKUP%%-%), ( )]
```

```
BROADCAST COMMENCING SYSTEM SHUTDOWN
CONTROL @EXEC DISABLE/ALL
```

```
CEO.SYSTEM STOP           ; System supplied macros.
SECRETARY_DOWN
DOWN.NETWORK
```

```
[!EQ, ,%0/BACKUP%]
  TERMINATE/2=ERROR OP:EXEC
[!END]
```

```
INFOS_DOWN               ; System supplied macros.
DB.DOWN
COMMENT TERMINATE/2=I OP:MINISAM
TERMINATE/2=I OP:SQL(1,2)
```

```
[!EQ,,%0/BACKUP%]  
  PUSH  
  SUPERUSER ON  
  DIRECTORY :  
  SEA/K  
  RELEASE/2=IGN UDD          ; System specific disk structure.  
  RELEASE/2=IGN/1=IGN UTIL  
  POP  
[!END]  
  
WRITE  
CHECKTERMS  
WRITE  
  
[!EQ,,%0/BACKUP%]  
  BYE  
[!END]  
  
[!ELSE]  
  WRITE *ERROR*  
  WRITE %0\% IS VALID ONLY WHEN INVOKED BY THE ROOT CLI  
  WRITE REQUIRED ARGUMENTS: NONE  
  WRITE OPTIONAL SWITCHES: ,,/BACKUP  
[!END]
```

```

comment +-----+
comment |          ARISTO.UP.CLI          |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This non-interactive driver macro brings up system processes on
| ARISTO.
|
| This UP macro is system dependent.  Submacros and programs are
| executed that may not be on every system.  Devices are started that
| may not be on your system.
|
| Remember, this macro is an example of an UP macro and would be
| modified for your system.
|
+-----[!end]

```

```

[!EQUAL, ([!UNE,002,[!PID]]ERROR[!END]&
%0\BACKUP\NOCONSOLES\NOINFOS\NOBMS\NOSQL\NONET\NOCEO\NOLOCK\UNIX%%-%),()]
[!EQ,,%0/BACKUP%]
  SUPERUSER ON
  ACL @MTB- OP,,-,WARE
  INIT/1=ign/2=ign @DPJ1
  INIT/1=IGN/2=IGN @DPJ2
  SUPERUSER OFF

  STRING PID[!ASC 240]2
  PROMPT HOST STRING DIRECTORY CHECKTERM
  SEARCHLIST :UTIL:MACROS :UTIL :UTIL:CEO_DIR :
  SYSID Aristotle/[!ASC 216]VS[!ASC 234 217] - Customer Ed.[!ASC 235]

  WRITE Starting QCMP now...
  write
  XEQ QCMP/YES
  WRITE QCMP has finished.

```

```
CHAR/DEF/HARDCOPY/LPP=66/CPL=80/BAUD=1200/CHARLEN=8/ON/ST/NAS/OFC/OFF/WRP &
/ULC @CON52;COMMENT THIS IS TO RUN THE LQP AT 20mA
```

```
CHAR/DEF/BAUD=1200 @CON(2 3);COMMENT These are the LQPs.
CHAR/DEF/CHARLEN=8/605X @CON56; COMMENT This is for Video Monitor.
CHAR/DEF/CHARLEN=8 @CON7 ;COMMENT This is for the Scantron Reader.
CHAR/DEF/CHARLEN=8/6130 @CON47; COMMENT G300 IN RESOURCE CTR
```

```
PROCESS/DEFAULT/DIRECTORY=@/NAME=EXEC/PRI=1 EXEC
:NET:UTIL:WAIT_FOR_PORT @EXEC
CONTROL @EXEC PROMPTS OFF
```

```
STARTSFLQP          ; These macros start the printers. See the
STARTLQP1           ; appropriate reference materials on Queue
STARTLTPB           ; management to write your own.
[!END]
```

```
[!EQ,,%0/NOINFOS%]
PUSH
```

```
    INFOS_UP          ; Supplied with INFOS_II.
    POP
[!END]
```

```
[!EQ,,%0/NODBMS%]
    DB.UP/DELETE      ; Supplied with DBMS.
[!END]
```

```
[!EQ,,%0/NOSQL%]
    UP.SQL            ; Supplied with SQL.
[!END]
```

```
[!EQ,,%0/NONET%]
    UP.NETWORK/UP     ; Supplied with XODIAC.
[!END]
```

```
[!EQ,,%0/NOCEO%]
    :UTIL:INFOS:IRECOVER/RESTART/DELETE :CEO_FILES:CEO_INDEX.DB
    CEO.SYSTEM START CHECKPOINT
    SECRETARY_UP      ; System specific macro.
[!END]
```

```

[!EQ,,%0/NOCONSOLES%]
CONTROL @EXEC ENABLE/ALL
[!END]

CONTROL @EXEC CONTINUE (1 2)
CONTROL @EXEC SILENCE
CONTROL @EXEC PROMPTS ON

[!EQ,,%0/NOLOCK%]
  POPO                ; Recursive macro that returns to level 0.
  DIR :
  OFF                 ; Turns off superuser.
  X LOCK_CLI PROMPT HOST DIR
[!END]

[!ELSE]
WRITE *ERROR*
WRITE %0\% IS VALID WHEN INVOKED BY THE ROOT CLI
WRITE OPTIONAL ARGUMENTS: NONE
WRITE OPTIONAL SWITCHES: /NOCONSOLES,,/NOINFOS,,/NOBMS,,/NONET,,/NOCEO,, & /N
OLOCK,,/NOSQL,,/UNIX,,/BACKUP
[!END]

```

```
comment +-----+
comment |          BACKUP.CLI          |
comment +-----+
```

```
[!equal,large,comment]_____+
```

This macro, a heavily modified version of CEO.BACKUP<,1>.CLI, does labeled multivolume tapes. Volids are: TSDV
where: T(ype) is either F(ull) or I(ncr),
S(eries) is A or B,
D(ay) is either 0 for F, or 1 through 9 for I, and
V(olume) is 1 - 9. For F, DV together can be 01 - 99.

This allows rotating two series of tapes, FA0v and IA1..9v, and FB0v and IB1..9v. A full backup can be extended through 99 volumes, and incremental through 9 volumes.

Each tapeset will have two dump files:
CEO_FILES:# CEO_MAIL:#

Arguments are required for batch, optional otherwise:

- 1 F or I
- 2 A or B
- 3 (if I) a digit 1 through 9

Examples:

```
QBATCH BACKUP F A      (full backup)
QBATCH BACKUP I A 1    (first incremental backup)
QBATCH BACKUP I A 2    (second incremental backup)
etc.
```

As shown, the macro can be QBATCHED.

This macro uses the following submacros:

BACKUP.FULL.INC.	BACKUP.SERIES
BACKUP.DAY	CEO.YES.NO
CEO.BEFORE.INCR.DUMP	BACKUP2
CEO.AFTER.INCR.DUMP	CEO.AFTER.FULL.DUMP
BACKUP1	BACKUP2

```
[!end]+
```

```

[!EQUAL, ([!CON]),]
  WRITE
  WRITE You can't run the BACKUP macro from the master console.
  WRITE Please go to a user console and log on as OP.
  WRITE
[!ELSE]
  [!NEQUAL, (ON), ([!OPERATOR])]
    WRITE
    WRITE You need to turn operator ON.
    WRITE Go to the master console and type
    WRITE,,,,CX OPERATOR ON
    WRITE then come back and try again.
    WRITE
[!ELSE]

[!EQUAL, [!USER], OP]
  PUSH
  PROMPT POP
  SUPERUSER ON
  CLASS1 WARNING
  CLASS2 WARNING
  WRITE
  WRITE BACKUP macro for labeled tapes
  WRITE -----
  WRITE Now bringing down CEO.
  WRITE Don't restart CEO until the backup is done.
  WRITE
  CEO.SYSTEM.CLI STOP
  BACKUP1.CLI %1-%
  [!EQU, [!STR], OK]

```


COMMENT Moved notification to BACKUP1.CLI

```
[!ELSE]
  WRITE
  [!EQU,[!STR],NO]
  WRITE No dump was done.
  [!ELSE]
  WRITE Problems occurred with the backup.
  SEND 2 Problems occurred with the backup.
  WRITE Please correct the problem before trying again.
  SEND 2 Please correct the problem before trying again.
  [!END]
  WRITE
[!END]
POP
[!ELSE]
  WRITE
  WRITE You must be logged on as username 'OP'
  WRITE in order to run the BACKUP macro.
  WRITE
[!END]

[!END]
POP
POP
[!END]
```

```
comment +-----+
comment |          BACKUP.DAY.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This CEO backup macro is called by BACKUP1.CLI (driver macro is
| BACKUP.CLI). It reads the day in the series of labeled tape.
|
+-----+ [!end]
```

```
VAR9 0
[!EQUAL,(),(%1%)]
  WRITE Which day in the incremental series?
  VAR9/1=I/2=I [!READ { 1 2 ... },]
[!ELSE]
  VAR9/1=I/2=I %1%
[!END]

[!UGT,[!VAR9],9]
  VAR9 0
[!END]
[!UEQ,[!VAR9],0]
  WRITE Please enter a digit 1 through 9.
  %0%
[!END]
```

```
comment +-----+
comment |          BACKUP.FULL.INC.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This CEO backup macro is called from BACKUP1.CLI (driver macro is
| BACKUP.CLI). It asks whether this backup is to be full or
| incremental.
|
+-----[!end]
```

```
[!EQUAL,(),(%/ERR%1%)]
WRITE
WRITE You can either do a full or incremental backup.
WRITE Incremental backup will only backup those files that
WRITE were modified since the previous full dump.
WRITE
[!END]
[!EQUAL,(),(%1%)]
WRITE Would you like full or incremental backup?
STRING [!READ F/I,]
[!ELSE]
STRING %1%
[!END]

[!NEQUAL,([!STRING]),(F)]
[!NEQUAL,([!STRING]),(I)]
STRING BAD
[!END]
[!END]
[!EQUAL,([!STRING]),(BAD)]
WRITE Please answer 'F' or 'I'.
%0%/ERR
[!END]
```

```
comment +-----+
comment |          BACKUP.SERIES.CLI          |
comment +-----+
```

```
[!equal, large, comment]-----+
|
| This CEO backup macro is called from BACKUP1.CLI (driver macro is
| BACKUP.CLI). It reads the tape series for labeled tape.
|
+-----[!end]
```

```
[!EQUAL, (), (%1%)]
  WRITE Which tape series - A or B?
  STRING [!READ A/B,]
[!ELSE]
  STRING %1%
[!END]

[!NEQ, ([!STR]), (A)]
  [!NEQ, ([!STR]), (B)]
  STRING BAD
  [!END]
[!END]
[!EQUAL, ([!STRING]), (BAD)]
  WRITE Please type 'A' or 'B'.
  %0%
[!END]
```

```

comment +-----+
comment |                BACKUP1.CLI                |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This CEO backup macro is called from BACKUP.CLI.  It has already
| been established that the user is OP and CEO is stopped.
|
| It uses the following submacros:
|     BACKUP.FULL.INC.          BACKUP.SERIES
|     BACKUP.DAY                CEO.YES.NO
|     CEO.BEFORE.INCR.DUMP     BACKUP2
|     CEO.AFTER.INCR.DUMP      CEO.AFTER.FULL.DUMP
|
+-----+ [!end]

```

```

WRITE
PUSH
PROMPT POP
SUPERUSER ON
[!NEQU,[!PATH,:CEO_FILES:CEO_LOG:CURR_LOGFILE],]
    VAR4 1
[!ELSE]
    VAR4 0
[!END]
    [!EQU,[!LOGON],BATCH]
        VAR3 0
        STRING %1%
        [!EQU,[!STR],I]
            WRI INCREMENTAL DUMP of databases -- files modified since,&
                [:CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP].
            VAR1 10
        [!ELSE]
            [!EQU,[!STR],F]
                WRI FULL DUMP of databases -- last full dump was made,&
                    [:CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP].
                VAR1 11
        [!ELSE]
            WRI Argument 1 should be: I - incremental dump, or
            WRI,,,,,,,,,,,,,,,,,,,,,F - full dump.
            VAR3 1
        [!END]

```

```

[!END]
STRING %2%
[!EQU,[!STR],A]
    VAR0 10
[!ELSE]
    [!EQU,[!STR],B]
        VAR0 11
    [!ELSE]
        WRI Argument 2 is should be: A or B
        VAR3 1
    [!END]
[!END]
VAR9 0
[!EQU,[!VAR1],10]
    VAR9 0
    VAR9/1=I/2=I %3%
    [!UGT,[!VAR9],9]
        VAR9 0
    [!END]
    [!UEQ,[!VAR9],0]
        WRI Argument 3 should be a digit 1 through 9
        VAR3 1
    [!END]
[!END]
[!EQU,[!VAR3],0]
    COMMENT No errors in argument.
    [!EQU,[!VAR1],10]
        STRING I
    [!ELSE]
        STRING F
    [!END]
    [!EQU,[!VAR0],10]
        STRING [!STR]A[!VAR9]
    [!ELSE]
        STRING [!STR]B[!VAR9]
    [!END]
    WRI Volids are [!STR]{0 1 2 ...}

```

```

[!ELSE]
  WRITE
  WRITE Invalid arguments - backup cancelled.
  WRITE Proceeding with IVERIFY anyway.....
  WRITE
[!END]
COMMENT --- ALWAYS VERIFY INFOS FILE.
X IVERIFY :CEO_FILES:CEO_INDEX.DB
WRITE
WRITE If IVERIFY found problems, you should consult your
WRITE Data General representative.
COMMENT --- ALWAYS CONTINUE WITH DUMP BUT REALLY SHOULD CHECK RESULTS.
[!ELSE]

```

```

COMMENT Logon is CONSOLE, interactive dialog.
[!EQ,%0/ERR%,]
  CLASS1 WARNING
  CLASS2 WARNING
  [!NEQUAL,[!PATH :CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP],]
  WRITE
  WRITE The last full dump was made on,&
  [ :CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP]
  WRITE

```

```

COMMENT Ask user whether full or incremental - F or I
  BACKUP.FULL.INC %1%

  [!EQ,[!STR],I]
  VAR1 10
  [!ELSE]
  VAR1 11
  [!END]
[!ELSE]

```

```

    CRE :CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP
    ACL :CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP OP OWARE + R
    [!EQU,[!VAR4],0]
        DEL/2=IGN :CEO_FILES:CEO_LOG:OLD_LOGS_LIST
        CRE :CEO_FILES:CEO_LOG:OLD_LOGS_LIST
        ACL :CEO_FILES:CEO_LOG:OLD_LOGS_LIST OP OWARE + R
    [!END]
    WRITE This macro will do a full dump of the databases
    WRITE since it has not been done earlier.
    WRITE
    VAR1 11
    [!END]
[!END]
WRITE

COMMENT Ask user which series - A or B
BACKUP.SERIES %2%

    [!EQ,[!STR],A]
        VAR0 10
    [!ELSE]
        VAR0 11
    [!END]
    VAR9 0
    [!EQU,[!VAR1],10]
        WRITE
        BACKUP.DAY %3%

    [!END]

COMMENT
COMMENT --- ARGUMENTS ACCEPTED
COMMENT

COMMENT --- OPTIONALLY VERIFY INFOS FILE.
    WRITE
    WRITE Even when INFOS runs in differential file mode it is a
    WRITE good idea to verify the validity of the file periodically.
    WRITE

```



```

WRITE Would you like to verify the CEO_INDEX before the backup?
PUSH ; PROMPT POP
:UTIL:CEO_DIR:CEO.YES.NO
[!EQ,[!STR],YES]
    [!EQU,[!PATH IVERIFY.PR],]
        WRITE IVERIFY cannot be found in your searchlist.
        SEA [!SEA] [!READ In what directory is IVERIFY.PR? ]
    [!END]
X IVERIFY :CEO_FILES:CEO_INDEX.DB
WRITE
WRITE If IVERIFY found problems, you should consult your
WRITE Data General representative.
WRITE
[!END]
POP
VAR3 0
WRITE
WRITE You have selected:
WRITE
WRITE ,,,[!EQ,[!VAR1],10]INCREMENTAL[!ELSE]FULL[!END] backup
WRITE ,,,Series [!EQ,[!VAR0],10]A[!ELSE]B[!END]
[!EQ,[!VAR1],10]WRITE ,,,Day [!VAR9][!END]
WRITE
WRITE Would you like to proceed with the backup?
:UTIL:CEO_DIR:CEO.YES.NO
[!EQU,[!STR],NO]
    VAR3 1
[!ELSE]
    [!EQU,[!VAR1],10]
        STRING I
    [!ELSE]
        STRING F
    [!END]
    [!EQU,[!VAR0],10]
        STRING [!STR]A[!VAR9]
    [!ELSE]
        STRING [!STR]B[!VAR9]
    [!END]
[!END]
[!END]

```

```

COMMENT
COMMENT   IF NO ERROR SO FAR - EXECUTE DUMP.
COMMENT

[!EQU,[!VAR3],0]
  WRITE
  WRITE Starting the dump now.....
  WRITE

  [!EQ,[!VAR1],10]
  COMMENT --- INCREMENTAL DUMP ---
  [!EQU,[!VAR4],1]
  CEO.BEFORE.INCR.DUMP.CLI
  [!END]

COMMENT   Call BACKUP2 with /I for incremental and supply date.
          BACKUP2/I[!EQ,[!LOGON],BATCH]/L[!END] [!STR] &
          [[!PATH,:CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP]]

          [!EQU,[!VAR4],1]

COMMENT   If DUMP_II bombed, string will be non-empty (*ABORT*)
COMMENT   and we don't want to call CEO.AFTER.FULL.DUMP.
          [!EQU,(],[!STR]])
          CEO.AFTER.INCR.DUMP.CLI
          [!END]
          [!END]
[!ELSE]
  COMMENT --- FULL DUMP ---
  [!EQU,[!VAR4],1]
  CEO.BEFORE.FULL.DUMP.CLI
  [!END]

COMMENT   Call BACKUP2 without /I for full dump.
          BACKUP2[!EQ,[!LOGON],BATCH]/L[!END] [!STR]

          [!EQU,[!VAR4],1]

```

```

COMMENT      If DUMP_II bombed, string will be non-empty (*ABORT*)
COMMENT      and we don't want to call CEO.AFTER.INC.DUMP.
              [!EQU,(),([!STR])]
              CEO.AFTER.FULL.DUMP.CLI
              [!END]
[!ELSE]
              DEL/2=IGN :CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP
              WRI/L=:CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP &
[!DATE]:[!TIME][!ASCII,46]
              ACL :CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP OP OWARE + R
              [!END]
[!END]
[!EQUAL,(),([!STRING])]
              WRITE
              WRITE      Done with backup.
              WRITE
              WRITE      User OP - as PID 2 - can now start the CEO system.
              SEND 2 User OP - as PID 2 - can now start the CEO system.
              STRING OK
[!END]
[!END]

```

```

comment +-----+
comment |          BACKUP2.CLI          |
comment +-----+

[!equal,large,comment]-----+
|
| This CEO backup macro is called from BACKUP1.CLI.  It does the
| actual MOUNT and DUMP.
|
+-----[!end]

```

```

SUPERUSER ON
DELETE/2=IGNORE :UDD:[!USERNAME]:BACKUP<,.LIST>

```

```

[!NEQ,(),(%/L%)]
  LISTFILE :UDD:[!USERNAME]:BACKUP.LIST
  WRITE The full listing of files backed up is in [!list].
  WRITE
[!END]

```

```

MOUNT/EXTEND/VOLID=%1%1/VOLID=%1%2/VOLID=%1%3&
/VOLID=%1%4/VOLID=%1%5/VOLID=%1%6/VOLID=%1%6/VOLID=%1%7&
/VOLID=%1%8/VOLID=%1%9 BACKUP &
[!EQ,(),(%0/I%)]Full[!ELSE]Incremental[!END] Backup

```

```

PRTYPE [!PID] PREEMPTIBLE

```

```

DIR :
XEQ/S DUMP_II/V/BUFFER=8192/RETAIN=0%/L%&
[!NEQUAL,(),(%0/I%)]/AFTER/TLM=%2%[!END]&
:UDD:[!USERNAME]:BACKUP:ROOT &
CEO_FILES:# CEO_MAIL:#

```

```

DISMOUNT/1=W/2=W BACKUP
LISTFILE/K

```

```
comment +-----+
comment |                BE.CLI                |
comment +-----+
```

```
[!equal,large,comment]-----+
|
|   This macro expects, as a first argument, the Username of the
|   person you would like to become.  You must have Change User-
|   name privilege to use this.
|
|   It calls the macro BESTART.CLI so that the new Username will
|   blink as a prompt.
|
+-----[!end]
```

```
[!NE,%1%,[!USER]]
WRITE
WRITE,,,When CLI comes up, you will be %1% !!
WRITE
WRITE,,,,,Type 'BYE' to become yourself again
WRITE
PROC/DEF/BLOCK/IOC/SONS/NAME=[!USER]/USER=%1%/DIR=:UDD:%1% :CLI BESTART
[!ELSE]
WRITE
WRITE,,,[!ASC 207,207,207,207]YOU ALREADY ARE %1% !!
WRITE
[!END]
```

```
comment +-----+
comment |           BESTART.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
| This macro is called by BE.CLI.  It changes the PREFIX to be |
| a blinking [!USER]                                           |
+-----[!end]
```

```
PREFIX [!ASC 216][!USER]:[!ASC 217]
```

```
comment +-----+
comment |           BOMB.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
|   This interactive macro terminates users by their PID number.
|   You must have Superprocess to use it.
|
+-----[!end]
```

```
PUSH
PROMPT POP
SUPERPROCESS ON
VARO [!READ Enter PID number of user to be terminated:,,]
TERMINATE [!VARO]
POP
```

```
comment +-----+
comment |          BUILD_CEO_USERS.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro executes PREDITOR to create a CEO user, using the
| existing profile CEO.USER.  It expects, as a first argument, the
| username to be created.  The users' password will be the same as
| their username.
|
| Assumptions:  CEO.USER is an existing profile; initial IPC will
| be :MACROS:CHAIN_CEO.CLI.
|
| PRIVILEGES:  SUPERUSER
|
| CAUTION: This macro is REV specific.  Changes are needed as the
| PREDITOR dialog changes.
|
|-----[!end]
```

```
X/M PREDITOR
U
CEO.USER
C
%1%
Y
%1%
Y
:MACROS:CHAIN_CEO.CLI
```



```
comment +-----+
comment |           CCHAR.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
|   This macro displays the ASCII codes for all the console characters.
|   This is very helpful to format the [!ASCII ] pseudo-macro.
|   CCHAR.CLI needs file CCHAR.TXT
|
+-----[!end]
```

```
TYPE/1=W/2=W CCHAR.TXT
```

CCHAR.TXT

This is the text file used with the CCHAR.CLI macro.

*** Note: This box is not part of the text file.***

203	Enable Blink	231	Cursor Left	260-271	0-9
204	Disable Blink	232	Cursor Down	272	:
205	Start Blink	234	Start Dim	273	;
206	Stop Blink	235	Stop Dim	274	<
207	Bell	240	Space	275	=
210	Home	241	!	276	>
211	Tab	242	"	277	?
212	New-Line	243	#	300	@
213	Erase EOL	244	\$	301-332	A-Z
214	Erase Page	245	%	333	[
215	Carriage Return	246	&	334	\
216	Start Blink	247	'	335]
217	Stop Blink	250	(341-372	a-z
220, Col, Row	Set Cursor Pos *	251)	373	{
222	Enable Roll	252	*	374	
223	Disable Roll	253	+	375	}
224	Start Underline	254	,	376	~
225	Stop Underline	255	-		
227	Cursor Up	256	.		
230	Cursor Right	257	/		

* 0,0 is Upper Left

```

comment +-----+
comment |           CE.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro will pass your Control @EXEC command onto a process
| of the name OP. After the command has been processed, the OP
| process will be terminated.
|
| The user must have change username and superuser privilege.
|
+-----[!end]

```

```

[!EQ,[!USERNAME],OP]
CONTROL @EXEC %-%
[!ELSE]
[!EQ,%0/PASS1%,]
DEL/1=IGN/2=IGN CE.[!PID].TMP
CRE CE.[!PID].TMP
ACL CE.[!PID].TMP +,OWARE
WRITE/L=CE.[!PID].TMP ON
WRITE/L=CE.[!PID].TMP CONTROL @EXEC %1-%
WRITE/L=CE.[!PID].TMP PAUSE 2.0
WRITE Passing your command to OP...
%0\%/PASS1 CE.[!PID].TMP
[!ELSE]
PUSH
PROMPT POP
SUPERUSER ON
PROC/DEF/IOC/INPUT=%1%/OUTPUT=@CONSOLE/BLOCK/USERNAME=OP :CLI.PR
DEL/1=IGN/2=IGN %1%
POP
[!END]
[!END]

```

```

comment +-----+
comment |          CEO.AFTER.FULL.DUMP.CLI          |
comment +-----+

```

```

[!equal,large,comment]-----+
| This CEO backup macro is called from BACKUP1.CLI (driver macro is |
| BACKUP.CLI). |
+-----[!end]

```

```

DEL/2=IGN :CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP
WRITE/L=:CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP &
          [!DATE]:[!TIME][!ASCII,46]
ACL :CEO_FILES:CEO_LOG:CEO_LAST_FULL_DUMP OP OWARE + RE
DEL/2=IGN :CEO_FILES:CEO_LOG:DUMPED_OLD_LOGS_LIST
DEL/2=IGN :CEO_FILES:CEO_LOG:OLD_LOGS_LIST
WRITE/L=:CEO_FILES:CEO_LOG:OLD_LOGS_LIST,&
        :CEO_FILES:CEO_LOG:OLD_LOG_[ :CEO_FILES:CEO_LOG:?.BACKUP.[!PID].TMP]
ACL :CEO_FILES:CEO_LOG:OLD_LOGS_LIST OP OWARE + R
PUSH
DIR :CEO_FILES:CEO_LOG
DEL/2=IGN OLD_LOG +\OLD_LOG_[ :CEO_FILES:CEO_LOG:?.BACKUP.[!PID].TMP]
DEL/2=IGN :CEO_FILES:CEO_LOG:?.BACKUP.[!PID].TMP
POP

```

```
comment +-----+
comment |           CEO.AFTER.INCR.DUMP.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This CEO backup macro is called from BACKUP1.CLI (driver macro is |
| BACKUP.CLI). |
+-----[!end]
```

```
DEL/2=IGN :CEO_FILES:CEO_LOG:?.BACKUP.[!PID].TMP
```

```
comment +-----+
comment |           CEO.BEFORE.FULL.DUMP.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
| This CEO backup macro is called from BACKUP1.CLI (driver macro is |
| BACKUP.CLI) |
+-----[!end]
```

```
CEO.LOGFILE.TIME
PERMAN :CEO_FILES:CEO_LOG:Curr_LOGFILE OFF
RENAME :CEO_FILES:CEO_LOG:Curr_LOGFILE,&
        OLD_LOG_[:CEO_FILES:CEO_LOG:?.BACKUP.[!PID].TMP]
DEL/2=IGN :CEO_FILES:CEO_LOG:DUMPED_OLD_LOGS_LIST
REN/1=WARN :CEO_FILES:CEO_LOG:OLD_LOGS_LIST DUMPED_OLD_LOGS_LIST
```

```
comment +-----+
comment |          CEO.BEFORE.INCR.DUMP.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
| This CEO backup macro is called from BACKUP1.CLI (driver macro is |
| BACKUP.CLI). |
+-----[!end]
```

```
CEO.LOGFILE.TIME
PERMAN :CEO_FILES:CEO_LOG:Curr_LOGFILE OFF
RENAME :CEO_FILES:CEO_LOG:Curr_LOGFILE,&
        OLD_LOG_[ :CEO_FILES:CEO_LOG:?.BACKUP.[!PID].TMP ]
WRITE/L=:CEO_FILES:CEO_LOG:OLD_LOGS_LIST,&
        :CEO_FILES:CEO_LOG:OLD_LOG_[ :CEO_FILES:CEO_LOG:?.BACKUP.[!PID].TMP ]
```

```
comment +-----+
comment |           CEO.LOGFILE.TIME.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
| This CEO backup macro is called by CEO.BEFORE.INCR.DUMP (driver |
| macro is BACKUP.CLI)                                           |
+-----[!end]
```

```
PUSH
CEO.FIL.CRE.TIME    [!PATH :CEO_FILES:CEO_LOG:CURR_LOGFILE]
GET.TIME.STAMP.CLI [!STR]
DEL/2=IGN :CEO_FILES:CEO_LOG:?.BACKUP.[!PID].TMP
WR/L=:CEO_FILES:CEO_LOG:?.BACKUP.[!PID].TMP [!STR]
POP
```



```
comment +-----+
comment |          CEO.YES.NO.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro sets the STRING to YES or NO depending on a user's
| answer to a question in a driver macro.
|
+-----[!end]
```

```
STRING/1=IGNORE/2=IGNORE [!READ YES/NO ]
[!EQUAL, ([!STRING]), (Y) ]
  STRING YES
[!END]
[!EQUAL, ([!STRING]), (N) ]
  STRING NO
[!END]
[!NEQUAL, ([!STRING]), (YES) ]
[!NEQUAL, ([!STRING]), (NO) ]
  WRITE Please respond 'YES' or 'NO'
  :UTIL:CEO_DIR:CEO.YES.NO
[!END]
[!END]
```

```

comment +-----+
comment |                CEO?.CLI                |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro lists all current CEO Server processes, and their
| tree structure.
|
+-----+ [!end]

```

```

PUSH
PROMPT POP LEVEL
CLASS(1,2) WARNING
WRITE [!ASCII 214]
VAR0 0
[!NEQUAL,[!PATH @CEO_CSA],]VAR0 [!UADD,[!VAR0],1][!END]
[!NEQUAL,[!PATH @CEO_FSA],]VAR0 [!UADD,[!VAR0],1][!END]
[!NEQUAL,[!PATH @CEO_POA],]VAR0 [!UADD,[!VAR0],1][!END]
[!NEQUAL,[!PATH @CEO_QMA],]VAR0 [!UADD,[!VAR0],1][!END]
[!EQUAL,[!VAR0],4]
WRITE,,,,,,,,,All components of CEO are functional
[!ELSE]
WRITE,,,,,,,,,CEO is not completely functional
WRITE
[!EQUAL,[!VAR0],0]
WRITE,,,,,,,,,No CEO servers are running
[!ELSE]
WRITE,,,,,,,,,Only [!VAR0] servers are running
[!END]
[!END]
[!NEQUAL,[!VAR0],0]
WRITE
WRITE,,,,,,,,,List of CEO Servers
WRITE
WHO/1=I/2=I CEO_MGR:CEO_(CSA,FSA,POA,QMA,LOG) OP:CHECKPOINT
WRITE
WRITE,,,,,,,,,CEO process structure
WRITE
TREE/1=I/2=I CEO_MGR:CEO_(CSA,FSA,POA,QMA,LOG) OP:CHECKPOINT

```

```
WRITE  
WRITE,,,,,,,,,,,,,,,,,,,,,End of Listing  
WRITE  
[!END]  
POP
```

```
comment +-----+
comment |           CERTS.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
|   This driver macro creates a process with the name of OP to run
|   a menu to change paper form between (8-1/2" X 11") to Certificate
|   forms.
|
|   Assumptions:  Executed from user console;
|
|   Submacros:   DO.CERTS.CLI
|
|   PRIVILEGES:  CHANGE USERNAME
|
+-----[!end]
```

```
PROC/DEF/BLOCK/IOC/SONS/NAME=[!USER]/USER=OP/DIR=:UDD:OP  :CLI DO.CERTS
bye
```

```

comment +-----+
comment |           CHGSEA.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro will either remove one directory from your searchlist,
| or add one or more directories to the beginning of your searchlist.
|
| /ADD to add a directory that is %1-% to your searchlist
|     e.g., ) CHGSEA/ADD :UTIL:PRESENT
|
| To delete one directory use LIST without switches and pass in as
|     e.g.  ) CHGSEA :UTIL:COBOL
|
+-----[!end]

```

```

[!EQ,%1%,]
    WRITE This macro requires as an argument a directory to be
    WRITE added or removed from your searchlist. To add a
    WRITE directory to the beginning of your searchlist use
    WRITE format: CHGSEA/ADD directory-to-be-added
    WRITE To delete a directory use the following format:
    WRITE           CHGSEA directory-to-be-deleted
    WRITE

```

```

[!ELSE]
COMMENT On pass 1 we will recurse appending the current searchlist
COMMENT to the arguments

```

```

[!EQ,%0/ADD%,/ADD]
    SEA %1-% [!SEA]
    WRITE Your current searchlist is [!SEA]

```

```

[!ELSE]
[!NEQ,%0/PASS2%,/PASS2]
    [!NEQ,%0/PASS3%,/PASS3]
        VAR9 0
        [!PATH CHGSEA.CLI]/PASS2 %1% [!SEA]
    [!END]

```

```

[!END]
COMMENT On pass2 null the searchlist and recurse with all args

```

```

[!EQ,%0/PASS2%,/PASS2]
    SEARCH/K
    %0\%/PASS3 %1-%

```

```

[!END]

```

```

COMMENT On pass3 check to see if argument one is on the searchlist
[!EQ,%0/PASS3%,/PASS3]
  COMMENT Check to see if we've reached the end of the searchlist
  [!NEQ,%2%,]
    [!EQ,%1%,%2%]
      COMMENT Here we've found it!
      SEARCH [!SEA] %3-%
      VAR9 1
      WRITE The directory %1% has been removed from your searchlist
      WRITE Your current searchlist is [!SEA]
    [!ELSE]
      COMMENT Not the one- add it back to the
      COMMENT searchlist and try next dir on sea
      SEARCH [!SEARCH] %2%
      %0\%/PASS3 %1% %3-%
    [!END]
  [!ELSE]
    [!EQ,[!VAR9],0]
      WRITE The directory %1% is not on your current searchlist
      WRITE Your current searchlist is [!SEA]
    [!END]
  [!END]
[!END]
[!END]
[!END]

```

```

comment +-----+
comment |           CHG_PASSWORD.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This interactive macro may be used to change a user's password
| on one or more systems.
|
+-----[!end]

```

```

PUSH
DIR/I
DEL/1=IGN/2=IGN =?.PSW.[!PID]
YES_OR_NO Do you need help with this macro?
[!EQUAL,[!STRING],YES]
WRITE,,,,,,,,,,,,,
WRITE,,,You are now going to enter some text into a file.
WRITE,,,End each line with a NEWLINE except the line which contains your
WRITE,,,current password. End that line with either ^L or ERASE PAGE.
WRITE,,,When you receive the [!asc 251 251] prompt use the following
WRITE,,,format:
WRITE,,,[!ASCII 251 251] [!ASCII 274]NEWLINE[!ASCII 276]
WRITE,,,[!ASCII 251 251] CURRENT USERNAME [!ASCII 274]NEWLINE[!ASCII 276]
WRITE,,,[!ASCII 251 251] CURRENT PASSWORD [!ASCII 274]ERASE PAGE[!ASCII 276]
WRITE,,,[!ASCII 251 251] NEW PASSWORD [!ASCII 274]NEWLINE[!ASCII 276]
WRITE,,,[!ASCII 251 251 251]
WRITE,,,,,,,,,,,,,
WRITE,,,If done correctly the file should look something like this[!asc 073]
WRITE,,,[!ASC 251 251]
WRITE,,,[!ASCII 251 251] JOAN
WRITE,,,[!ASCII 251 251] BREAD
WRITE,,,[!ASCII 251 251] WATER
WRITE,,,[!ASCII 251 251 251]
WRITE,,,,,,,,,,,,,
[!END]
YES_OR_NO Do you want your changes to be displayed on the screen?
[!EQUAL,[!STRING],NO]
CHAR/OFF/EBO

```

```

[!END]
WRITE [!ASC 214]
CHAR/PM/ON
CRE/I =?.PSW.[!PID]
CHAR/OFF/PM
WRITE,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
WRITE [!ASC 214 212 212 212 212 212 212 212]
CHAR/EBO/ON
WRITE,,,,,Now enter the names of the HOSTS you would like your password
WRITE,,,,,changed on. Enter the following format:
WRITE,,,,,[!ASCII 251 251 250]HOST1,HOST2,HOST3[!ASCII 251] [!ASCII 274]NEWLINE[!ASCII 276]
WRITE,,,,,[!ASCII 251 251 251] [!ASCII 274]NEWLINE[!ASCII 276]
WRITE,,,,,
WRITE,,,,,Your responses will look something like this:
WRITE,,,,,[!ASCII 251 251 250]ARISTO,ZELDA[!ASCII 251]
WRITE,,,,,[!ASCII 251 251 251],,,,,,,,,,
WRITE,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
DEL/2=IGN =?.TMP.SYSTEMS
CRE/I =?.TMP.SYSTEMS
WRITE,,,,,,,,,,,,,[!ASC 214 212 212 212 212 212 212 212]
YES OR NO Do you wish to continue ?
[!NEQUAL,[!STRING],YES]
    POP
    YES OR NO Do you wish to start over?
    [!EQUAL,[!STRING],YES]
        %0%
    [!END]
[!ELSE]
WRITE,,,,,If you wish to view the actual changes respond Yes to the
WRITE,,,,,following question. If you respond No the screen will not
WRITE,,,,,show the changes but will however remain locked while the
WRITE,,,,,changes are in progress. NOTE: It may be wise not to view the
WRITE,,,,,actual process as it will show your password on each change.

```



```

WRITE,,,,,,,,,,,,,,,,,,,,,
YES_OR_NO Do you wish to see the changes on your console?
[!EQUAL,[!STRING],NO]
    WRITE,,,,,,,,,,,,,,,,,[!ASC 214 212 212 212 212 212 212 212]
    WRITE,,,,,The changes are now being made
    WRITE,,,,,,,,,,,,,,,,,,,,,
    WRITE,,,,,Please be patient
    WRITE,,,,,,,,,,,,,,,,,,,,,
    CRE ?.[!PID]
    PROC/DEF/INPUT=? .PSW.[!PID]/OUTPUT=?.[!PID]/CONSOLE=@[!CON]/BLOCK &
    UVTA [=?.TMP.SYSTEMS]
    WRITE,,,,,,,,,,,,,
    WRITE,,,,,If you did not view the changes[!asc 254] it may be wise to
    WRITE,,,,,look at the output file to ensure all went smoothly !!
    YES_OR_NO Do you wish to view the results?
    [!equal,[!string],YES]
        TYPE =?.[!PID]
    [!END]
[!ELSE]
    PROC/DEF/INPUT=? .PSW.[!PID]/OUTPUT=@[!cons]/CONSOLE=@[!CON]/BLOCK &
    UVTA [=?.TMP.SYSTEMS]
[!END]
WRITE [!ASC 214]
WRITE [!ASC 212 212 212]
WRITE ,,,Your password has now been changed on the following system[!ASC 250] s[!ASC 251]
WRITE,,,,,,,,,,,,,[=?.TMP.SYSTEMS]
DEL/2=IGN ?.TMP.SYSTEMS ?.PSW.[!PID] ?.[!PID]
POP
[!END]

```

```
comment +-----+
comment |           CK.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro performs Quality Control checking, telling the user
| what the searchlist is, what CEO and Network processes are running,
| and gives a summary of queue information.
|
+-----[!end]
```

```
WRITE,,,,,,,,,,,,,QUALITY CONTROL
WRITE,,,,,,,,,,,,,FOR
WRITE,,,,,,,,,,,,,[!EXPLODE,[!HOST]]
WRITE,,
WRITE,,,,,,,,,,,,,[!DATE],,,,,[!TIME]
WRITE
SEA
WR
WRITE,,,,,,,,,CEO CHECK.....
WRITE
WRITE
?CEO
WRITE
WRITE
WRITE,,,,,,,,,NETWORK CHECK.....
WRITE
WRITE
?NET
WRITE
WRITE,,,,,,,,,QUEUE CHECK.....
WRITE
WRITE
QD/S/QUE=BATCH_OUTPUT
QD/S/QUE=LPT
QD/S/QUE=LQP
WRITE
WRITE
WRITE,,,,,,,,,TRANSMIT CHECK.....
WRITE
```

WRITE
QD/S/QUE=FTQ
WRITE
WRITE

```
comment +-----+
comment |          COUNT_FILES.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
| This macro will count the number of files in your current working |
| directory.                                                         |
+-----[!end]
```

```
DEL/2=IGNORE =?count?
WRITE/L==?count? [!EDIRECTORY,[!FILENAMES,+]]
WR There are [!UDIV,[!SIZE,?count?],2] files in [!DIRECTORY]
DEL/2=IGNORE =?count?
```

```
comment +-----+
comment |           DC.DOWN.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
|   This non-interactive driver macro brings down the processes
|   brought up by DC.UP.CLI.
|
|   This DOWN macro is system dependent.  Submacros and programs are
|   executed that may not be on every system.  Devices are stopped that
|   may not be on your system.
|
|   Remember, this macro is an example of an DOWN macro and would be
|   modified for your system.
|
+-----[!end]
```

```
[!EQUAL,002,[!PID]]
```

```
WRITE          -- NOW DISABLING ALL CONSOLES --
CONTROL @EXEC DISABLE/ALL
```

```
WRITE          -- NOW STOPPING THE PRINTERS --
:DOWN.PRINTERS
```

```
:UTIL:CEO_DIR:CEO.SYSTEM STOP
CHECKTERMS
```

```
comment   Bring down the network at this point.
```

```
comment   The :DOWN.INFO.S.CLI is normally just a link to
comment   the standard system macro :INFO.S:INFO.S_DOWN.CLI.
```

```
WRITE          -- NOW BRINGING DOWN THE INFO.S PROCESS --
:DOWN.INFO.S.CLI
CHECKTERMS
```

```
WRITE          -- NOW BRINGING DOWN THE EXEC PROCESS --
TERMINATE/1=WARNING/2=WARNING OP:EXEC
PAUSE 2.00
CHECKTERMS
```

```
comment  If the site has a second disk, make the following lines
comment  operational and verify the correct LDU name.
```

```
comm WRITE          -- NOW RELEASING THE SECOND DISK --
comm RELEASE/1=WARNING/2=WARNING LD2
comm PAUSE 2.00
```

```
WRITE
WRITE -- THERE SHOULD NOW BE ONLY TWO PROCESSES LEFT --
WRITE
WHO/1=IGNORE/2=IGNORE <,1,2,3,4,5,6,7,8,9><0,1,2,3,4,5,6,7,8,9>
WHO/1=IGNORE/2=IGNORE 1<,1,2,3,4,5,6,7,8,9><0,1,2,3,4,5,6,7,8,9>
WRITE
WRITE -- TYPING 'UP' WILL BRING THE SYSTEM BACK UP --
WRITE -- TYPING 'BYE' WILL SHUT DOWN THE SYSTEM ENTIRELY --
WRITE
```

```
[!ELSE]
WRITE [!ASCII,207]&
-- ERROR: ONLY PID 2 CAN BRING DOWN THE SYSTEM --
[!END]
```

```

comment +-----+
comment |                DC.UP.CLI                |
comment +-----+

```

```

[!equal,large,comment] -----+
|
|   The list of submacros listed below are called from this UP
|   macro.  They have not been included in the document.
|
|       :UP.DISK.CLI           Initializes other LDUs.
|       :UP.SYSLOG.CLI        Starts SYSLOG.
|       :UP.SET_LINES.CLI*    Initializes line characteristics.
|       :UP.EXEC.CLI          Brings up EXEC.
|       :UP.BATCH.CLI         -- Not implemented --
|       :UP.INFOF.CLI         Invokes :INFOF:INFOF_UP.CLI
|       :UP.NET.CLI           -- Not imlemented --
|       :UP.RECOVER.CLI       Recovers CEO after a crash.
|       :UP.IVERIFY           Optionally verifies the databases.
|       :UP.CEO.SYSTEM.CLI    Invokes :UTIL:CEO_DIR:CEO.SYSTEM.CLI
|       :UP.PRINTERS.CLI*     Starts the printers.
|       :UP.CONSOLES.CLI*     Enables the consoles.
|
|-----+ [!end]

```

```

[!EQUAL,002,[!PID]]
[!EQUAL,,[!PATHNAME,@EXEC]]
  PREFIX PID 2[!ASCII 251]
  PROMPT CHECKTERMS

```

```

comment   Edit the SYSID if desired.  32 characters or less.
comment   +-----+                    maximum length

```

```

      SYSID  CEO Office Automation System

```

```

comment   If slave printer is used, make the following line
comment   operational.

```

```

comm     SLAVE.ON

```

```
comment  Edit the system searchlist if necessary.  If users will
comment  be running PRESENT, include :UTIL:PRESENT.  Likewise for
comment  TRENDVIEW.  :PER is unnecessary.
```

```
SEARCHLIST :UTIL:CEO_DIR,,:UTIL
```

```
PUSH ; PROMPT POP
SUPERUSER ON
DIRECTORY :
ACL @NULL +,WARE
ACL @MTB- +,WARE
CHECKTERMS/L=@NULL
```

```
comment  If the site has a second disk, make the following
comment  lines operational by erasing "comm".  /YES prevents
comment  the question from being asked.
```

```
comm     WRITE          -- NOW INITIALIZING THE SECOND DISK --
comm     :UP.DISK/YES    @DPJ1      +,E
```

```
WRITE          -- NOW STARTING THE SYSLOG --
:UP.SYSLOG
```

```
WRITE          -- NOW INITIALIZING THE IAC LINES --
:UP.SET_LINES
```

```
[!EQUAL, ( ) , (%/NOQCMP%)]
WRITE          -- NOW COMPACTING THE QUEUES --
XEQ QCMP/YES
CHECKTERMS
[!END]
```

```
WRITE          -- NOW CREATING THE EXEC PROCESS --
:UP.EXEC
```

```
comment  If system uses batch streams, make these lines
comment  operational by erasing "comm" and edit UP.BATCH.CLI
comment  as desired.
```



```

comm      [!EQUAL,(),(%/NOBATCH%)]
comm      :UP.BATCH
comm      [!END]

      [!EQUAL,(),(%/NOINFOS%)]
      WRITE          -- NOW CREATING THE INFOS PROCESS --

comment   The :UP.INFO.SCLI is normally just a call to the
comment   standard system macro :INFO.SINFO.SUP.SCLI. If the
comment   operator used /LOGGING, the switch will be passed
comment   through, but the switch isn't necessary for a CEO-only
comment   system because CEO has its own logging facility.

      :UP.INFO.S%0/LOGGING%
      CHECKTERMS
[!END]

comment   If a communications network is installed then
comment   make the following lines operational. Then
comment   edit :UP.NET.SCLI as appropriate. It can simply
comment   call the DGC-supplied network macro(s).

comm      [!EQUAL,(),(%0/NONET%)]
comm      WRITE          -- NOW BRINGING UP THE NETWORK --
comm      :UP.NET
comm      CHECKTERMS
comm      [!END]

      [!NEQUAL,(),(%0/IVERIFY%)]
      WRITE          -- NOW VERIFYING THE INFOS DATABASES --
      :UP.IVERIFY
[!END]

      [!EQUAL,(),(%0/NOCEO/RECOVER%)]
      :UP.CEO.SYSTEM %1%
[!END]

      [!EQUAL,(),(%0/NOPR/NOPRINTERS/RECOVER%)]
      WRITE          -- NOW BRINGING UP THE PRINTERS --
      :UP.PRINTERS
[!END]

```

```

[!EQUAL,(),(%0/NOCON/NOCONS/NOCONSOLES/RECOVER%)]
  WRITE          -- NOW ENABLING THE USERS' CONSOLES --
  :UP.CONSOLES%/LOCAL/NOMODEMS%
[!END]

PRTYPE 2 PREEMPTIBLE
PRIORITY 2 1
SUPERUSER OFF

comment  If system console is TP2, make following lines operational
comment  by erasing "comm".

comm     CHARACTERISTICS/DEF/HARDCOPY
comm     CHARACTERISTICS/RESET

[!NEQUAL,(),(%0/RECOVER%)]
  WRITE          -- NOW STARTING THE RECOVERY PROCEDURE --
  :UP.RECOVER/YES
[!END]

[!EQUAL,(),(%0/NOLOCK/RECOVER%)]
  SUPERUSER ON
  XEQ :LOCK_CLI.PR
  WRITE -----
  WRITE NOTE:,,The process terminations listed above -if any-
  WRITE have occurred since the system was brought up.
  WRITE They may have been deliberately terminated by operator
  WRITE command and do NOT necessarily indicate a problem.
  WRITE -----
[!END]

[!ELSE]
  WRITE [!ASCII,207]&
-- ERROR: EXEC IS RUNNING - BRING THE SYSTEM DOWN FIRST --
[!END]
[!ELSE]
  WRITE [!ASCII,207]&
-- ERROR: ONLY PID 2 CAN BRING THE SYSTEM UP --
[!END]

```

```

comment +-----+
comment |                DF.CLI                |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro allows the user to abbreviate the directory name when
| changing the default directory to a subordinate directory.
|
|     FORMAT:      DF <directory>
|-----[!end]

```

```

[!EQUAL,%0/%,]
  [!EQUAL,([!FILENAMES,%1%+]),()]
    WRITE NO MATCH
  [!ELSE]
    DF/GO,%1%, [!FILENAMES,%1%+]
  [!END]
[!END]

[!EQUAL,%0/GO%,/GO]
  [!EQUAL,%3%,]
    DIR/1=I %2%
    DIR
  [!ELSE]
    WRITE POSSIBILITES ARE:
    WRITE
    F/TY=CPD/TY=DIR/TY=LDU/NHEAD/CPL=15 %1%+
    %0\% %1%[!READ APPEND ENOUGH CHARACTERS TO MAKE YOUR CHOICE UNIQUE&
      [!ASCII 212 207] DF %1%]
  [!END]
[!END]

```

```

comment +-----+
comment |          DIVERT.CLI          |
comment +-----+

```

```

[!equal,large,comment]-----+
|
|   This macro diverts the network messages from the system console
|   to a user console.  The argument ON diverts the message, OFF
|   resets back to system console.  The user must become OP before
|   executing the macro.
|
+-----[!end]

```

```

[!EQ,[!USER],OP]
  [!EQ,%1%,ON]
    CONTROL @(X25,LMGR,FTA,SVTA,RMA) SET/OUT=@[!CON]
  [!ELSE]
    [!EQ,%1%,OFF]
    CONTROL @(X25,LMGR,FTA,SVTA,RMA) SET/OUT
  [!ELSE]
    [!EQ, ,%1%]
    STR [!READ ARGUMENTS ARE "ON" OR "OFF":]
    [!EQ,[!STR],ON]
    CONTROL @(X25,LMGR,FTA,SVTA,RMA) SET/OUT=@[!CON]
    [!ELSE]
    [!EQ,[!STR],OFF]
    CONTROL @(X25,LMGR,FTA,SVTA,RMA) SET/OUT
    [!END]
  [!END]
[!END]
  [!END]
[!END]
[!ELSE]
  WR YOU MUST BE OP TO USE THIS MACRO! [!AS 207]
[!END]

```

```
comment +-----+
comment |          DO.CERTS.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This menu macro is called by CERTS.CLI.  It allows printer forms
| to be changed from 8-1/2" X 11" paper, to Certificate Forms.
|
| Assumptions:  Printer forms are in :UTIL:FORMS; printer is @CON52.
|
+-----[!end]
```

```
write [!asc 214]
wr
wr
wr,,,,,,,,,,,,,,,,,,,,,,,,,[!asc 235]Certificate Printing Utility[!asc 234]
wr
wr,,,,,,,,[!asc 235 250]1[!asc 251 234],,, Change paper in printer to the blankcertificates
and print.
wr
wr,,,,,,,,[!asc 235 250]2[!asc 251 234],,, Change certificates in printer back to blank paper.
wr
wr,,,,,,,,[!asc 235 250]3[!asc 251 234],,, Exit utility and log off terminal.
wr
string [!read Enter option from menu: ]

[!eq [!string],1]
    comment Changing to certificate_forms
    control @exec pause @con52
    pause 1.50
    wr
    wr [!asc 207] When the [!asc 235] Printer [!asc 234]stops printing the current job,
    wr set it up for the [!asc 235]CERTIFICATES[!asc 234] and press the [!asc 235]
    NEW LINE[!asc 234] key. [!read,,]

    control @exec forms @con52 certificate_form
    control @exec cont @con52
    pause 2.00
    do.certs
```

```

[!else]
[!eq [!string],2]
    comment Changing back to plain paper.
    control @exec pause @con52
    pause 1.50
    wr
    wr [!asc 207] When the [!asc 235] Printer [!asc 234]stops printing the current job,
    wr set it up for [!asc 235]PLAIN PAPER [!asc 234] and press the [!asc 235]NEW LINE
    [!asc 234] key. [!read,,]

    control @exec forms @con52
    control @exec cont @con52
    pause 2.00
    do.certs

[!ELSE]
[!EQ [!string],3]
    write [!asc 214]
    write [!asc 235]Good Bye...[!asc 234]
    bye

[!else]
[!neq [!string],1]
    [!neq [!string],2]
        [!neq [!string],3]
            WR
            WR
            write [!asc 207] [!string] ,NOT A VALID OPTION!!!
            PAU 2.00
            do.certs
            [!end]
        [!end]
    [!end]
[!end]
[!end]
[!end]
[!end]

wr [!asc 214]

```

```

comment +-----+
comment |           DTSTAMP.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro takes the current time and date and changes its
| format to one acceptable as a CLI filename (e.g., no dashes).
| It puts the time/date in STRING, which can be later used to
| create a unique filename.
|
+-----[!end]

```

```

[!EQUAL,(),(%2-%)]
%0%,%1%, [!EXPLODE [!DATE][!TIME]]
[!ELSE]
STRING/K
STRING %1![!NEQUAL,,%1%]_[!END]
[!EQUAL,-,%4%]
[!EQUAL,:,%13%]
STRING [!STRING]%2%%3%%5%%6%%7%%9%%10%_%11%%12%%14%%15%%17%%18%
[!ELSE]
STRING [!STRING]%2%%3%%5%%6%%7%%9%%10%_0%11%%13%%14%%16%%17%
[!END]
[!ELSE]
[!EQUAL,:,%12%]
STRING [!STRING]0%2%%4%%5%%6%%8%%9%_%10%%11%%13%%14%%16%%17%
[!ELSE]
STRING [!STRING]0%2%%4%%5%%6%%7%%8%%9%_0%10%%12%%13%%15%%16%
[!END]
[!END]
[!END]

```

```
comment +-----+
comment |           DUMP_UTIL.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro dumps the entire contents of :UTIL onto labeled
| magnetic tape.  You may use READ_UTIL.CLI to verify the tape made,
| and LOAD_UTIL.CLI to delete :UTIL and reload from tape.
| This macro must be run from PID 2.
|
| PRIVILEGES:  PID 2 must have SUPERUSER, BATCH
|
+-----[!end]
```

```
PUSH
PROMPT POP
CHAR/OFF/NRM
CONTROL @EXEC OPERATOR ON
WRITE Prepare to back up the :UTIL directory.  Please mount a
WRITE scratch tape on the tape drive.
WRITE [!READ Hit NEWLINE when ready:,,,)
WRITE
X LABEL @MTB0 UTILA1
WRITE
WRITE Sending dump to batch stream....
WRITE
QBATCH/M
SUPERUSER ON
DIR :UTIL
X DUMP_II/V/BUFF=8192 @LMT:UTILA1:FILES #
SEND 2 Dump complete
)
WRITE
WRITE Dump sent to batch stream.  Please wait for mount and
WRITE dismount requests from EXEC and respond accordingly.
WRITE
WRITE Type READ_UTIL to verify the tape
POP
```



```
comment +-----+
comment |           ENABLE.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This interactive macro will enable/disable consoles at a
| particular time via a batch job.  This macro is executable only
| by username OP.
|
| PRIVILEGES: OP must have BATCH
|
+-----[!end]
```

```
[!EQ,[!USER],OP]
  DEL/2=IGN ??cons??
  WRITE/L=??cons?? @CON[!READ Enter console numbers to be disabled: ][!asc 46]
  WRITE The time is [!TIME]
  QBATCH/QOUT=@NULL/QLIST=@null/M/AFTER=[!READ Enter time to be executed: ]
  CONTROL @EXEC DISABLE [??cons??]
  PAUSE 5.00
  CONTROL @EXEC ENABLE [??cons??]
  comment DEL/2=IGN ??cons??
)
[!ELSE]
  WRITE
  WRITE Only OP can execute this macro.
[!END]
```

```

comment +-----+
comment |           FDEL.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro allows you to delete a file by first checking via
| a Filestatus that this is indeed the file you wish to delete.
|
| When used with a /POFF switch, it will turn permanence off on the
| file to be deleted.
|
+-----+-----[!end]

```

```

[!EQUAL,%0/HELP%,/HELP]
WR,,,,,DESC:,,,THIS MACRO WAS WRITTEN TO HELP PREVENT ANY ACCIDENTAL
WR,,,,,DELETES.,,USING ARGUMENT 1, A FILESTATUS IS PERFORMED
WR,,,,,AND THE QUESTION "IS THIS WHAT YOU WANT TO DELETE?" IS
WR,,,,,ASKED.,,IF THE REPLY IS "Y" THEN ARGUMENT 1 IS DELETED.
WR,,,,,SWITCHES:,,,,,IN ADDITION TO THE NORMAL SWITCHES FOR THE
WR,,,,,FILESTATUS COMMAND, THE
WR,,,,,FOLLOWING SWITCH IS ALSO ACCEPTED.
WR,,,,,/POFF,,,,,THIS WILL TURN PERMANENCE OFF PRIOR TO EXECUTING
WR,,,,,THE DELETE COMMAND.
[!ELSE]
FILESTATUS/ASS/S%0\POFF% %1-%
[!EQUAL,[!READ IS THIS WHAT YOU WANT TO DELETE? ],Y]
    [!EQUAL,%/POFF%,/POFF]
        PERMANENCE (%1-%) OFF
    [!END]
    DELETE/V %1-%
[!ELSE]
WRITE NOTHING WAS DELETED
[!END]
[!END]

```

```
comment +-----+
comment |           FDIR.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
| Displays the names of all the directories in the current directory|
+-----[!end]
```

F/AS/S/TYP=DIR/TYP=CPD

```
comment +-----+
comment |           FIND_PID.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro is called from LOCATE.CLI.  It stores the PID number
| of the Username passed to LOCATE.
|
+-----[!end]
```

```
DEL/2=IGN %3%.PID
STRING %2%
WR/Q/L=%3%.PID [!STRING]
```

```
comment +-----+
comment |           FIND_USER.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro is called from LOCATE.CLI. It passes back the
| results of a WHO command to the driver macro.
|
+-----[!end]
```

```
[!EQ,%1%,OP]
  WHO/Q/L=OP.ID/2=IGN OP:OP
[!ELSE]
  WHO/Q/L=%1%.ID/2=IGN %1%:CON(<,1,2,3,4,5,6,7,8,9>&
<0,1,2,3,4,5,6,7,8,9>)
  WHO/Q/L=%1%.ID/2=IGN %1%:VCON(<,1,2,3,4,5,6,7,8,9>&
<0,1,2,3,4,5,6,7,8,9>)
[!END]
```

```
comment +-----+
comment |           FORMFEED.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
| This submacro is called from other macros to clear the screen. |
|-----[!end]
```

```
WRITE [!ASCII 214]
```

```
comment +-----+
comment |           FS.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
| This is a shortcut filestatus command. |
+-----[!end]
```

```
FILESTATUS/SORT/ASSORTMENT %-%
```

```
comment +-----+
comment |           GET.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
|   This macro moves files from the specified directory into the
|   current directory.  It also sets the ACL to DEFACL.
|
|   FORMAT:   GET  directory  filename...
+-----[!end]
```

```
PUSH
PROMPT POP
DIR %1%
MOVE/V/R/NACL%0\NACL\V% [!DIR/P] %2-%
POP
```



```
comment +-----+
comment |           GET.LISTS.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This interactive macro will print the Host's UP macros, DOWN
| macros, SYSGEN specifications and NETGEN text file.
|
| Assumptions: Macros are in :UTIL:MACROS; UP and DOWN macros start
| with the characters UP and DOWN and are in the root; :SYSGEN
| contains the sysgen information; and :NET:NETGEN contains the
| network information.
|
| Privileges: SUPERUSER
|
+-----[!end]
```

```
PUSH; PROMPT POP
SUPERUSER ON
SEARCHLIST :, :NET:NETGEN, :SYSGEN, :UTIL, :UTIL:MACROS
WRITE
YES_OR_NO Do you want a copy of [!HOST]'s UP macros?
[!EQUAL, ([!STRING]), (YES)]
    QPRINT/V :UP.+CLI
[!END]
WRITE
YES_OR_NO How about the DOWN macros for [!HOST]?
[!EQUAL, ([!STRING]), (YES)]
    QPRINT/V :DOWN.+CLI
[!END]
WRITE
YES_OR_NO Would you like a copy of the SYSGEN listing for [!HOST]?
[!EQUAL, ([!STRING]), (YES)]
    QPRINT/V :SYSGEN:[!HOST].CSF
[!END]
WRITE
YES_OR_NO Would you like a copy of the NETGEN listing for [!HOST]?
[!EQUAL, ([!STRING]), (YES)]
    QPRINT/V :NET:NETGEN:+.TXT
[!END]
POP
```

```

comment +-----+
comment |           GET.TIME.STAMP.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
| This macro is used in CEO Backup (driver macro is BACKUP.CLI). |
+-----[!end]

```

```

[!EQU,%3%,-]
  STR %4%%5%%6%
[!ELSE]
  STR %3%%4%%5%
[!END]
[!EQ,[!STR],JAN]
  STR 01
[!ELSE]
  [!EQ,[!STR],FEB]
    STR 02
  [!ELSE]
    [!EQ,[!STR],MAR]
      STR 03
    [!ELSE]
      [!EQ,[!STR],APR]
        STR 04
      [!ELSE]
        [!EQ,[!STR],MAY]
          STR 05
        [!ELSE]
          [!EQ,[!STR],JUN]
            STR 06
          [!ELSE]
            [!EQ,[!STR],JUL]
              STR 07
            [!ELSE]
              [!EQ,[!STR],AUG]
                STR 08
              [!ELSE]
                [!EQ,[!STR],SEP]
                  STR 09

```



```
comment +-----+
comment |           GET_STRING.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro is called from a driver macro.  It will repeat until
| the user enters a text string.  The text is placed in STRING.
|
+-----[!end]
```

```
STRING [!READ %-%,]
[!EQUAL,(),([!STRING])]
    %0% %-%
[!END]
```

```

comment +-----+
comment |           HASH.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This interactive macro will determine the optimal hashframe
| size for a directory.
|
+-----[!end]

```

```

[!neq,%0/p%,/p]
  push;prompt pop
  [!eq,%1%,]
    string [!read Enter directory pathname to be counted : ]
  [!else]
    string %1%
  [!end]
  del/2=ign :udd:[!user]:??<count hash name>??
  push;dir/2=error [!string]
  wr/l=:udd:[!user]:??name?? [!ef [!f +]]
  wr/l=:udd:[!user]:??count?? [!ed [!f +]]
  f/hash/l=:udd:[!user]:??hash?? =
  pop
  comment ..... get the file count from the ??count?? file size
  var0 [!udiv [!size :udd:[!user]:??count??],2]
  comment ..... get the average byte length of filenames
  var1 [!usub [!udiv [!size :udd:[!user]:??name??],[!var0]],1]
  comment ..... get the optimum hashframe size
  var2 [!udiv [!var0] [!udiv 310 [!var1]]]
  [!eq,[!var2],0]var2 2[!end]
  wr Directory = [!path [!string]] has a filecount = [!var0] &
  and a hashframe size [:udd:[!user]:??hash??]
  wr Average filename size = [!var1] bytes .. Computed hashframe size is [!var2]
  wr Now computing next higher prime number for hashframe optimization .....
  del :udd:[!user]:??<count hash name>??
  %0%/p
  pop

```

```

[!else]
comment ..... var2 contains the exact hash size, get the next higher prime
[!neq,%0/r%,/r]
    var0 1;var1 2;%0%/r
[!else]
    [!eq,[!umod [!var2] [!var1]],0]var0 0;var1 [!usub [!var2],1][!end]
    var1 [!uadd [!var1],1]
    [!neq,[!var2],[!var1]]
        %0%
    [!else]
        [!neq,[!var2],251]
            [!eq,[!var0],1]
                wr Directory hashframesize should be [!var2] !
            [!else]
                var2 [!uadd [!var2],1]
                %0\%/p
            [!end]
        [!else]
            wr Maximum hashframe size is 251!
        [!end]
    [!end]
[!end]
[!end]

```

```
comment +-----+
comment |           HOW.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro will do a WHO and RUNTIME command on whatever user
| PID is passed in as the first argument.  You may pass in as a
| second argument how many seconds you want the WHO and RUNTIME
| repeated.  The default is 5 seconds.  This macro may be used
| interactively if no arguments are passed in.  In order to stop
| the display, ^C^A.
|
+-----[!end]
```

```
[!equal,%1%,]
  HOW [!READ PID number? ] 5
[!else]
  [!equal,%2%,]
    HOW %1% 5
  [!else]
    WRITE [!ASCII 213 227]
    WHO %1%
    RU %1%
    PAU %2%
    WHO/2=ERROR/L=@NULL %1%
    WRITE [!ASCII 227 227 227]
    HOW %1% %2%
  [!end]
[!end]
```

```
comment +-----+
comment |           KILL.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This interactive macro assumes that you know the PID number of
| a process that you wish to terminate. You must have Super-
| process privilege to be able to use it.
|
+-----[!end]
```

```
PUSH
PROMPT POP
SUPERPROCESS ON
WRI
WRI In order to "kill" a process, you must know its PID number.
WRI If you don't know the PID number, use the ? macro to find it.
WRI
WRI Enter the PID number of the process you wish to terminate or
STRING [!READ press NEWLINE to exit this macro:,, ,]
[!NEQ,[!STRING],]
    TERM [!STRING]
    WRI
    WRI PID [!STRING] has been terminated.
[!END]
POP
    TERM OP:SQL1
    TERM OP:SQL2
    PAUSE 4.000
    CHECK
[!END]
```



```
DEL/1=I/2=I ?WHO.TMP
WHO/1=I/2=I/L=?WHO.TMP CEO_MGR:CEO_FSA
WHO/1=I/2=I/L=?WHO.TMP CEO_MGR:CEO_POA
WHO/1=I/2=I/L=?WHO.TMP CEO_MGR:CEO_CSA
WHO/1=I/2=I/L=?WHO.TMP CEO_MGR:CEO_QMA
[!NEQ,0,[!SIZE ?WHO.TMP]]
    WRITE
    WRITE
    WRITE Bringing down CEO ...
    :UTIL:CEO_DIR:CEO_DOWN
    PAUSE 4.000
    CHECK
```

[!END]

```
DEL/1=I/2=I ?WHO.TMP
WHO/1=I/2=I/L=?WHO.TMP CEO_MGR:CEO_SECRETARY
[!NEQ,0,[!SIZE ?WHO.TMP]]
    WRITE
    WRITE
    WRITE Bringing down CEO SECRETARY ....
    TERM CEO_MGR:CEO_SECRETARY
    PAUSE 4.000
    CHECK
```

[!END]

```
DEL/1=I/2=I ?WHO.TMP
WHO/1=I/2=I/L=?WHO.TMP OP:INFOS_II
[!NEQ,0,[!SIZE ?WHO.TMP]]
    WRITE
    WRITE
    WRITE Bringing down INFOS_II
    :INFOS:INFOS_DOWN
    PAUSE 4.000
    CHECK
```

[!END]

```
DEL/1=I/2=I ?WHO.TMP
WHO/1=I/2=I/L=?WHO.TMP OP:NETOP
[!NEQ,0,[!SIZE ?WHO.TMP]]
    WRITE
    WRITE
    WRITE Bringing down the NETWORK
    :NET:DOWN.NETWORK
    PAUSE 4.000
    CHECK
[!END]

[!EQ,Y,[!READ,DO YOU WANT TO STOP SYSLOG (Y OR N) ? ]]
    SYSLOG/STOP
[!END]

    SUPERUSER OFF
    BYE
[!ELSE]
    WRITE
    WRITE Please terminate users gracefully and re-enter the "DOWN" macro.
[!END]
```

```
comment +-----+
comment |           LA.DOWN.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This driver macro brings down those processes brought up in
| LA.UP.CLI.
|
+-----[!end]
```

```
[!NEQUAL,002,[!PID]]
    WRITE *ERROR*
    WRITE %0\% IS VALID ONLY WHEN INVOKED BY THE ROOT CLI
    WRITE REQUIRED ARGUMENTS: NONE
    WRITE OPTIONAL SWITCHES:,,NONE
    WHO/1=E/2=E/L=@NULL END_OF_MACRO
```

```
[!END]
:MACROS:?
```

```
WRITE
[!EQ,Y,[!READ,Are all the USERS logged off? (Y,<N>): ]]
    SUPERUSER ON
    WRITE
    WRITE Bringing down Exec...
    TERM/2=I OP:EXEC
    PAUSE 4
    CHECK
```

```
    DEL/1=I/2=I ?WHO.TMP
    WHO/1=I/2=I/L=?WHO.TMP OP:UNIX
    [!NEQ,0,[!SIZE ?WHO.TMP]]
        WRITE
        WRITE
        WRITE Bringing down UNIX...
        TERM OP:UNIX
        PAUSE 3.000
        CHECK
```

```
[!END]
```

```
DEL/1=I/2=I ?WHO.TMP
WHO/1=I/2=I/L=?WHO.TMP OP:RES_ADA
[!NEQ,0,[!SIZE ?WHO.TMP]]
    WRITE
    WRITE
    WRITE Bringing down ADA ...
    TERM OP:RES_ADA
    PAUSE 3.000
    CHECK
[!END]
```

```
DEL/1=I/2=I ?WHO.TMP
WHO/1=I/2=I/L=?WHO.TMP OP:SQL1
WHO/1=I/2=I/L=?WHO.TMP OP:SQL2
[!NEQ,0,[!SIZE ?WHO.TMP]]
    WRITE
    WRITE
    WRITE Bringing down SQL ...
```

```

comment +-----+
comment |           LA.UP.CLI           |
comment +-----+

```

```
[!equal,large,comment]-----+
```

```

This driver UP macro brings up the LA system.

There are two ways to bring up the system, the default method
which brings up everything available on the system, or
a selective method wherein the macro prompts you for the software
and devices you want to bring up.

Implement these methods by if asking the user wants to
bring up the default system (everything).  If so, warn them
to be sure that all printers are on-line and ready, also
warn them that the DPJ10 disk should be on-line and ready.

If the user decides to selectively bring up the system, they
will be prompted UP-FRONT for their desired options, which are
then tabled.  The table and its allowable values are:

```

VARIABLE	MEANING	'YES' VALUE	'NO' VALUE
VAR0	START INFOS_II PROCESS?	1	0
VAR1	START THE NETWORK?	1	0
VAR2	START CEO PROCESSES?	1	0
VAR3	START SQL PROCESSES?	1	0
VAR4	START THE FORMA-TEXT PROCESS?	1	0
VAR5	START THE LINE PRINTER LPB?	1	0
VAR6	START THE NEC LETTER QUALITY PRINTER?	1	0
VAR7	START THE DRIED APRICOT PRINTER?	1	0
VAR8	ENABLE ALL CONSOLES?	1	0
VAR9	---- RESERVED ----	XXX	XXX

```
-----+-----[!end]
```

```

PUSH;PROMPT POP
[!NEQ,002,[!PID]]
    WR ,,,,,You must be OP,-,,PID 002 to run this macro.
    WHO/1=E/2=E/L=@NULL END_OF_JOB
[!END]

```

```

WRITE [!ASC 207 214];WR;WR;WR
WRITE ,,,,,,,You may bring up the DEFAULT system which includes all
WRITE ,,,,,,,software and peripherals known to the system, or you
WRITE ,,,,,,,may selectively bring up a CUSTOM combination of software
WRITE ,,,,,,,and peripherals.
WRITE
WRITE ,,,,,,,To bring up the DEFAULT system .... press NEW-LINE.
WRITE ,,,,,,,To bring up a CUSTOM combination, enter 'C' and NEW-LINE.
WRITE
WRITE
STRING [!READ,,,,,,,,,ENTER YOUR CHOICE .....:,,]

COM      Here you initialize the selector variables to 0 and depending
COM      on the user's reply to the DEFAULT or CUSTOM question, you
COM      set all the selector variables on for DEFAULT bring-up, or you
COM      go through a selector macro which gets the combinations desired
COM      by the user.

VAR(0 1 2 3 4 5 6 7 8 9) 0
[!EQ,C,[!STR]]
      :UP_SELECTIVE
[!ELSE]
      VAR(0 1 2 3 4 5 6 7 8 9) 1
[!END]

WRITE [!ASC 207 214];WR;WR;WR

[!UEQ,1,[!VAR5]]
      WR ,,,,,,,Be sure the LINE-PRINTER is ON-LINE and READY .....
[!END]

[!UEQ,1,[!VAR6]]
      WR ,,,,,,,Be sure the NEC SPINWRITER is ON-LINE and READY .....
[!END]

[!UEQ,1,[!VAR7]]
      WR ,,,,,,,Be sure the DRIED APRICOT PRINTER is ON-LINE and READY .....
[!END]

```

```

WR ,,,,,,,Be sure the DPJ10 ARGUS DRIVE is ON-LINE and READY .....
WR
WR
WR
[!READ ,,,,,,,Press NEW-LINE when ready to proceed ....]

WRITE [!ASC 207 214];WR;WR;WR
WRITE
WRITE,,,,,,Starting SYSLOG ....
SUPERUSER ON
DEL/2=I  OLD_LOG
CRE/2=I  SYSLOG
SYSLOG/START OLD_LOG
SUPERUSER OFF

COM      SET THE SYSID ... SET THE PROMPT ... SET THE ACL ON THE TAPE
COM      INIT THE SECOND ARGUS DRIVE ... RECREATE PAGE AND SWAP DIRECTORY
COM
SYSID [!ASC,235]LA Education Center [!ASC,224]MERLIN[!ASC,225,234,207]
PROMPT Dir Check
SEA :Util :Macros :
SUPERUSER on
ACL @Mtb- +.+, , +,WARE
CRE/link @Mta0 @Mtb0

[!Equal, , [!path, :Udd]]
    Write,Initializing...
    Init @DPJ10
    Pause 2
    Acl :Udd + E
[!End]

[!Ne, :Udd:Page, [!path, :Page]]
    Del/2=I :Old_Page, :Old_Swap
    Ren (Page, Swap) (Old_<Page, Swap>)
    Cre/Link/2=I (Page, Swap) :Udd:(Page, Swap)
[!End]

WRITE
WRITE,,,,,,Now executing QCMP .... compressing Queues.
X/1=W Qcmp/Yes

```

```

WRITE
WRITE,,,,,,,,,Now bringing up Exec...
SEA :Macros :Util :
Proc/Def/Dir=@/Name=Exec :Util:Exec
Pause 2

WRITE
WRITE,,,,,,,,,Now bringing up BATCH STREAM ONE ....
Cont @Exec Prompts off
Cont @Exec Continue 1
Cont @Exec Silence 1

Sea :Util,:Macros,:Per,:Infos,:
[!UEQ,1,[!VAR0]]
    WRITE
    WRITE,,,,,,,,,Now bringing up 32-bit INFOS...
    :INFOS:INFOS_UP
    PAUSE 5
[!END]

[!UEQ,1,[!VAR1]]
    WRITE
    WRITE,,,,,,,,,Now bringing up the NETWORK ....
    :Net:Up.network
    PAUSE 5
[!END]

[!UEQ,1,[!VAR2]]
    WRITE
    WRITE,,,,,,,,,Now bringing up CEO ....
    :UTIL:CEO_DIR:CEO.SYSTEM START CHECKPOINT 10
    PAUSE 2
    WRITE,,,,,,,,,Initiating the CEO_SECRETARY process.
    Proc/def/dir=@/name=ceo_secretary/pri=2, secretary
    PAUSE 2
[!END]

[!UEQ,1,[!VAR3]]
    WRITE
    WRITE,,,,,,,,,Now bringing up the SQL processes .....
    :UTIL:SQL_UP
    PAUSE 18
[!END]

```



```

[!UEQ,1,[!VAR4]]
    WRITE,,,,,,,,,Initiating FORMA_TEXT   WPEXC   Print Utility
    PUSH
    DIR :WRDP
    UPWPEXC
    POP
[!END]

[!UEQ,1,[!VAR5]]
    WRITE
    WRITE,,,,,,,,,Now bringing up the LINE-PRINTER ....
    Cont @Exec Start (Batch_<output,list> Lpt) @Lpb
    Pause 3
    Cont @Exec Continue @Lpb
    Cont @Exec Silence @Lpb
    Cont @Exec Operator on
    Pause 3
[!END]

[!UEQ,1,[!VAR6]]
    WRITE,,,,,,,,,Now bringing up the NEC letter-quality printer ....
    WRITE,,,,,,,,,Ignore the little error messages from EXEC.
    CE DISABLE @CON56
    PAUSE 2
    :UTIL:CEO_DIR:CEO.PRINTER START LQP
    PAUSE 5
[!END]

[!UEQ,1,[!VAR7]]
    WRITE,,,,,,,,,Now bringing up the DRIED APRICOT printer ....
    WRITE,,,,,,,,,Ignore the little error messages from EXEC.
COM    :UTIL:CEO_DIR:CEO.PRINTER START APRICOT
COM    CONTROL @EXEC Start LQP1 @Con8
    PAUSE 5
[!END]

```

```
[!UEQ,1,[!VAR8]]
COM      ENABLE ALL CONSOLES
WRITE
WRITE,,,,,,Now enabling all CONSOLES ....
Control @Exec Enable/All
PAUSE 10
WRITE
WRITE,,,,,,Now disabling CONSOLE 50 for use by the SCANTRON ...
CE DISABLE @CON50
PAUSE 5
```

```
[!END]
```

```
WRITE Now bringing up the UNIX DEMON .....
:UNIX:BIN:START_UNIX_DEMON
PAUSE 3
```

```
WRITE
WRITE
WRITE [!Asc 224]Console locking[!Asc 225]
Sysid LA Education Center MERLIN
Pri OP:LPB 1
LOCK
```

```
comment +-----+
comment |                LDUS.CLI                |
comment +-----+
```

```
[!equal,large,comment]-----+
| This macro displays the names of all LDU's on the current host. |
+-----[!end]
```

```
PUSH; PROMPT POP
SUPERUSER ON
DIR :
WRITE The following are LDU's on System [!HOST]
WRITE
F/AS/S/TYP=LDU +
POP
```

```

comment +-----+
comment |           LINE.PRINTER.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro keeps telling you how many entries are in LPT until
| there are 0 entries.
|
+-----[!end]

```

```

[!EQUAL,%0/LOOP2%,]
  [!EQUAL,%0/LOOP1%,]
    PUSH;PROMPT POP LEVEL
    CLASS(1,2) WARNING
    STRING :UDD:[!USER]:?[!PID].%0/%.TMP
    PERM/L=@NULL [!STRING]
    DELETE/L=@NULL [!STRING]
    %0%/LOOP1
    WRITE [!ASCII 207]
    POP
  [!ELSE]
    QDISPLAY/SUMMARY/QUE=LPT/L=[!STRING]
    %0%/LOOP2 [[!STRING]]
  [!END]
[!ELSE]
DELETE [!STRING]
WRITE There are %4% entries in the LPT queue.
[!NEQUAL,%4%,0]
  PAUSE 5
  %0\%%0\LOOP2%
[!END]
[!END]

```

```
comment +-----+
comment |                LIST.PROFILES.CLI                |
comment +-----+
```

```
[!equal,large,comment]-----+
|
|   This macro displays the names of all the user directories and
|   user profiles on the host system.
|
+-----[!end]
```

```
PUSH; PROMPT POP
SUPERUSER ON
LISTFILE [!HOST].PROFILES
DIRECTORY :UDD
DELETE/2=IGNORE ?+BRK
WRITE/L User Directories for System [!HOST]
FILESTATUS/SORT/ASSORTMENT/L
WRITE/L
WRITE/L
WRITE/L ,,,,,,~~ End of User Directories for System [!host] ~~
DIR :UPD
DELETE/2=IGNORE ?+BRK
WRITE/L
WRITE/L User Profiles for system [!HOST]
FILESTATUS/SORT/ASSORTMENT/L
WRITE/L
WRITE/L
WRITE/L ,,,,,,~~ End of User Profiles for System [!HOST] ~~
LISTFILE/K
POP
QPRINT/DELETE/TITLE/DEST=AOS_COORDINATOR/COPIES=4 [!HOST].PROFILES
```

```
comment +-----+
comment |          LOAD_UTIL.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro deletes the entire contents of :UTIL and reloads it
| from the labeled magnetic tape created by DUMP_UTIL.CLI.
| This macro must be run from PID 2.
|
+-----[!end]
```

```
PUSH
PROMPT POP
CHAR/OFF/NRM
CONTROL @EXEC OPERATOR ON
WRITE
WRITE Prepare to reload the :UTIL directory
WRITE
WRITE Sending load to batch stream
WRITE
QBATCH/M
SUPERUSER ON
DIR :UTIL
DELETE +:#
X LOAD_II/V/BUFF=8192 @LMT:UTILA1:FILES #
SEND 2 Load Complete.
)
WRITE
WRITE Load sent to batch stream. Please wait for mount and
WRITE dismount requests from EXEC and respond accordingly.
WRITE
POP
```

```
comment +-----+
comment |          LOCATE.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
| This interactive macro finds the PID number for whatever Username |
| you give it.  It calls macros FIND_USER.CLI and FIND_PID.CLI     |
|-----[!end]
```

```
PUSH
PROMPT POP
WR
WR Hold on please!
WR
WR Paging %1% .....
FIND_USER %1%
FIND_PID [%1%.ID]
STRING [%1%.PID]
WR
WR %1% is hiding out as PID: [!STRING]
WR
STRING/K
DEL %1%.<P,>ID
POP
```

```

comment +-----+
comment |                MAKER1.CLI                |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro is called by P_MAKER.CLI. It is a recursive macro to
| create all the users specified as arguments to P_MAKER.CLI.
|
+-----+-----[!END]

```

```

WRITE/L,USE
[!EQUAL,,%0/def=%]wr/l,usr[!ELSE]wr/l,%0/def=%[!END]
WRITE/L,CREATE
[!EQUAL,%1%,]
    STRING [!READ Input Username:,,]
    WRITE/L [!STRING]
[!ELSE]
    WRITE/L %1%
[!END]
WRITE/L,Y
[!EQUAL,[!VAR0],1]
    WRITE/L,PASSWORD
[!ELSE]
    WRITE/L,[!READ,Input password for [!EQUAL,,%1%][!STRING][!ELSE]%1%[!END]:,,]
[!END]
WRITE/L(,,,,,,,,,,,,,,,,,,,,,,,,,,,,)
WRITE/L,Y
[!EQUAL,,%1%]
    [!EQUAL,,%0/def=%]
        WRITE/L,[!STRING],,[!DATE],,DEF=USR
    [!ELSE]
        WRITE/L,[!STRING],,[!DATE],,DEF=%0/def=%
    [!END]
[!ELSE]
    [!EQUAL,,%0/def=%]
        WRITE/l,%1%,,[!DATE],,DEF=USR
    [!ELSE]
        WRITE/L,%1%,,[!DATE],,def=%0/def=%
    [!END]
[!END]

```



```
WRITE/L, BYE
PROC/DEF/BLOCK/IOC/IN=?[!pid].profile.in/OUT=?[!pid].profile.out, &
  PREDITOR
[!NEQUAL, , %2%]
  %0%/CONT, %2-%
[!END]
```

```
comment +-----+
comment |           MASS.PROFILE.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This interactive macro sets up a default Creditor profile and
| then allows you to create as many profiles as necessary using
| that default.  It calls MASS.PROFILE1.CLI.
|
| PRIVILEGES: SUPERUSER
|
+-----+ [!end]
```

```
WRITE [!ASCII 214]
DEL/2=IGNORE PRED.IN MP.LOG
WR/L=PRED.IN USE
WR
WR ,, , , , , , , Welcome to our Create Profile program
WRI
WR/L=PRED.IN [!READ Please enter the Default Username .....:, ,]
WR
WR
WR .... One moment please while I setup the default profile.
WR
WR
WR/L=PRED.IN QUESTION
COM      1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
WR/L=PRED.IN (Y,N,N,N,N,N,N,N,N,N,N,Y,Y,N,N,N,N,&
N,N,N,N,N,N,N,N,Y,N,N,N,N,N,N Y)
```

```
COM      7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
MASS.PROFILE1
WR/L=PRED.IN BYE
WR
WR
DEL/2=IGN MP.LOG
CRE/2=IGNORE MP.LOG
WR/L=MP.LOG [!ASC 214]
WR/L=MP.LOG THESE PROFILES CREATED BY [!USER]
WR/L=MP.LOG ON [!DATE] AT [!TIME]
WR/L=MP.LOG
WR/L=MP.LOG
PROC/DEF/BL/IN=PRED.IN/OUT=MP.LOG/CON/DATA/LIST PREDITOR
WRITE
WRITE Profiles have been created!
WRITE
DEL/2=IGN MP.LOG PRED.IN
```

```

comment +-----+
comment |           MASS.PROFILE1.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
|
|   This macro is called from MASS.PROFILE1.CLI.  It asks for infor-
|   mation on each user you want to create, and then passes that
|   information back to the driver macro.
|
+-----[!end]

```

```

WRITE
STRING [!READ Enter Username or * to end.....:, ,]
[!NEQ, (*), ([!STRING])]
    WR/L=PRED.IN CREATE
    WR/L=PRED.IN [!STRING]
    WR/L=PRED.IN Y
    WR/L=PRED.IN [!READ Enter Password (3-15 characters).....:, ,]
    WR/L=PRED.IN N
    WR/L=PRED.IN Y
    WR/L=PRED.IN [!READ Enter Number of Sons (0-255).....:, ,]
    WR/L=PRED.IN Y
    WR/L=PRED.IN [!READ Enter Disk Quota (0-2147483647).....:, ,]
    WR/L=PRED.IN Y
    WR/L=PRED.IN [!READ Enter User Comment.....:, ,]
MASS.PROFILE1
[!END]

```

```
comment +-----+
comment |           MEGABYTES.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
| This interactive macro tells the user how many megabytes of memory |
| the system has.  It calls MEGS.CLI.                               |
+-----[!end]
```

```
PUSH; PROMPT POP
STRING [!READ Do you need help on this macro? (YES/NO),,,]
[!NEQ,([!STRING]),(NO)]
    WRI
    WRI This macro will tell you how much memory your
    WRI system currently has.
    WRI
    WRI In order to use this macro successfully, you must
    WRI find out the number of "Pages" on the system.
    WRI
    WRI You can find the number of pages by the following
    WRI results from SYSINFO.  You are looking for
    WRI SYSTEM MEMORY:  nnnn PAGES
    WRI
    SYSINFO
    PAUSE 2
    WRI
    MEGS
[!ELSE]
    MEGS
[!END]
POP
```

```
comment +-----+
comment |           MEGS.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
|   This macro is called from MEGABYTES.  It tells the user how
|   many megabytes of memory the system has.
|
+-----[!end]
```

```
PUSH; PROMPT POP
VAR1 [!Read How many pages? (Don't put in commas):,,,,]
WRI
WRI Memory Information:
WRI for [!HOST]
WRI [!DATE]
WRI
WRI Pages: [!VAR1]
VAR2 [!UDIV,[!VAR1],512]
WRI ,,,,,,,,,,,,,,MEGABYTES: [!VAR2]
WRI
POP
```

```

comment +-----+
comment |                MM.CLI                |
comment +-----+

[!equal,large,comment]-----+
|
|  This macro gives process information on the user.
|
+-----+-----[!end]
[!EQUAL,%0/LOOP6%,]
  [!EQUAL,%0/LOOP5%,]
    [!EQUAL,%0/LOOP4%,]
      [!EQUAL,%0/LOOP3%,]
        [!EQUAL,%0/LOOP2%,]
          [!EQUAL,%0/LOOP1%,]
            PUSH
            CLASS(1,2) WARNING
            STRING :UDD:[!USER]:?[!PID].%0\%.TMP
            PR(T,I)/L=@NULL [!PID] %1\%
            ACL/L=@NULL [!STRING] [!USER],OWARE
            PERM/L=@NULL [!STRING] OFF
            DELETE/L=@NULL [!STRING]
            WHO/L=[!STRING] [!PID]
            %0%/LOOP1 [[!STRING]]
            POP
          [!ELSE]
            DELETE/L=@NULL [!STRING]
            %0%/LOOP2/PID=%2\%/USER=%3\%/CON=%4\%/PROG=%5\%
          [!END]
        [!ELSE]
          PRT/L=[!STRING]
          %0%/LOOP3 [[!STRING]]
        [!END]
      [!ELSE]
        DELETE/L=@NULL [!STRING]
        %0%/LOOP4/PRT=%1\%
      [!END]
    [!ELSE]
      PRI/L=[!STRING]
      %0%/LOOP5 [[!STRING]]
    [!END]

```

```
[!ELSE]
  DELETE/L=@NULL [!STRING]
  %0%/LOOP6/PRI=%1\%
[!END]
[!ELSE]
WRITE
WRITE Your process information follows :
WRITE
WRITE,,,,,Username : %0/USER=%
WRITE,,,,,Process ID : %0/PID=%
WRITE,,,,,Program : %0/PROG=%
WRITE,,,,,Process Name : %0/CON=%
WRITE,,,,,Process Type : %0/PRT=%
WRITE,,,,,Process Priority : %0/PRI=%
WRITE
[!END]
```



```
comment +-----+
comment |           MON.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro allows users to monitor a specific process ID in order
| to verify utilization of CPU time.  It expects a PID number as
| the first argument.
|
| It will display help if /HELP is used on macro invocation.
|
| It must be terminated by pressing CTRL-C CTRL-A.
|-----[!end]
```

```
[!EQUAL,%0/LOOP%,]
PUSH
CLASS(1,2) IGNORE
LOGIT %0-%
POP
[!NEQUAL,%0/H%%0/HE%%0/HEL%%0/HELP%,]
WRITE
WRITE PURPOSE : ALLOW ALL USERS TO MONITOR A SPECIFIC PROCESS ID
WRITE,,,,,,,,,IN ORDER TO VERIFY UTILIZATION OF CPU TIME
WRITE
WRITE WARNING : 1. SINCE THIS MACRO PLACES ITSELF IN AN INFINITE LOOP
WRITE,,,,,,,,,BY CALLING ITSELF RECURSIVELY - IT MUST BE TERMINATED
WRITE,,,,,,,,,BY THE CONTROL SEQUENCE 'CTRL-C CTRL-A'
WRITE
WRITE,,,,,,,,,2. DUE TO ITEM 1 ABOVE - AND THE DESIRE TO LEAVE THE USER
WRITE,,,,,,,,,AT THE SAVE CLI PUSH LEVEL - THE CONTENTS OF THE
WRITE,,,,,,,,,CLI 'STRING' VARIABLE WILL BE UNDEFINED
WRITE
WRITE GLOBAL SWITCHES : /HELP - DISPLAY HELP INFORMATION
WRITE
WRITE NOTE : GLOBAL SWITCHES MAY BE ABBREVIATED
WRITE
WRITE ARGUMENT 1 REQUIRED : PROCESS ID REQUIRED
WRITE
WRITE LOCAL SWITCHES : NONE
```

```

WRITE
[!ELSE]
  [!EQUAL,%1\%,]
  WRITE
  WRITE ARGUMENT 1 REQUIRED : PROCESS ID NUMBER
  WRITE
  [!ELSE]
  PAUSE 0.2
  WRITE [!ASCII 214 212 212 212 212 212 212]
  WRITE [!ASCII 211 211 211]Automatic Process Monitor
  WRITE
  WRITE [!ASCII 211 211 211 211 203 216 224]Working![!ASCII 217 225]
  WRITE
  WRITE [!ASCII 211 211],,,Enter 'CTRL-C CTRL-A' to terminate
  WRITE [!ASCII 212 212 212 212 212 212 212 212]
  STRING :UDD:[!USER]:?![!PID].%0\%.TMP
  PERM/1=I/2=I [!STRING] OFF
  DELETE/1=I/2=I [!STRING]
  %0\%/LOOP/REQID=%1\%
  [!END]
  [!END]
[!ELSE]
  [!EQUAL,%0/RUN_SW%,]
  [!EQUAL,%0/WHO_SW%,]
  WHO/L=[!STRING] %0/REQID=%
  %0\%/WHO_SW [!STRING]
  [!ELSE]
  DELETE/1=I/2=I [!STRING]
  [!NEQUAL,(%1\%),(PID:)]
  WRITE [!ASCII 210 212 212 212 212 212 212 212 212]
  WRITE [!ASCII 211 211 211],,,Process %0/REQID=% is inactive,,,,,,,,,&
  //////////////////////////////////////////////////
  WRITE [!ASCII 212 212 212 212 212 212 212 212 212 212]
  %0\%%0\WHO_SW
  [!ELSE]
  RUN/L=[!STRING] %0/REQID=%
  %0\%/RUN_SW/USER=%3\%/PROG=%5\% [!STRING]
  [!END]
[!END]

```

```

[!ELSE]
DELETE/1=I/2=I [!STRING]
[!NEQUAL, (%1\%), (ELAPSED)]
WRITE [!ASCII 210 212 212 212 212 212 212 212 212]
WRITE [!ASCII 211 211 211],,Process %0/REQID=% is inactive,,,,,,,,,&
////////////////////////////////////
WRITE [!ASCII 212 212 212 212 212 212 212 212 212]
%0\%%0\RUN_SW\USER\PROG%
[!ELSE]
WRITE [!ASCII 210 212 212 212 212 212 212 212 212]
WRITE User = %0/USER=%,,PID = %0/REQID=%,,CPU Time = %4%&
,,Program = [!EFILE %0/PROG=%]
WRITE [!ASCII 212 212 212 212 212 212 212 212 212]
%0\%/LOOP/REQID=%0/REQID=%
[!END]
[!END]
[!END]

```

```
comment +-----+
comment |           MOVE.ZELDA.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This interactive macro will transfer files from host ARISTO to
| host ZELDA.  It needs to know the name of the files to be moved
| and the directory on ARISTO to which they are to be moved.
|
| Assumptions: Source HOST is ARISTO, destination HOST is ZELDA;
| same username and password on both systems; user profile allows
| use of FTA.
|
| PRIVILEGES:  SUPERUSER
|
+-----[!end]
```

```
PUSH
PROMPT POP
SUPERUSER ON
STRING [!READ What files do you want moved to ZELDA?,,,,]
WRI
comment Push now so you won't lose the previous
comment value of STRING
PUSH
PROMPT POP
WRI What directory should they be moved to?
STRING [!READ e.g., :UTIL:MACROS,,,,]
MOVE/V/FTA%0/% :NET:ZELDA[!STRING] [!STRING/P]
POP
pop
wri
wri Done!
wri
wri Don't forget to change the files ACLs if necessary
wri
```

```
comment +-----+
comment |           NOW.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|           This macro displays the current date and time.           |
|-----[!end]
```

```
DATE
TIME
```

```
comment +-----+
comment |           ON.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro turns Superuser on, Superprocess on, or both on
| depending on what arguments are passed.  If no arguments are
| passed in, it will turn Superuser on.
|
| PRIVILEGES:  SUPERUSER, SUPERPROCESS
|
+-----+ [!end]
```

```
[!EQ,,%1%]
  SUPERUSER ON
[!ELSE]
  [!EQ,*,%1%]
    SUPERUSER ON
  [!ELSE]
    [!EQ,+,%1%]
      SUPERPROCESS ON
    [!ELSE]
      [!EQ,#,%1%]
        SUPERUSER ON
        SUPERPROCESS ON
    [!END]
  [!END]
[!END]
[!END]
```

```

comment +-----+
comment |                PASCAL.MENU.CLI                |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This driver menu macro allows the user to select the following
| choices for PASCAL program development:
|
| {0} Bye
| {1} Create/Edit Source File
| {2} Compile
| {3} Link
| {4} Print Listing
| {5} Execute Program
|
| It calls the following submacros:
|     PASCAL.MENU_1          PASCAL.MENU_2
|     PASCAL.MENU_3          PASCAL.MENU_4
|     PASCAL.MENU_5
|
+-----[!end]

```

```

[!NEQUAL,%0/cont%,/cont]
PUSH
PROMPT POP
[!END]
CLASS1 WARNING
WRITE [!ASCII 214 210,234,203][!DATE],,,,,,,,,,,,,,,,,,,,,,,,,,,,,, &
,,,,,,,,,,,,,[!TIME][!ASCII 212 212]&
,,,,,,,,,,,,,[!ASCII 235 224]PASCAL PROGRAMMING MENU&
[!ASCII 234 225][!ASCII 212]&
[!ASCII 212],,,,,,,,,,,,,,{0},,,Bye&
[!ASCII 212],,,,,,,,,,,,,,{1},,,CREATE/EDIT SOURCE FILE&
[!ASCII 212],,,,,,,,,,,,,,{2},,,COMPILE&
[!ASCII 212],,,,,,,,,,,,,,{3},,,LINK&
[!ASCII 212],,,,,,,,,,,,,,{4},,,PRINT LISTING&
[!ASCII 212],,,,,,,,,,,,,,{5},,,EXECUTE PROGRAM&
[!ASCII 212 212 212 235]
WRITE ,,,,,,,,,,,,,,Any other key to repeat menu[!ASCII 212 212]
STRING [!READ ,,,,,,,,,,,,,,ENTER OPTION 0 TO 5: ]
[!EQUAL,( [!STRING]),(0)]

```

```
WRITE [!ASCII 220,221,204,216]&
,{0},,,BYE[!ASCII 217 220 210 226]
PAUSE 1
COMMENT BYE
[!END]
[!EQUAL, ([!STRING]), (1)]
    PASCAL.MENU_1
    %0%/cont
[!END]
[!EQUAL, ([!STRING]), (2)]
    PASCAL.MENU_2
    %0%/cont
[!END]
[!EQUAL, ([!STRING]), (3)]
    PASCAL.MENU_3
    %0%/cont
[!END]
[!EQUAL, ([!STRING]), (4)]
    PASCAL.MENU_4
    %0%/cont
[!END]
[!EQUAL, ([!STRING]), (5)]
    PASCAL.MENU_5
    %0%/cont
[!END]
```



```
comment +-----+
comment |           PASCAL.MENU_1.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|                                     |
|   This macro is called from PASCAL.MENU.CLI.   |
|                                     |
+-----+ [!end]
```

```
PUSH
WRITE [!ASCII 220,221,205,216]&
,{1},,,CREATE/EDIT SOURCE FILE&
[!ASCII 217,220,210,226]
PAUSE 2
X SED [!READ Enter filename to be edited:,,]
POP
```

```
comment +-----+
comment |           PASCAL.MENU_2.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
|   This submacro is a menu option called from PASCAL.MENU.CLI.
|
+-----+-----[!end]
```

```
PUSH
WRITE [!ASCII 220,221,206,216]&
,{2},,,COMPILE&
[!ASCII 217,220,210,226]
PAUSE 2
SEA :UTIL:PASCAL :UTIL:CLRE :UTIL:SWAT :UDD:BARB:MACROS :UTIL :UTIL:MACROS :
STRING [!READ Enter PASCAL filename to be compiled:,,]
DEL/2=IGN =[!STRING].LS
X PASCAL/LINEID/PROCID/DEBUG/L==[!STRING].LS [!STRING]
POP
```

```
comment +-----+
comment |           PASCAL.MENU_3.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
| This menu option is called from PASCAL.MENU.CLI. |
+-----[!end]
```

```
PUSH
WRITE [!ASCII 220,221,207,216]&
,{3},,,LINK&
[!ASCII 217,220,210,226]
PAUSE 2
STRING [!READ Enter PASCAL Filename to be linked:,,]
SEA :UTIL:PASCAL :UTIL:CLRE :UTIL:SWAT [!SEA]
PASLINK/DEBUG [!STRING]
POP
```

```
comment +-----+
comment |          PASCAL.MENU_4.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This menu option is called from PASCAL.MENU.
|-----[!end]
```

```
PUSH
WRITE [!ASCII 220,221,210,216]&
,{4},,,PRINT LISTING&
[!ASCII 217,220,210,226]
PAUSE 2
STRING [!READ Enter PASCAL File name (without .LS ending):,,]
[!NEQ,[!PATH,[!STRING]],]
  QP1/FOLDLONG [!STRING].LS
[!ELSE]
  WRI
  WRI No listing appears in your directory.  Recompile
  WRI your program[!ASCI 254] and then choose this option again.
[!END]
POP
```

```
comment +-----+
comment |           PASCAL.MENU_5.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|                                     |
| This menu option is called from PASCAL.MENU.CLI. |
|-----[!end]
```

```
PUSH
WRITE [!ASCII 220,221,210 232,216]&
,{5},,,EXECUTE PROGRAM&
[!ASCII 217,220,210,226]
PAUSE 2
STRING [!READ Enter PASCAL program to execute:,,]
X [!STRING]
POP
```

```
comment +-----+
comment |          PASSWORD.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
|   This interactive macro asks for the name of a user, and then
|   displays that user's password.  You must have Superuser privilege
|   to be able to use this macro.
|
+-----[!end]
```

```
PUSH
PROMPT POP
SUPERUSER ON
WRI  Enter the Username of the person whose Password you want
STRING [!READ to find out:,,]
:UPD:[!STRING]
WRI
POP
```

```

comment +-----+
comment |           PCT_FULL.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro displays the percentage full of the CPD given as an
| argument.
|
| Format:  PCT_FULL  arg
|
+-----+-----[!end]

```

```

[!EQUAL,%0/LOOP%,]
PUSH
PROMPT POP LEVEL
CLASS(1,2) WARNING
STRING :UDD:[!USER]:?![!PID].%0\%.TMP
ACL/D/L=@NULL [!STRING]
PERM/L=@NULL [!STRING] OFF
DELETE/L=@NULL [!STRING]
SPACE/L=[!STRING] %1%
PUSH
STRING/K
STRING/L=@NULL %2-%
%0%/LOOP %1% [[!STRING/P]]
POP
DELETE/L=@NULL [!STRING]
POP
[!ELSE]
VAR1 [!UDIV,%5%,1000]
VAR2 [!UDIV,%3%,1000]
VAR1 [!UMUL,[!VAR1],100]
VAR0 [!UDIV,[!VAR1],[!VAR2]]
VAR3 [!UDIV,%5%,2000]
WRITE%0/L%
WRITE%0/L%,,,,,[!EQUAL,([!STR]),()]Directory %1%[!ELSE]&
[!STRING][!END] is [!VAR0] percent full&
[!ASCII 254,240,250][!VAR3] megabytes[!ASCII 251].
[!END]

```

```

comment +-----+
comment |           P_MAKER.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This interactive macro creates a new profile using default settings
| of an existing user specified in the /DEF= switch of the macro
| invocation.  If no /DEF= switch then default settings are taken
| from a generic username USR, which must already exist.
| This macro calls MAKER1.CLI
|
| PRIVILEGES: SUPERUSER
|
+-----[!end]

```

```

[!NEQ,%0/CONT%,/CONT]
    PUSH;PROMPT POP

```

```

[!END]

```

```

DEL/2=IGNORE,?[!pid].profile.<in,out>
CREATE/2=IGNORE,(?[!pid].profile.in,?[!pid].profile.out)
LIST,=?[!pid].profile.in

```

```

[!EQUAL,%0/def=%,]
    [!EQUAL,,[!FILENAME,:udd:[!user]:upd:usr]]

```

```

    WRITE This macro requires a username USR in the UPD directory
    WRITE created by creditor in your working dir.  If you have not
    WRITE created such a username then you must use the
    WRITE /DEF= username switch during invocation and the username
    WRITE must exist in your UPD directory.

```

```

[!END]

```

```

[!ELSE] ;COMMENT Ok to proceed

```

```

    WRITE

```

```

    WRITE Do you wish to use a mass password for

```

```

    [!EQUAL,(Y),([!READ,all users being created ? {Y}es or NL,,)]]

```

```

        var0 1 ;COMMENT set flag for MAKER1.CLI

```

```

        WRITE;WRITE The mass password will be 'PASSWORD'

```

```

[!END]

```



```
WRITE
MAKER1%0/%,%-% ;COMMENT recursive macro to create users
WRITE
WRITE The dialog of this profile create session is saved
WRITE in a file called ?[!pid].profile.out.
[!NEQUAL,(Y),([!READ,Do you wish to save it {Y}es or NL:,,)]
    DEL/2=IGNORE,?[!pid].profile.out
[!END]
DEL/2=IGNORE,?[!pid].profile.in
[!END]
POP
```

```
comment +-----+
comment |          READ_UTIL.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro verifies the tape made with DUMP_UTIL.CLI.
| This macro must be run from PID 2.
|
+-----[!end]
```

```
PUSH
PROMPT POP
CHAR/OFF/NRM
CONTROL @EXEC OPERATOR ON
WRITE
WRITE Prepare to verify the dump tape of :UTIL
WRITE
WRITE Sending load to batch stream....
WRITE
QBATCH/M
SUPERUSER ON
DIR :UTIL
X LOAD_II/N/BUFF=8192 @LMT:UTILA1:FILES #
SEND 2 Load complete
)
WRITE
WRITE Load sent to batch stream. Please wait for mount and
WRITE dismount requests from EXEC and respond accordingly.
WRITE
WRITE Type LOAD_UTIL to delete :UTIL and reload it from tape.
POP
```

```

comment +-----+
comment |              REVISIONS.CLI              |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro allows you to easily check the revisions of products
| on the system. To update the list, type REVISIONS/U.
| To just print the list leave off the switch.
| When new software products are added to the system,
| it is necessary to modify the file REVISIONS.PROGRAMS to
| include the pathname to the new product.
|
| Assumptions:  Macros are in :UTIL:MACROS; a directory named
| :UTIL:MACROS:REVISIONS exists; access to SORT/MERGE; access to
| SPEED.
|
| PRIVILEGES:  SUPERUSER
|
+-----[!end]

```

```

PUSH
PROMPT POP
SUPERUSER ON
[!EQ,%/U%,/U]
  PUSH
  DIR :UTIL:MACROS:REVISIONS
  DEL/1=IG/2=IG :UTIL:MACROS:REVISIONS:REVLIST
  WRITE/L=:UTIL:MACROS:REVISIONS:REVLIST LAST UPDATE [!DATE] [!TIME]
  WRITE/L=:UTIL:MACROS:REVISIONS:REVLIST
  WRITE;WRITE Please wait ... Determining latest revisions...
  REV/1=I/2=I/V/L=:UTIL:MACROS:REVISIONS:?REVLIST.TMP ([REVISIONS.PROGRAMS])
  X SPEED/I=REVISIONS.SCF :UTIL:MACROS:REVISIONS:?REVLIST.TMP
  WRITE;WRITE ,,,,,,,Sorting results into new file...
  REVISIONS.SORT
  COPY/A :UTIL:MACROS:REVISIONS:REVLIST :UTIL:MACROS:REVISIONS:?REVLIST.SORT.TMP
  WRITE;WRITE Compiling updated list...
  DEL/1=IG/2=IG :UTIL:MACROS:REVISIONS:?REVL+.TMP
  ACL :UTIL:MACROS:REVISIONS:REVLIST OP,OWARE +,RE
  POP
[!END]

```


REVISIONS.PROGRAMS

This text file is used by REVISIONS.CLI. It contains the names of all the program files that the user wants the revision number.

*** Note: This box is not part of the text file.***

:AGENT.PR &
:CLI.PR &
:UTIL: BASIC: BASIC.PR &
:UTIL: C: CC.PR &
:UTIL: CEO_DIR: CEO_CP.PR &
:UTIL: CEO_DIR: CEO_SPELL.PR &
:UTIL: CEO_DIR: CDB_SPD.PR &
:UTIL: COBOL: COBOL.PR &
:UTIL: COMPUCALC: CEO_COMPUCALC.PR &
:UTIL: DBMS: DB.CONTRÖLLER.PR &
:UTIL: DBMS: DB.DDF.PR &
:UTIL: DBMS: DB.RDA.PR &
:UTIL: F77: F77.PR &
:UTIL: INFOS: INFOS_GS.PR &
:UTIL: ICOBOL: ICOBOL.PR &
:UTIL: PASCAL: PASCAL.PR &
:UTIL: PL1: PL1.PR &
:UTIL: PRESENT: PRESENT.PR &
:UTIL: PROXI: PROXI\$00.PR &
:UTIL: SGU: SGU.PR &
:UTIL: SORT: SORT.PR &
:UTIL: SWAT: SWAT.PR &
:UTIL: TRENDVIEW: TRENDVIEW.PR &
:UTIL: TPMS: TPMSTCP.PR &
:NET: RIA.PR &
:NET: X25_LMGR.PR &
:UTIL: ADE: PROGRAMS: ADB_AGENT.PR &
:UTIL: DGL: DGL.PR &
:UTIL: ICOBOL: MLS.PR &
:UTIL: MONITOR: MONITOR.PR &

:UTIL:SQL:SQL1.PR &
:UNIX:BIN:UNIX_DEMON.PR &
:UTIL:COMLOG:COMLOGVS.PR &
:UTIL:RDOS_AGENT:RDOS_AGENT.PR &
:UTIL:SQL:SQL2.PR &

REVISIONS.SCF

This is a speed command file called from the macro REVISIONS.CLI.

*** Note: This box is not part of the text file.***

```
<C ;>
<C
  ;>
<C:
  :;>
<C ;>
<C:SYSGEN:;>
<C:UTIL:;>
<C:NET:;>
<S.SY;30-VM<I >;>
<S.PR;30-VM<I >;>
FUH
```

REVISIONS.SMF

This is a SORT/MERGE command file called from REVISIONS.CLI

*** Note: This box is not part of the text file.***

INPUT FILE IS "?REVLIST.TMP"

,RECORDS ARE DATASENSITIVE UPTO 80 CHARACTERS.

OUTPUT FILE IS "?REVLIST.SORT.TMP".

KEY 1/3.

SORT.

END.


```
comment +-----+
comment |           REVISIONS.SORT.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
| This submacro is called from REVISIONS.CLI. |
+-----[!end]
```

```
DEL/1=IG/2=IG REVISIONS.<SMF.LS SMF.SORT>
SORT/C=REVISIONS.SMF/O/L=REVISIONS.SMF.LS
```

```
comment +-----+
comment |           SETUP.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro is the Initial IPC file executed before the User's
| individual IPC file.
|
| Assumptions:  :UTIL:NEWS.TXT contains text to be displayed
| whenever a user logs onto the system.
|
+-----[!end]
```

```
SEARCHLIST :UTIL : :UTIL:MACROS :UTIL:CEO_DIR
[!EQUAL,CONSOLE,[!LOGON]]
  [!NEQUAL,[!SIZE :UTIL:NEWS.TXT],0]
  WRITE,[!ASCII,214,207]
  TYPE/1=ERROR/2=IGNORE,:UTIL:NEWS.TXT
  [!END]
  WRITE [!ASCII 212]
  [!EQUAL,,[!FILE CONSOLE_LOGON.CLI]]
  WRITE * You can create a file named CONSOLE_LOGON.CLI, which will
  WRITE * invoke whatever functions desired at logon time,
  WRITE * e.g., modifying your Searchlist, changing your prefix
  WRITE * and prompt.
  WRITE
  [!ELSE]
  CONSOLE_LOGON
  [!END]
  WRITE
  WRITE OPERATOR IS [!OPERATOR] DUTY
  WRITE
  WRITE
[!ELSE]
  PRIORITY/2=IGNORE/1=IGNORE [!PID] 3
[!END]
```

```
comment +-----+
comment |          SHUTDOWN.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This interactive macro assists the Operator in shutting down the
| system by broadcasting system shutdown at regular intervals.
| It calls SYSTEM.CLI.
|
+-----[!end]
```

```
[!UEQUAL,[!PID],2]
    PUSH
    PROMPT POP
    VAR0 [!READ, Enter minutes before System Shutdown:,,]
    VAR1 [!READ, Enter minutes to decrement the time and display shutdown message:,,]
    SYSTEM
    POP
[!ELSE]
    WRI
    WRI Only PID 2 can use this macro.
[!END]
```

```
comment +-----+
comment |           SOFF.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro displays a fake disable message on a terminal. To
| exit the macro, press the BREAK sequence, and then ^C^A.
|
+-----[!end]
```

```
WRITE [!ASCII 214,212,212,212,212]
WRITE [!ASCII 234] ---- [!HOST]/[!ASCI 216]VS[!ASCI 217] - &
      [!ASCI 235],/,Console DISABLED from logging on ----[!ASCII 235]
COPY/A/B @NULL @CONSOLE
```

```
comment +-----+
comment |          SPACE.LIST.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This interactive macro will list how much Space is being
| used in :UDD and :CEO_FILES.  You must have Superuser privilege
| to use the macro.
|
+-----[!end]
```

```
PUSH
PROMPT POP
SUPERUSER ON
STRING [!READ Enter the name of the person requesting the SPACE LIST: ]
DELETE/2=IGNORE [!DIR/P]:[!HOST].SPACE.LIST
DIR :UDD
WRI/L=[!DIR/P]:[!HOST].SPACE.LIST
WRI/L=[!DIR/P]:[!HOST].SPACE.LIST,,,,,,,,,SPACE USED IN :UDD,,,[!DATE],,,[!TIME]
SPACE/V/2=IGNORE/L=[!DIR/P]:[!HOST].SPACE.LIST +
DIR :CEO_FILES
WRI/L=[!DIR/P]:[!HOST].SPACE.LIST,,,,,,,,,SPACE USED IN :CEO_FILES,,,[!DATE],,,[!TIME]
SPACE/V/2=IGNORE/L=[!DIR/P]:[!HOST].SPACE.LIST +
QPRINT/TITLES/DELETE/DEST=[!STRING] [!DIR/P]:[!HOST].SPACE.LIST
POP
```

```

comment +-----+
comment |          SYSMAP.CLI          |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro dirs into specified system directories and compiles a
| list of SPACE and FILESTATUS information about the directories.
| It then calls the list up to print at LPT.
|
| Assumptions: Directory structure same as specified in DIRECTORY
| command.
|
| PRIVILEGES: SUPERUSER, BATCH
|
+-----[!end]

```

```

SUPERUSER ON
QBATCH/QPRIORITY=255/M/QOUT=:UTIL:?.SYSMAP.QOUT
SUPERUSER ON
DIR :

```

```

LISTFILE :UTIL:SYSMAP
WRITE/L [!ASC 212 212 212 212]
WRITE/L *****
WRITE/L ////////////////////////////////////////////////////////////////////
WRITE/L //,,,,,[!HOST] UTILIZATION MAP
WRITE/L //:
WRITE/L ////////////////////////////////////////////////////////////////////
WRITE/L *****
WRITE/L [!ASC 212 212]

```

```

DIR :
SPACE/2=IGNORE/V/L
WRITE/L [!ASC 212 212 212]
F/S/AS/L +
WRITE/L [!ASCII 214]

```

```

DIR :NET
WRITE/L [!ASC 212 212 212]
WRITE/L *****
WRITE/L //////////////////////////////////////
WRITE/L //, [!HOST] UTILIZATION MAP
WRITE/L //, NET
WRITE/L //////////////////////////////////////
WRITE/L *****
WRITE/L [!ASC 212 212]
SPACE/V/L
WRITE/L [!ASC 212 212 212]
F/S/AS/L +
WRITE/L [!ascii 214]

DIR :UDD
WRITE/L *****
WRITE/L //////////////////////////////////////
WRITE/L //, [!HOST] UTILIZATION MAP
WRITE/L //, :UDD
WRITE/L //////////////////////////////////////
WRITE/L *****
WRITE/L [!ASC 212 212]
SPACE/V/L
WRITE/L [!ASC 212 212 212]
F/S/AS/L +
WRITE/L [!asc 214]

DIR :UTIL
WRITE/L [!ASC 212 212 212]
WRITE/L *****
WRITE/L //////////////////////////////////////
WRITE/L //, [!HOST] UTILIZATION MAP
WRITE/L //, FOR :UTIL
WRITE/L //////////////////////////////////////
WRITE/L *****
WRITE/L
WRITE/L
SPACE/V/L
WRITE/L [!ASC 212 212 212]
F/S/AS/L +
WRITE/L [!ascii 214]

```

```

DIR :UTIL:MACROS
WRITE/L [!ASC 212 212 212]
WRITE/L *****
WRITE/L ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
WRITE/L,,,,,,,,,,,,,[!HOST] UTILIZATION MAP
WRITE/L,,,,,,,,,,,,,FOR :UTIL:MACROS
WRITE/L,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
WRITE/L *****
WRITE/L [!ASC 212 212]
SPACE/V/L
WRITE/L [!ASC 212 212 212]
F/S/AS/L
WRITE/L [!ascii 214]
DIR :UTIL:BASIC
WRITE/L [!ASC 212 212 212]
WRITE/L *****
WRITE/L ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
WRITE/L,,,,,,,,,,,,,[!HOST] UTILIZATION MAP
WRITE/L,,,,,,,,,,,,,FOR :UTIL:BASIC
WRITE/L,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
WRITE/L *****
WRITE/L
WRITE/L
SPACE/V/L
WRITE/L [!ASC 212 212 212]
F/S/AS/L +
WRITE/L [!ascii 214]
DIR :UTIL:COBOL
WRITE/L [!ASC 212 212 212]
WRITE/L *****
WRITE/L ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
WRITE/L,,,,,,,,,,,,,[!HOST] UTILIZATION MAP
WRITE/L,,,,,,,,,,,,,FOR :UTIL:COBOL
WRITE/L,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
WRITE/L *****
WRITE/L [!ASC 212 212 212]
SPACE/V/L
WRITE/L [!ASC 212 212 212]
SPACE/V/L +
WRITE/L [!ASC 212 212 212]
F/S/AS/L +
WRITE/L [!ascii 214]

```



```
DIR :UTIL:DBMS
  WRITE/L [!ASC 212 212 212]
  WRITE/L *****
  WRITE/L '
  WRITE/L, , , [!HOST] UTILIZATION MAP
  WRITE/L, , , FOR :UTIL:DBMS
  WRITE/L '
  WRITE/L *****
  WRITE/L
  WRITE/L
  SPACE/V/L
  WRITE/L [!ASC 212 212 212]
  F/AS/S/L +
  WRITE/L [!ascii 214]
DIR :UTIL:INFOS
  WRITE/L [!ASC 212 212 212]
  WRITE/L *****
  WRITE/L '
  WRITE/L, , , [!HOST] UTILIZATION MAP
  WRITE/L, , , FOR :UTIL:INFOS
  WRITE/L '
  WRITE/L *****
  WRITE/L
  WRITE/L
  SPACE/V/L
  WRITE/L [!ASC 212 212 212]
  F/S/AS/L
  WRITE/L [!ascii 214]
  LISTFILE/K
QPR/DELETE/DESTINATION=SYSMAP :UTIL:SYSMAP
)
```

```
comment +-----+
comment |          SYSTEM.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro is called by SHUTDOWN.CLI. It calculates when
| the BROADCAST message should be sent to users to log off the
| system, and sends the message.
|
+-----[!end]
```

```
[!NEQ,0,[!VAR0]]
SEND/2=IGNORE [!PID],[!TIME]: System Shutdown in [!VAR0] mts. Please log off.
    VAR0,[!USUB,[!VAR0],[!VAR1]]
    PAUSE [!UMULT,60,[!VAR1]]
    SYSTEM
[!ELSE]
    Wri
    Wri You can bring down the system now.
[!END]
```

```
comment +-----+
comment |          SYSUP.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro will tell the user how long the system has been up.
| It calls SYSUP2.CLI AND SYSUP3.CLI.
|
+-----[!end]
```

```
PUSH; PROMPT POP
DEL/2=IGNORE :util:rtemp
RUN/L=:UDD:[!USER]:RTEMP 1
STR :UDD:[!USER]:RTEMP
SYSUP2 [:UDD:[!USER]:RTEMP]
DEL/2=IG :UDD:[!USER]:RTEMP
POP
```

```
comment +-----+
comment |           SYSUP2.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
| This macro is called from SYSUP.CLI.  It explodes the elapsed time |
| from the RUNTIME command.  It passes this onto SYSUP3.CLI         |
+-----+-----[!end]
```

```
sysup3 [!explode %2%]
```

```

comment +-----+
comment |                SYSUP3.CLI                |
comment +-----+

```

```

[!equal,large,comment]-----+
|
|   This macro is called from SYSUP2.CLI (which is called from
|   SYSUP.CLI).  It calculates how long the system has been up
|   in a user-friendly format.
|
+-----[!end]

```

```

var0 377
[!eq,%2%,:]
  str %1%
  var0 %3%%4%
  var1 %6%%7%
[!else]
  [!eq,%3%,:]
    str %1%%2%
    var0 %4%%5%
    var1 %7%%8%
  [!else]
    [!eq,%4%,:]
      str %1%%2%%3%
      var0 %5%%6%
      var1 %8%%9%
    [!else]
      wr ERROR: LOKI has not been up for 40 DAYS and 40 NIGHTS!
    [!end]
  [!end]
[!end]

[!neq,[!var0],377]
  wr [!host] has been up for [!udiv [!str] 24] DAYS [!umod [!str] 24] HOURS &
    [!var0] MINS [!var1] SECS.
[!end]

```

```
comment +-----+
comment |          TESTABORT.CLI          |
comment +-----+
```

```
[!equal,large,comment]-----+
|
|   This interactive macro executes a first program.  If that
|   program executes successfully, it will execute a second program.
|
+-----[!end]
```

```
PUSH
PROMPT POP
WRI Beginning Execution  [!DATE] [!TIME]
WRITE You are now in Step 1
XEQ/S/1=WARNING [!READ Enter first program's name:,,]
COMMENT Check now to see if String still blank
[!EQ,[!STRING],]
    WRITE You are now in Step 2
    WRITE The first program ran okay
    XEQ/S/1=WARNING [!READ Enter second program's name:,,]
[!ELSE]
    WRITE
    WRITE The first program aborted
    WRITE
[!END]
WRI Ending Execution  [!DATE] [!TIME]
POP
```

```

comment +-----+
comment |           TT.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro is used for maintaining a personal telephone
| in :UDD:[!USERNAME]:TELEPHONES.DR.
|
| It accepts the following switches:
|
|   /ADD  To Create an entry with Argument 1 (person's name) and
|         argument 2 (phone number) as link name
|
|   /DEL  Uses Argument 1 (person's name) to delete them from
|         your telephone directory
|
|   /SEE  Uses Argument 1 (person's name) to perform a Filestatus
|         If no Argument 1 passed in, does a Filestatus on phone
|         directory
|
|   /PRINT Prints your telephone directory to LPT
|
+-----[!end]

```

```

CRE/2=IGN/DIR :UDD:[!USER]:TELEPHONES.DR
[!EQUAL,%/DEL%,/DEL]
  PUSH
  F/ASSORTMENT/SORT :UDD:[!USERNAME]:TELEPHONES.DR:%1-%
  STRING [!READ Is this what you want to delete? (Y/N),,]
  [!EQUAL [!STRING],Y]
  DELETE/V :UDD:[!USERNAME]:TELEPHONES.DR:%1-%
  [!END]
  POP
[!ELSE]
[!EQUAL,%/SEE%,/SEE]
  [!EQ,%1%,]
  F/ASSORTMENT/SORT :UDD:[!USERNAME]:TELEPHONES.DR:+
[!ELSE]
  F/ASSORTMENT/SORT :UDD:[!USERNAME]:TELEPHONES.DR:%1-%
[!END]

```

```
[!ELSE]
[!EQUAL,%/ADD%,/ADD]
  CREATE/LINK :UDD:[!USERNAME]:TELEPHONES.DR:%1% %2%
  F/ASSORTMENT/SORT :UDD:[!USERNAME]:TELEPHONES.DR:%1%
[!ELSE]
[!EQUAL,%/PRINT%,/PRINT]
  F/ASSORTMENT/SORT/L=[!USER].TMP :UDD:[!USERNAME]:TELEPHONES.DR:+
  QPRINT/DELETE [!USER].TMP
[!END]
[!END]
[!END]
[!END]
```



```

comment +-----+
comment |                UDDBACKUP.CLI                |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro will do a full dump of UDD directory as a batch job, |
| on labeled tapes. It assumes tapes are labeled ARUDF0, ARUDF1, |
| ARUDF2, ARUDF3, ARUDF4, and ARUDF6.                          |
|
| Assumption: UDDBACKUP.CLI is in :UTIL:BACKUP.                 |
|
| PRIVILEGES: SUPERUSER                                         |
|
+-----+-----[!end]

```

```

[!NE,[!OPERERATOR],ON]
WRITE [!ASC 207] JOB WON'T START UNTIL OPERATOR IS ON DUTY!
[!END]

```

```

WRITE This will be a full dump of the UDD directories.
write The last full dump was taken on [last_udd_dump]
write

```

```

[!EQUAL,([!FILENAME :UDD:[!USER]:TAPE]),()]
PUSH
PROMPT POP
SUPERUSER ON

QB/M/OP
SUPERUSER ON
MOUNT/EXTEND/VOLID=ARUDF0/VOLID=ARUDF1/VOLID=ARUDF2/VOLID=ARUDF3&
/VOLID=ARUDF4/VOLID=ARUDF5/VOLID=ARUDF6 TAPE Please mount.
DIR :UDD
X DUMP_II/DEN=1600/BUFF=8192/RETAIN=75/MEM=MAX :UDD:[!USER]:TAPE:BACKUP
DIR :UTIL:BACKUP
DEL/1=IGN/2=IGN LAST_UDD_DUMP
WRITE/L=LAST_UDD_DUMP [!DATE]:[!TIME]
ACL last_udd_dump + re
)

```

```
[!ELSE]
  WRITE [!asc 207] The file 'TAPE' cannot exist in your initial
  write directory ':UDD:[!user]'. Delete or rename then restart this macro.
[!end]
```

```
comment +-----+
comment |           USERS.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
|
| This macro will calculate the number of actual users logged on the |
| system.  It looks for Simple Process Names of CONn.              |
|
| This driver macro calls USERS.IN and uses SORT.                  |
|
+-----[!end]
```

```
WHO/2=IGN/L==POOL.FILE <,1,2><0 1 2 3 4 5 6 7 8 9><0 1 2 3 4 5 6 7 8 9>
PROC/DEF/BLOCK/INPUT=USERS.IN/OUTPUT=@null SORT/C/O
WR The number of users on the system is: [!udiv,[!size POOL.FILE.OUT],80]
WR
TYPE =POOL.FILE.OUT
DEL/2=IGN =POOL.FILE =POOL.FILE.OUT
[!READ Press the NEW-LINE key to return to your prompt:]
```

USERS.IN

This is a SORT/MERGE command file used in USERS.CLI.

*** Note: This box is not part of the text file.***

INPUT FILE IS "POOL.FILE",
RECS ARE DATA SENS DELIMITED BY "<012>" UPTO 180 CHARS.
OUTPUT FILE IS "POOL.FILE.OUT".
REPLACE ALL "<012>" IN 1/LAST WITH "<040>".
PAD TO 80 CHARACTERS WITH "<040>".
INSERT "<012>" IN 80/80.
IF 28/30:<>:"CON" AND 29/31:<>:"CON" THEN SKIP.
IF 50/53:="XLPT" THEN SKIP.
COPY.
END.

```
comment +-----+
comment |           WHERE.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
| This macro displays the current host and directory. |
+-----[!end]
```

```
HOST
DIRECTORY
```

```
comment +-----+
comment |           WHO?.CLI           |
comment +-----+
```

```
[!equal,large,comment]-----+
| This macro displays information about the current user. |
+-----[!end]
```

```
WRITE [!USER]
WRITE [!CONSOLE]
WRITE [!PID]
```

```

comment +-----+
comment |           YES_OR_NO.CLI           |
comment +-----+

```

```

[!equal,large,comment]-----+
|
| This macro is meant to be called as a submacro. It accepts a
| question from the driver macro and performs the READ statement
| here. It tests for Y, YE, YES, N, or NO and returns either a YES
| or NO in the STRING. It repeats the question until an answer using
| the above format is given.
|
+-----+ [!end]

```

```

STRING [!READ %1-%,, ,]
[!EQUAL, (YES), ([!STRING])]
  STRING YES
  [!ELSE]
    [!EQUAL, (YE), ([!STRING])]
      STRING YES
      [!ELSE]
        [!EQUAL, (Y), ([!STRING])]
          STRING YES
          [!ELSE]
            [!EQUAL, (NO), ([!STRING])]
              STRING NO
              [!ELSE]
                [!EQUAL, (N), ([!STRING])]
                  STRING NO
                  [!END]
                [!END]
              [!END]
            [!END]
          [!END]
        [!END]
      [!END]
    [!END]
  [!END]
[!NEQ, (NO), ([!STRING])]
[!NEQ, (YES), ([!STRING])]
  WRITE
  %0% %1-%
[!END]
[!END]

```