

```

SUBROUTINE DRAW3D(Z, IDIM)
C
C
C
C
000005  CDE PACKAGE
000005  DIMENSION Z(IDIM,10)
000005  COMMON/COM3DP/XMIN,XMAX,YMIN,YMAX,ZMIN,ZMAX,ZLIM(2),PLOTX,PLOT
000005  1PLOTORG(2),CAMXYZ(3),MX,NY,FMX,FNY,KBEAM,ZETA,CAMWKG(32),XORG(
000005  2GX(3),FX(2),KSCALE,ZORG,NOUT,CENTER(2),PQLIMIT
000005  DIMENSION LIMIT(2),FLIM(2),AMTX(3,3),KCAMWKG(32)
000005  0EQUIVALENCE(U,CAMXYZ),(V,CAMXYZ(2)),(W,CAMXYZ(3)),(MX,LIMIT),
1(FMX,FLIM),(AMTX,CAMWKG(17)),(CAMWKG,KCAMWKG)
C
C
C
C
000005  CDE PACKAGE FOR DRAW3D, DRAWPQ, GNOMON
000005  COMMON/COM3DPA/PC(3),QC(3),P(3),Q(3),ENDA(6),ENDB(6),OLDDQ(4),
000005  1PW(3),QW(3),T(6),PK(3),QK(3),PHIP,PHIQ,PHIA,IBEAM,KDIM
C
C
C
C
000005  INTEGER PHIP,PHIQ,PHIA
000005  END CDE PACKAGE

C
C
C
C
000005  SAVE Z DIMENSION IN COMMON TO PASS ALONG THROUGH DRAWPQ TO IVI
000005  KDIM=IDIM
C
C
C
C
000006  INDEX OF COORDINATE BEING STEPPED ALONG A LINE
000006  KSCAN=1
C
000007  INDEX OF COORDINATE BEING HELD FIXED
000007  KFIX=2
C
000010  SET FIXED COORDINATE + INCREMENT
000010  PC(KFIX)=1,0
000012  DELFIX=1,0
C
000013  SET ROVING COORDINATE + INCREMENT INITIALLY
000014  DELSCAN=1,0
000014  QC(KSCAN)=1,0
C
000015  BEGIN SCANNING A LINE
000015  101 QC(KFIX)=PC(KFIX)
000017  IBEAM=0
C
000020  NEXT POINT IN LINE SCAN
000020  102 PC(KSCAN)=QC(KSCAN)
000022  QC(KSCAN)=PC(KSCAN)+DELSKAN
C
000025  WORKING INDICES
000025  JPC=IFIX(PC(1))
000026  KPC=IFIX(PC(2))
000030  JQC=IFIX(QC(1))
000032  KQC=IFIX(QC(2))
C
000034  PHI FUNCTIONS
000034  PC(3)=Z(JPC,KPC)
000041  PHIP=PC(3),AND,000000000003B
000043  PHIP=PHIP-1
000045  QC(3)=Z(JQC,KQC)
000051  PHIQ=QC(3),AND,000000000003B
000053  PHIQ=PHIQ-1
000055  CALL DRAWPQ(Z)
C
000056  TEST IF LINE IS DONE
000056  IF((QC(KSCAN)-1,0)*(QC(KSCAN)-FLIM(KSCAN)),LT,0,0)GO TO 102
C
000066  LINE DONE, ADVANCE FIXED COORDINATE,
000066  PC(KFIX)=PC(KFIX)+DELFIX

```

```

000071 C TEST IF FIXED COORDINATE NOW OFF LIMITS
IF((PC(KFIX)-1,0)*(PC(KFIX)-FLIM(KFIX)).GT,0,0)GO TO 55
000077 C FLIP INCREMENT, SCAN BEGINS AT QC OF PREVIOUS LINE,
DELSKAN=-DELSKAN
000100 GO TO 101
000100 C TEST IF WE HAVE DONE Y SCAN YET,
55 IF(KSCAN,EQ,2)RETURN
000103 C NO, SCAN Y DIRECTION AT FIXED X,
KSCAN=2
000104 KFIX=1
000105 C START FIXED X AT X OF LAST TRAVERSE
PC(1)=QC(1)
000107 C THEN STEP X IN OPPOSITE DIRECTION
DELFIX=-DELSKAN
000110 C WE ENDED UP AT MAX, Y, SO FIRST Y SCAN GOES BACKWARDS
DELSKAN=-1,0
000111 C INITIAL Y FOR FIRST LINE
QC(2)=FNY
000113 GO TO 101
000113 END

```

*FOLLOWING VARIABLES EQUIVALENCED BUT NOT REFERENCED

V
W
U
MX
FMX

RUN VERSION 2,3 - CDC LEVEL 26 - NYU LEVEL 11

DRAW3D

SUBPROGRAM LENGTH
000152

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

55 - 000100 101 - 000015 102 - 000020

BLOCK NAMES AND LENGTHS

COM3DP - 000103 COM3DPA - 000063

VARIABLE ASSIGNMENTS

AMTX	-	000045C01	CAMWKG	-	000025C01	CAMXYZ	-	000014C01	CENTER	-	000100C01
DELIX	-	000144	DELSCAN	-	000145	ENDA	-	000014C02	ENDB	-	000022C01
FLIM	-	000021C01	FNY	-	000022C01	FX	-	000073C01	GX	-	000070C01
IBEAM	-	000061C02	JPC	-	000146	JQC	-	000150	KCAMWKG	-	000025C01
KDIM	-	000062C02	KFIX	-	000143	KPC	-	000147	KQC	-	000151
KSCAN	-	000142	LIMIT	-	000017C01	OLDQ	-	000030C02	P	-	000006C01
PC	-	000000C02	PHIA	-	000060C02	PHIP	-	000056C02	PHIQ	-	000057C01
PK	-	000050C02	PLOTORG	-	000012C01	PK	-	000034C02	Q	-	000011C01
QC	-	000003C02	QK	-	000053C02	QW	-	000037C02	T	-	000042C01
XORG	-	000065C01	ZLIM	-	000006C01						

START OF CONSTANTS
000115

START OF TEMPORARIES
000122

START OF INDIRECTS
000134

UNUSED COMPILER SPACE
022600

```

SUBROUTINE GAUFFRE(NUMGRID,Z,IDIM)
DRAW GRIDS OF LINES ON SUITABLE SIDES OF THE BOX

```

```

C
C
C
C

```

```

CDE PACKAGE

```

```

000006
000006

```

```

DIMENSION Z(IDIM,10)
COMMON/COM3DP/XMIN,XMAX,YMIN,YMAX,ZMIN,ZMAX,ZLIM(2),PLOTX,PLOTY
1PLOTORG(2),CAMXYZ(3),MX,NY,FMX,FNY,KBEAM,ZETA,CAMWKG(32),XORG(3)
2GX(3),FX(2),KSCALE,ZORG,NOUT,CENTER(2),PQLIMIT
DIMENSION LIMIT(2),FLIM(2),AMTX(3,3),KCAMWKG(32)
0EQUIVALENCE(U,CAMXYZ),(V,CAMXYZ(2)),(W,CAMXYZ(3)),(MX,LIMIT),
1(FMX,FLIM),(AMTX,CAMWKG(17)),(CAMWKG,KCAMWKG)
END CDE PACKAGE

```

```

C
C
C
C

```

```

LOCAL CDE

```

```

000006
000006
000006
000006

```

```

DIMENSION WLIM(2,3),CF(2),START(3),END(3),NUMGRID(4)
EQUIVALENCE(END,START(4))
INTEGER CYCLUS(3)
DATA CYCLUS/2,3,1/,CF/1.0,-1.0/

```

```

C
C

```

```

000006
000010
000013
000016

```

```

IF(NUMGRID(1),LE,0)RETURN
NUMBER POINTS TO CHECK FOR VISIBILITY
MX32=MX*3/2
NY32=NY*3/2
NP=MAX0(MX32,NY32,64)

```

```

C
C
C
C

```

```

000024
000026
000027
000030
000031
000034

```

```

X AND Y LIMITS EXPRESSED AS INDICES OF ARRAY Z, Z LIMIT IN SAME
UNITS AS VALUES OF Z IN THE ARRAY, CAMXYZ ARE THE CAMERA LENS
COORDINATES EXPRESSED THE SAME WAY.
WLIM(1,1)=1,0
WLIM(2,1)=FMX
WLIM(1,2)=1,0
WLIM(2,2)=FNY
WLIM(1,3)=(ZMIN-ZORG)/GX(3)
WLIM(2,3)=(ZMAX-ZORG)/GX(3)

```

```

C
C
C

```

```

000037
000040
000042

```

```

IF XI IS WITHIN LIMITS, DRAW GRID IN XJ-XK PLANE AS IT WILL NOT
COME BETWEEN CAMERA AND FIGURE
DO100I=1,3
KY=CYCLUS(I)
KZ=CYCLUS(KY)

```

```

C
C

```

```

000044

```

```

L=1 FOR BEING ABOVE LOWER LIMIT, 2 FOR BEING BELOW UPPER
DO100L=1,2

```

```

C
C

```

```

000045
000046

```

```

PREPARE TO DRAW JUST EDGES OF BOX
IY=2
IZ=2

```

```

C
C

```

```

000047
000050

```

```

IF GNOMON GETS JP=1, DOES NOT EDIT FOR VISIBILITY
JP=1
IF(CF(L)*(CAMXYZ(I)-WLIM(L,1)),LE,0.0)GO TO 3

```

```

C
C

```

```

000055
000057
000061
000065

```

```

IN BOUNDS, GET COORDINATES + NUMBER OF DIVISIONS,
NUMBER GRID LINES = 1+ NUMBER OF DIVISIONS
IY=1+NUMGRID(KY+1)
IZ=1+NUMGRID(KZ+1)
IY=MAX0(2,IY)
IZ=MAX0(2,IZ)

```

```

000070      JP=NP
000071      C      FIXED COORDINATE PLANE IS XI
000072      3      START(I)=WLIM(L,I)
000073      END(I)=START(I)
000074      C      LINES OF CONSTANT KY AS KZ MOVES
000075      STY=WLIM(1,KY)
000101      DY=WLIM(2,KY)-WLIM(1,KY)
000104      DO4ITER=1,2
000106      START(KZ)=WLIM(1,KZ)
000111      END(KZ)=WLIM(2,KZ)
000113      FIY=FLOAT(IY-1)
000116      DO5J=1,IY
000117      C      REVERSE Z ENDS AT EACH TRAVERSE
000121      TEMP=START(KZ)
000122      START(KZ)=END(KZ)
000123      END(KZ)=TEMP
000125      START(KY)=STY+FLOAT(J-1)*DY/FIY
000133      END(KY)=START(KY)
000135      5      CALL GNOMON(START,JP,Z,IDIM)
000136      C      NOW CONSTANT KZ AS KY MOVES, INITIAL KZ TO BE LAST KZ USED,
000137      C      ARRANGE SO INITIAL FLIP OF KY ENDS CAUSES US TO START AT LAST I
000146      DY=START(KZ)-END(KZ)
000150      STY=END(KZ)
000152      KTEMP=KZ
000153      KZ=KY
000154      KY=KTEMP
000155      IY=IZ
000156      4      CONTINUE
000160      100    CONTINUE
000164      RETURN
000165      END

```

*FOLLOWING VARIABLES EQUIVALENCED BUT NOT REFERENCED

V
W
U

RUN VERSION 2,3 - CDC LEVEL 26 - NYU LEVEL 11

GAUFFRE

SUBPROGRAM LENGTH
000253

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
3 - 000071

BLOCK NAMES AND LENGTHS
COM2DP - 000103

VARIABLE ASSIGNMENTS

AMTX	-	000045C01	CAMWKG	-	000025C01	CAMXYZ	-	000014C01	CENTER	-	000100C01
CF	-	000225	CYCLUS	-	000227	DY	-	000245	END	-	000214
FIY	-	000247	FLIM	-	000021C01	FMX	-	000021C01	FNY	-	000022C01
FX	-	000073C01	GX	-	000070C01	I	-	000235	ITER	-	000246
IY	-	000241	IZ	-	000242	J	-	000250	JP	-	000243
KCAMWKG	-	000025C01	KTEMP	-	000252	KY	-	000236	KZ	-	000237
L	-	000240	LIMIT	-	000017C01	MX	-	000017C01	MX32	-	000232
NP	-	000234	NY	-	000020C01	NY32	-	000233	PLOTORG	-	000012C01
START	-	000211	STY	-	000244	TEMP	-	000251	WLIM	-	000217
XORG	-	000065C01	ZLIM	-	000006C01	ZMAX	-	000005C01	ZMIN	-	000004C01
ZORG	-	000076C01									

START OF CONSTANTS
000167

START OF TEMPORARIES
000172

START OF INDIRECTS
000202

UNUSED COMPILER SPACE
022400

```

SUBROUTINE GNOMON(XY,KP,Z,IDIM)
C   DRAW LINE FROM POINT XY(J,1) TO XY(J,2). DIVIDE INTO NP+1 SEGMENTS
C   AND CHECK VISIBILITY AT EACH DIVISION.
C
C   CDE PACKAGE
000007   DIMENSION Z(IDIM,10)
000007   COMMON/COM3DP/XMIN,XMAX,YMIN,YMAX,ZMIN,ZMAX,ZLIM(2),PLOTX,PLOTY
1PLOTORG(2),CAMXYZ(3),MX,NY,FMX,FNY,KBEAM,ZETA,CAMWKG(32),XORG(3)
2GX(3),FX(2),KSCALE,ZORG,NOUT,CENTER(2),PQLIMIT
000007   DIMENSION LIMIT(2),FLIM(2),AMTX(3,3),KCAMWKG(32)
000007   EQUIVALENCE(U,CAMXYZ),(V,CAMXYZ(2)),(W,CAMXYZ(3)),(MX,LIMIT),
1(FMX,FLIM),(AMTX,CAMWKG(17)),(CAMWKG,KCAMWKG)
C   END CDE PACKAGE
C
C   CDE PACKAGE FOR DRAW3D, DRAWPQ, GNOMON
000007   COMMON/COM3DPA/PC(3),QC(3),P(3),Q(3),ENDA(6),ENDB(6),OLDQ(4),
1PW(3),QW(3),T(6),PK(3),QK(3),PHIP,PHIQ,PHIA,IBEAM,KDIM
000007   INTEGER PHIP,PHIQ,PHIA
C   END CDE PACKAGE
C
C   LOCAL CDE
000007   DIMENSION DELTA(3),XY(3,2),XP(2,2),XYC(3,2),DC(3),XCHOP(2,6),
1XYW(3,2),FL(2)
C
C   PICK UP Z DIMENSION TO PASS ON TO IVIS
000007   KDIM=IDIM
C
C   TEST TO FIND PART OF LINE VISIBLE IN GIVEN PICTURE DIMENSIONS
C
000010   DELTA(1)=XY(1,2)-XY(1,1)
000012   DELTA(2)=XY(2,2)-XY(2,1)
000014   DELTA(3)=XY(3,2)-XY(3,1)
000017   DO2J=1,2
C   SCALE X,Y, Z TO INCHES IN BOX, W,R, TO CAMERA LOCATION
000020   P(1)=XMIN+(XY(1,J)-1,0)*GX(1)-CAMWKG(1)
000027   P(2)=YMIN+(XY(2,J)-1,0)*GX(2)-CAMWKG(2)
000036   P(3)=XY(3,J)*GX(3)+ZORG-CAMWKG(3)
C   ROTATE TO CAMERA SYSTEM + PROJECT (DO NOT SHIFT BY PLOTORG)
000043   CALL ROTATE(P,AMTX,XYC(1,J))
C   PROJECTED END POINTS
000050   XP(1,J)=(XYC(1,J)/XYC(3,J)-XORG(1))*CAMWKG(4)+CENTER(1)
000063   2 XP(2,J)=(XYC(2,J)/XYC(3,J)-XORG(2))*CAMWKG(4)+CENTER(2)
C   FIND SEGMENT WITHIN FRAME LIMITS
000074   CALL CHOP4(PLOTX,PLOTY,XP,XCHOP,INSIDE,NOUT)
C   OUT IF NOTHING WITHIN FRAME
000100   IF(INSIDE,LE,0)RETURN
C   IF JUST 1 POINT CALLED FOR, DRAW SEGMENT WITHOUT EDITING FOR V
000105   IF(KP.EQ,1)GO TO 10
C   DELTAS IN CAMERA SYSTEM IN INCHES
000107   DC(1)=XYC(1,2)-XYC(1,1)
000111   DC(2)=XYC(2,2)-XYC(2,1)
000113   DC(3)=XYC(3,2)-XYC(3,1)
000115   DO3J=1,2
C   UNSCALE ALLOWED PART IN FILM PLANE, GIVING X/Z, Y/Z
000116   DO4K=1,2

```

```

000117      4      XP(K,J)=(XCHOP(K,J)-CENTER(K))/CAMWKG(4)+XORG(K)
C          FRACTION OF LINE LENGTH (IN SPACE IN CAMERA SYSTEM)
000141      C      OFL(J)=(ABS(XYC(1,1)-XP(1,J)*XYC(3,1))+ABS(XYC(2,1)-XP(2,J)*XYC(
1))) / (ABS(XP(1,J)*DC(3)-DC(1))+ABS(XP(2,J)*DC(3)-DC(2)))
C          END POINT IN ORIGINAL UNITS
C          (X,Y IN GRID COORDINATES, Z IN MICROPODS)
000167      DO5K=1,3
000171      5      XYW(K,J)=XY(K,1)+DELTA(K)*FL(J)
000206      3      CONTINUE
C          NUMBER OF POINTS IN REMAINING SEGMENT
000210      NP=IFIX(FLOAT(KP)*ABS(FL(2)-FL(1)))+1
000216      NP=MIN0(NP,KP)
000221      FNP=FLOAT(NP)
000222      DO6J=1,3
000224      DELTA(J)=XYW(J,2)-XYW(J,1)
C          SET INITIAL POINT
000227      6      PC(J)=XYW(J,1)
000232      PHIP=IVIS(PC(1),PC(2),PC(3),Z,KDIM)
C          BEAM OFF BEFORE FIRST POINT
000243      IBEAM=0
000244      DO1J=1,NP
000246      F=FLOAT(J)/FNP
000250      QC(1)=XYW(1,1)+F*DELTA(1)
000253      QC(2)=XYW(2,1)+F*DELTA(2)
000256      QC(3)=XYW(3,1)+F*DELTA(3)
000261      PHIQ=IVIS(QC(1),QC(2),QC(3),Z,KDIM)
000273      CALL DRAWPG(Z)
000275      PC(1)=QC(1)
000301      PC(2)=QC(2)
000303      PC(3)=QC(3)
000304      1      PHIP=PHIQ
000310      RETURN
C
C          PLOT SEGMENT IMMEDIATELY WITHOUT EDITING
000311      10     IB=0
000312      DO11J=1,2
000314      CALL CALCMP(XCHOP(1,J)+PLOTORG(1),XCHOP(2,J)+PLOTORG(2),IB,1)
000325      11     IB=KBEAM+2
000334      RETURN
000334      END

```

*FOLLOWING VARIABLES EQUIVALENCED BUT NOT REFERENCED

V
W
U
MX
FMX

RUN VERSION 2,3 - CDC LEVEL 26 - NYU LEVEL 11

GNOMON

SUBPROGRAM LENGTH
000446

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

4 - 000117 5 - 000171 10 - 000311

BLOCK NAMES AND LENGTHS

COM3DP - 000103 COM3DPA - 000063

VARIABLE ASSIGNMENTS

AMTX	-	000045C01	CAMWKG	-	000025C01	CAMXYZ	-	000014C01	CENTER	-	000100C1
DC	-	000410	DELTA	-	000373	ENDA	-	000014C02	ENDB	-	000022C1
F	-	000444	FL	-	000435	FLIM	-	000021C01	FN	-	000443
FX	-	000073C01	GX	-	000070C01	IB	-	000445	IBEAM	-	000061C1
INSIDE	-	000440	J	-	000437	K	-	000441	KBEAM	-	000023C1
KCAMWKG	-	000025C01	KDIM	-	000062C02	LIMIT	-	000017C01	NOUT	-	000077C1
NP	-	000442	OLDQ	-	000030C02	P	-	000006C02	PC	-	000000C1
PHIA	-	000060C02	PHIP	-	000056C02	RHIQ	-	000057C02	PK	-	000050C1
PLOTORG	-	000012C01	PLOTX	-	000010C01	PLOTY	-	000011C01	PW	-	000034C1
Q	-	000011C02	QC	-	000003C02	QK	-	000053C02	QW	-	000037C1
T	-	000042C02	XCHOP	-	000413	XMIN	-	000000C01	XORG	-	000065C1
XP	-	000376	XYC	-	000402	XYW	-	000427	YMIN	-	000002C1
ZLIM	-	000006C01	ZORG	-	000076C01						

START OF CONSTANTS
000336

START OF TEMPORARIES
000342

START OF INDIRECTS
000363

UNUSED COMPILER SPACE
021600

```

FUNCTION IVIS(XI,ETA,ZIN,Z,IDIM)
      CORRECTED VERSION, 24FEB69
      DETERMINE IF POINT XI, ETA IS VISIBLE,
      POINT IS GIVEN BY XI, ETA, ZIN
      AND VISIBILITY IS TESTED WITH RESPECT TO SURFACE Z(X,Y)
      XI, ETA COORDINATES EXPRESSED AS INDICES OF ARRAY Z, BUT NEED
      BE INTEGERS IN GENERAL, FOR ENTRY IVIS, THEY MUST BE,

      CDE PACKAGE
000010      DIMENSION Z(IDIM,10)

000010      COMMON/COM3DP/XMIN,XMAX,YMIN,YMAX,ZMIN,ZMAX,ZLIM(2),PLOTX,PLOT
1PLOTORG(2),CAMXYZ(3),MX,NY,FMX,FNY,KBEAM,ZETA,CAMWKG(32),XORG(
2GX(3),FX(2),KSCALE,ZORG,NOUT,CENTER(2),PQLIMIT
000010      DIMENSION LIMIT(2),FLIM(2),AMTX(3,3),KCAMWKG(32)
000010      EQUIVALENCE(U,CAMXYZ),(V,CAMXYZ(2)),(W,CAMXYZ(3)),(MX,LIMIT),
1(FMX,FLIM),(AMTX,CAMWKG(17)),(CAMWKG,KCAMWKG)
      END CDE

000010      ZETA=ZIN

      IF ZETA OUT OF BOX, TEST IF POINT APPEARS IN PICTURE AREA
000011      IF((ZETA=ZLIM(1))*(ZETA=ZLIM(2)),LE,0,0)GO TO 5

      TEST IF OUT-OF-BOX POINT APPEARS IN PICTURE

      CONVERT INDICES TO INCHES IN X,Y
000016      R=XMIN+(XI-1,0)*GX(1)-CAMWKG(1)
000024      S=YMIN+(ETA-1,0)*GX(2)-CAMWKG(2)
000031      T=ZETA*GX(3)+ZORG-CAMWKG(3)
000035      ZW=AMTX(3,1)*R+AMTX(3,2)*S+AMTX(3,3)*T
000043      OW=((AMTX(1,1)*R+AMTX(1,2)*S+AMTX(1,3)*T)/ZW-XORG(1))*
1 CAMWKG(4)+CENTER(1)
      XW HAS TO BE WITHIN PICTURE SIZE
000055      IF(XW*(PLOTX-XW),LT,0,0)GO TO 40
000060      OW=((AMTX(2,1)*R+AMTX(2,2)*S+AMTX(2,3)*T)/ZW-XORG(2))
1 *CAMWKG(4)+CENTER(2)
000072      IF(YW*(PLOTY-YW),LT,0,0)GO TO 40

      INITIAL P FUNCTION
000075      PO=0,0
000076      R=U-XI
000100      S=V-ETA
000102      T=W-ZETA

      TEST IF WE CHECK ALONG X
000104      IF(ABS(R),LT,1,0)GO TO 20
      CONSTANTS FOR Y(X),Z(X)
000110      CY=S/R
000111      CZ=T/R
000113      DXI=SIGN(1,0,R)
      INITIAL POINT, TAKE AINT(XI) IF ,NE, XI AND STEPS IN RIGHT DIR
000115      XIW=AIN(XI)
000117      IF((XIW-XI)*DXI,LE,0,0)XIW=XIW+DXI
      SKIP IF OFF LIMITS (WE ARE ON EDGE OF PLOT REGION)
000124      IF((XIW-1,0)*(XIW-FMX),GT,0,0)GO TO 20
      FINAL POINT, TAKE AINT(U) IF IT MOVES OPPOSITE DXI, ELSE ROUND

```

```

000132      XIEND=AIN(T(U)
000134      IF((XIEND-U)*DXI,GE,0,0)XIEND=XIEND-DXI
C      BUT DO NOT GO BEYOND EDGES
000140      XIEND=AMAX1(1,0,AMIN1(XIEND,FMX))
C
C      AFTER TESTING, RE-ORDER THESE STATEMENTS
000147      J=IFIX(XIW)
000150      KDXI=IFIX(LXI)
000152      KXIEND=IFIX(XIEND)
000153      XW=XIW-U
C
C      IF LIMITS CROSS, NO TEST
000155      IF((XIEND-XIW)*DXI,LT,0,0)GO TO 20
C      GET Y(X)
000160      YW=V+XW*CY
C      IF Y IS OFF LIMITS, DONE
000163      IF((YW-1,0)*(YW-FNY))21,25,20
C      ON EDGE EXACTLY, NO INTERPOLATION
000171      K=IFIX(YW)
000173      IF(W+XW*CZ=Z(J,K))4,10,7
C      INDEX FOR LOWER Y OF INTERVAL
000203      K=IFIX(YW)
000205      DY=YW-FLOAT(K)
C      TEST Z OF LINE = Z OF SURFACE, ACCEPT ZERO DIFFERENCE,
000207      IF((W+XW*CZ)=(Z(J,K)+DY*(Z(J,K+1)-Z(J,K)))) 4, 10, 7
C      NEGATIVE, OK IF P0 NEG, OR ZERO, ELSE REJECT
000224      IF(P0)10,6,40
C      P0 WAS ZERO, SET NEG,
000226      P0 =-1,0
000230      GO TO 10
C      PLUS, OK IF P0 + OR ZERO, ELSE, REJECT,
000230      IF(P0) 40, 8, 10
C      SET PLUS
000232      P0 = 1,0
C      TEST IF DONE, ADVANCE IF NOT
000234      IF(J,EQ,KXIEND) GO TO 20
000236      J=J+KDXI
000237      XW=XW+DXI
000241      GO TO 3
C
C      CHECK IF WE TEST IN Y DIRECTION
000242      IF(ABS(S).LT,1,0)GO TO 45
C      CONSTANTS FOR X(Y),Z(Y)
000246      CX=R/S
000247      CZ=T/S
000251      DETA=SIGN(1,0,S)
000253      ETAW=AIN(T(ETA)
000255      IF((ETAW-ETA)*DETA,LE,0,0)ETAW=ETAW+DETA
C      CHECK WHETHER ON LIMITS
000262      IF((ETAW-1,0)*(ETAW-FNY),GT,0,0)GO TO 45
000270      ETAEND=AIN(T(V)
000272      IF((ETAEND-V)*DETA,GE,0,0)ETAEND=ETAEND-DETA
000276      ETAEND=AMAX1(1,0,AMIN1(FNY,ETAEND))
000305      K=IFIX(ETAW)
000306      KDETA=IFIX(DETA)
000310      YW=ETAW-V
000312      KETAEND=IFIX(ETAEND)
C      IF LIMITS CROSS, NO TEST, BUT TEST SINGLE POINT IF WE HAVE ALRI

```

```

000313 C TESTED X
000314 A=ETAEND=ETA W
000316 IF(A*DETA,LT,0,0)GO TO 45
IF(A,EQ,0,0,AND,P0,EQ,0,0)GO TO 45
C GET X(Y)
000324 23 XW=U+YW*CX
C IF X OFF LIMITS, DONE
000327 IF((XW-1,0)*(XW-FMX))44,46,45
000335 46 J=IFIX(XW)
000337 IF(W+YW*CZ=Z(J,K))24,30,27
000347 44 J=IFIX(XW)
000351 DX=XW-FLOAT(J)
000353 IF((W+YW*CZ)=(Z(J,K)+DX*(Z(J+1,K)-Z(J,K))))24,30,27
C NEG,, P0 MUST BE NEG OR ZERO ELSE REJECT
000367 24 IF(P0)30,26,40
C SET P0 NEG
000371 26 P0=-1,0
000373 GO TO 30
C POS, P0 MUST BE ZERO OR + ELSE REJECT
000373 27 IF(P0)40,28,30
000375 28 P0=1,0
C TEST IF DONE, ADVANCE IF NOT,
000377 30 IF(K,EQ,KETAEND)GO TO 45
000401 K=K+KDETA
000402 YW=YW+DETA
000404 GO TO 23
C
C REJECT THIS POINT, RETURN ZERO,
000405 40 IVIS=0
000406 RETURN
C
C ACCEPT, RETURN +/- 1 ACCORDING TO SIGN OF P
000407 45 IVIS=IFIX(P0)
C IF P0 ZERO, CAMERA WAS RIGHT OVER XI, ETA,
IF(P0,EQ,0,0)IVIS=IFIX(SIGN(1,0,T))
000411 RETURN
000416
000420 END

```

*FOLLOWING VARIABLES EQUIVALENCED BUT NOT REFERENCED
MX

RUN VERSION 2,3 - CDC LEVEL 26 - NYU LEVEL 11

IVIS

SUBPROGRAM LENGTH
000520

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

3	-	000160	4	-	000224	5	-	000075	6	-	000226
7	-	000230	8	-	000232	10	-	000234	20	-	000242
21	-	000203	23	-	000324	24	-	000367	25	-	000171
26	-	000371	27	-	000373	28	-	000375	30	-	000377
40	-	000405	44	-	000347	45	-	000407	46	-	000335

BLOCK NAMES AND LENGTHS
COM2DP - 000103

VARIABLE ASSIGNMENTS

A	-	000516	AMTX	-	000045C01	CAMWKG	-	000025C01	CAMXYZ	-	000014C01
CENTER	-	000100C01	CX	-	000510	CY	-	000476	CZ	-	000477
DETA	-	000511	DX	-	000517	DXI	-	000500	DY	-	000507
ETAEND	-	000513	ETAW	-	000512	ELIM	-	000021C01	FMX	-	000021C01
FNY	-	000022C01	FX	-	000073C01	GX	-	000070C01	IVIS	-	000466
J	-	000503	K	-	000506	KCAMWKG	-	000025C01	KDETA	-	000514
KDXI	-	000504	KETAEND	-	000515	KXIEND	-	000505	LIMIT	-	000017C01
PLOTORG	-	000012C01	PLOTX	-	000010C01	PLOTY	-	000011C01	PO	-	000475
R	-	000467	S	-	000470	T	-	000471	U	-	000014C01
V	-	000015C01	W	-	000016C01	XIEND	-	000502	XIW	-	000501
XMIN	-	000000C01	XORG	-	000065C01	XW	-	000473	YMIN	-	000002C01
YW	-	000474	ZETA	-	000024C01	ZLIM	-	000006C01	ZORG	-	000076C01
ZW	-	000472									

START OF CONSTANTS
000422

START OF TEMPORARIES
000426

START OF INDIRECTS
000464

UNUSED COMPILER SPACE
021500

```

000010 FUNCTION MIRBANE(XA,XB,XC,XD,XCUT)
C DIMENSION XA(2),XB(2),XC(2),XD(2),XCUT(2)
C VERSION OF 13JAN69, WITH ALL 4 POINTS SEPARATE ARGUMENTS
C
C DETERMINE IF TWO FINITE LINE SEGMENTS XA-XB AND XC-XD INTERSECT
C AT POINT WITHIN BOTH SEGMENTS,
C
C RETURN +1 IF SO, AND IF SO CALCULATE INTERSECTION POINT IN XCUT
C
C RETURN ZERO IF NOT
C RETURN -1 IF LINES INTERSECT BUT ARE ALSO CONCURRENT
C
C XCUT(1)=X OF INTERSECTION, XCUT(2)=Y
C
C METHOD * POINTS XC, XD MUST APPEAR TO LIE ON OPPOSITE SIDES OF
C LINE XA-XB WHEN VIEWED FROM EITHER XA OR XB, ALSO, POINTS XA AND
C XB MUST SEEM TO LIE ON EITHER SIDE OF XC-XD WHEN VIEWED FROM EITHER
C XC OR XD, IT IS NECESSARY TO CHECK THE VIEW FROM JUST ONE END,
C POINT OF EACH SEGMENT LOOKING AT THE ENDS OF THE OTHER SEGMENT
C IF ONE CONDITION IS TRUE, INTERSECTION POINT LIES WITHIN ONE OF
C SEGMENTS, IF NEITHER TRUE, INTERSECTION LIES WITHIN NEITHER,
C
C * * * NOTE ** IF THE END POINT OF ONE SEGMENT LIES ON THE OTHER
C IS CONSIDERED INTERSECTION, WHEN THIS HAPPENS ONE OF THE CROSS
C IN THE FIRST TWO IF STATEMENTS WILL BE ZERO, IT MAY BE DESIRABLE
C TO CHECK FOR THIS POSSIBILITY IN SOME USES, FOR NOW, IT IS OK.
C
000010 MIRBANE=0
000010 A1=XB(2)-XA(2)
000013 B1=XB(1)-XA(1)
C TEST XC, XD SEEN FROM XA
000014 0IF(((XC(1)-XA(1))*A1-(XC(2)-XA(2))*B1)*(B1*(XD(2)-XA(2))-
1A1*(XD(1)-XA(1))),LT,0,0)RETURN
C
C TEST XA, XB SEEN FROM XC,
000036 A2=XD(2)-XC(2)
000040 B2=XD(1)-XC(1)
000042 0IF(((XA(1)-XC(1))*A2-(XA(2)-XC(2))*B2)*(B2*(XB(2)-XC(2))-
1A2*(XB(1)-XC(1))),LT,0,0)RETURN
000064 MIRBANE=1
C INTERSECTION POINT, LINES IN FORM AX+BY+C=0
000065 C1=XA(1)*XB(2)-XB(1)*XA(2)
000071 C2=XC(1)*XD(2)-XD(1)*XC(2)
000074 DET=A2*B1-A1*B2
000100 IF(DET,EQ,0,0)GO TO 1
000101 XCUT(1)=(B1*C2-B2*C1)/DET
000105 XCUT(2)=(A1*C2-A2*C1)/DET
000111 RETURN
000112 1 MIRBANE=-1
000113 RETURN
000114 END

```

RUN VERSION 2,3 - CDC LEVEL 26 - NYU LEVEL 11

MIRBANE

SUBPROGRAM LENGTH

000157

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

1 - 000112

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS

A1	-	000150	A2	-	000152	B1	-	000151	B2	-	000153
C1	-	000154	C2	-	000155	DET	-	000156	MIRBANE	-	000147

START OF CONSTANTS

000116

START OF TEMPORARIES

000117

START OF INDIRECTS

000135

UNUSED COMPILER SPACE

023100

SUBROUTINE ROTATE (XIN,A,XOUT)
ROTATE VECTOR XIN BY MATRIX A TO GET XOUT

C
C
C
C
C
C
C

000006

DIMENSION XIN(3),A(9),XOUT(3)

000006

XW=XIN(1)

000006

YW=XIN(2)

000010

XOUT(1)=A(1)*XW+A(4)*YW+A(7)*XIN(3)

000017

XOUT(2)=A(2)*XW+A(5)*YW+A(8)*XIN(3)

000026

XOUT(3)=A(3)*XW+A(6)*YW+A(9)*XIN(3)

000035

RETURN

000035

END

RUN VERSION 2,3 - CDC LEVEL 26 - NYU LEVEL 11

ROTATE

SUBPROGRAM LENGTH
000065

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
XW - 000063 YW - 000064

START OF CONSTANTS
000037

START OF TEMPORARIES
000040

START OF INDIRECTS
000042

UNUSED COMPILER SPACE
023300

```

SUBROUTINE CALCMP(X,Y,IZ,K)
COMMON/CRTPLT/ICRT
000007 IF(ICRT,EQ,0) GO TO 30
000008 IX=100,0*X+12,0
000010 IY=100,0*Y+12,0
000013 IF(IZ)10,20,10
000016 IF(IZ)10,20,10
000017 10 CALL ABSVEC(IX,IY)
000021 IF(ICRT,LT,0) GO TO 30
000025 RETURN
000026 20 CALL ABSPOS(IX,IY)
000030 IF(ICRT,LT,0) GO TO 30
000034 RETURN
000035 30 J=2
000036 IF(IZ,EQ,0) J=3
000040 CALL PLOT(X,Y,J)
000042 RETURN
000043 END

```

CALCMP

FUNCTION ASSIGNMENTS

10 - 000017 20 - 000026 30 - 000035

VARIABLE ASSIGNMENTS

ICRT - 000000C01 IX - 000052 IY - 000053 J - 000054

START OF TEMPORARIES
000050

START OF INDIRECTS
000052

UNUSED COMPILER SPACE
023500

AUF1135 //// END OF LIST //// 00001812 LINES

[illegible]

RUN VERSION 2,3 - CDC LEVEL 26 - NYU LEVEL 11

DRAWPG

SUBPROGRAM LENGTH
000452

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

2	-	000014	3	-	000020	4	-	000022	5	-	000024
6	-	000027	7	-	000015	10	-	000034	11	-	000036
12	-	000050	13	-	000063	14	-	000064	15	-	000060
30	-	000071	31	-	000075	50	-	000412	200	-	000101
201	-	000114	205	-	000127	210	-	000174	216	-	000240
220	-	000267	300	-	000302	310	-	000345			

BLOCK NAMES AND LENGTHS

COM3DP - 000103 COM3DPA - 000063

VARIABLE ASSIGNMENTS

AMTX	-	000045C01	CAMWKG	-	000025C01	CAMXYZ	-	000014C01	CENTER	-	000100C1
ENDA	-	000014C02	ENDB	-	000022C02	FLIM	-	000021C01	FX	-	000073C1
GX	-	000070C01	IBEAM	-	000061C02	JGOTO	-	000446	KBEAM	-	000023C1
KCAMWKG	-	000025C01	KDIM	-	000062C02	KFLAG	-	000447	KGOTO	-	000445
LIMIT	-	000017C01	MFLAG	-	000450	OLDQ	-	000030C02	P	-	000006C1
PC	-	000000C02	PHIA	-	000060C02	PHIP	-	000056C02	PHIQ	-	000057C1
PK	-	000050C02	PLOTORG	-	000012C01	RGLIMIT	-	000102C01	PW	-	000034C1
Q	-	000011C02	QC	-	000003C02	QK	-	000053C02	QW	-	000037C1
SL	-	000451	T	-	000042C02	XMIN	-	000000C01	XORG	-	000065C1
YMIN	-	000002C01	ZLIM	-	000006C01	ZORG	-	000076C01			

START OF CONSTANTS
000415

START OF TEMPORARIES
000422

START OF INDIRECTS
000442

UNUSED COMPILER SPACE
021500