```
0051          AL4= AL2*AL2
0052          AG2= AG*AG
0053          AG3= AG*AG2
0054          L2= L*L
0055          L3= L2*L
0056          L4=L3*L
0057      C
0058          C1= -QK-QK/3.*AL2*(AL+3.)/(AL+1.)-QEI*L4/72.*((AL+3.)/(AL+1.))**2
0059          c2= 0.0
0060          C3= QK*AG*AG+QEI*L2/12.*(AL+3.)/(AL+1.)
0061          C4= -2.*QK/3.*AG3/(AL+1.)-QEI*L/I8.*(2.*AL2+6.*AL+3.)/(AL+1.)**2
0062          C5= QEI/24.
0063          C6= -QK
0064          C7= QK*(AL-1.)/(AL+1.)+QEI*L3/6./AG/(AL+1.)**2
0065          PO= 2.*Q/AG/(AL+1.)+2.*Q*L/3.*(2.*AL2+6.*AL+3.)/(AL+1.)**2
0066          PO= ABS(PO)
0067          WO= ABS(C1)
0068          RETURN
0069       3000 CONTINUE
0070          LM2  N/EI
0071          LM= SQRT(LM2)
0072          LM3= LM*LM2
0073          CLL= COSH(LM*L)
0074          SLL= SINH(LM*L)
0075      C
0076          LC=SQRT(K/4./EI)
0077          DC= N/4./EI
0078          AG=SQRT(LC+DC)
0079          BG=SQRT(LC-DC)
0080      C
0081          AG2=AG*AG
0082          BG2=BG*BG
0083          AO= SLL*(BG2-AG2)-CLL*LM*AG
0084          BO= 2.*AG*BG*SLL+BG*LM*CLL
0085          EO= -AG*(LM2+3.*BG2-AG2)
0086          FO= BG*(BG2-3.*AG2+LM2)
0087          L2= L*L
0088      C
0089          C5=  -QN/2.
0090          C6= -QK
0091          C7= (2.*                C5-(LM*AO+(1.-CLL)*EO)*C6)
0092         &      /(BO*LM+FO*(1.-CLL))
0093          C4= (2.*L*LM2*C5-EO*C6-FO*C7)/LM3
0094          C3  (BG*C7-AG*C6+(1  CLL)*LM*C4 2 *l*~5)/(LM*SLL)
0095          C2= -LM*C4
0096          C1= -QK-L2*C5+(LM*L-SLL)*C4-CLL*C3
0097          PO= 2.*LM3*EI*C4
0098          PO= ABS(PO)
0099          WO= ABS(C1 + C3)
0100          RETURN
0101          END
```

```
      SUBROUTINE STRESS(SMAX,STL)
      IMPLICIT REAL*8 (A-Z)
      INTEGER*4 I,IMAX
C
      PARAMETER (PI=3.141592654)
      COMMON /PARA/N,EI,L,K,QN,QK,LM,LM2,EM,OD,QEI,Q,SIGMA
      COMMON /CONST/W0,P0,C1,C2,C3,C4,C5,C6,C7
      COMMON /GREEK/AG,AG2,BG,BG2
C
      SLEN=0.0
      IMAX= NINT(L/6.0) + 1
      DX= L/FLOAT(IMAX)
      DX2= DX*DX
      M1M=0.0
      M2M=0.0
      IF (SIGMA) 1000,2000,3000
C
 1000 DO 1010 I=0,IMAX
      X= DX*FLOAT(I)
      X2= X*X
      LX= LM*X
      CLX= COS(LX)
      SLX= SIN(LX)
C
      W1= C1 + C2*X + C3*CLX + C4*SLX + C5*X2
      DW=W1-WOLD
      WOLD=W1
      IF(X.NE.0.0)THEN
        DS=SQRT(DW*DW+DX2)
        SLEN=SLEN+DS
      ENDIF
C
      M1= -LM2*(C3*CLX + C4*SLX) + 2.*C5
 1010 M1M= MAX(M1M,ABS(M1))
C
      IMAX= 10
      DX= 12.0
      DO 1020 I=0,IMAX
      X= DX*FLOAT(I)
      AX= AG*X
      BX=BG*X
      EBX= EXP(-BX)
      CAX= COS(AX)
      SAX= SIN(AX)
      M2= EBX*((C6*(BG2-AG2)-C7*(2.*BG*AG))*CAX
     &   +  (C6*(2.*BG*AG)+C7*(BG2-AG2))*SAX)
C
 1020 M2M= MAX(M2M,ABS(M2))
C
      GOTO 5000
```

```
0051     2000 DO 2010 I=0,IMAX
0052          X= DX*FLOAT(I)
0053          X2= X*X
0054          X3= X*X2
0055          X4= X2*X2
0056     C
0057          W1= C1 + C2*X + C3*X2 + C4*X3 +C5*X4
0058          DW=W1-WOLD
0059          WOLD=W1
0060          IF(X.NE.0.0)THEN
0061            DS=SQRT(DW*DW+DX2)
0062            SLEN=SLEN+DS
0063          ENDIF
0064     C
0065          M1= 2.*C3 +6.*C4*X + 12.*C5*X2
0066      2010 M1M= MAX(M1M,ABS(M1))
0067     C
0068          IMAX= 10
0069          DX= 12.0
0070          DO 2020 I=0,IMAX
0071          X= DX*FLOAT(I)
0072          AX= AG*X
0073          EAX= EXP(-AX)
0074          CAX= COS(AX)
0075          SAX= SIN(AX)
0076          M2= 2.*AG2*EAX*(-C7*CAX + C6*SAX)
0077     C
0078      2020 M2M= MAX(M2M,ABS(M2))
0079     C
0080          GOTO 5000
0081     C
0082      3000 DO 3010 I=0,IMAX
0083          X= DX*FLOAT(I)
0084          X2= X*X
0085          LX= LM*X
0086          CLX= COSH(LX)
0087          SLX= SINH(LX)
0088     C
0089          W1= C1 + C2*X + C3*CLX + C4*SLX + C5*X2
0090          DW=W1-WOLD
0091          WOLD=W1
0092          IF(X.NE.0.0)THEN
0093            DS=SQRT(DW*DW+DX2)
0094            SLEN=SLEN+DS
0095          ENDIF
0096     C
0097          M1= LM2*(C3*CLX + C4*SLX) + 2.*C5
0098      3010 M1M= MAX(M1M,ABS(M1))
0099     C
0100          IMAX  10
```

```
                DX= 12.0
                DO 3020 I=0,IMAX
                X= DX*FLOAT(I)
                AX= AG*X
                BX=BG*X
                EAX= EXP(-AX)
                CBX= COS(BX)
                SBX= SIN(BX)
                M2= EAX*((C6*(AG2-BG2)-C7*(2.*AG*BG))*CBX
               &   +  (C6*(2.*AG*BG)+C7*(AG2-BG2))*SBX)
            C
             3020 M2M= MAX(M2M,ABS(M2))
            C
                GOTO 5000
             5000 STL=EM*(SLEN/L-1.0)
                MMAX=MAX(M1M,M2M)
                SMAX=ABS(SIGMA)+EM*MMAX*OD/2.
                RETURN
            C
                END
```

PROGRAM PDROP

The computer program POROP calculates the profile and stresses for free deflection of a pipeline. The program is based on the equations for free deflection described in Appendix B. The input for the program consists of:

1. Outside diameter of the pipe, OD (inches)
2. Wall thickness of the pipe, t (inches)
3. Elastic modulus of the pipe material, E (psi)
4. Specific weight *of* the pipe material, **y** (lb/in^3)
5. Soil stiffness parameter, **k** (psi)
6. Specific gravity of the fluid in the pipe, SG (dimensionless)
7. Maximum lowering depth, H (inches)
8. Existing axial stress in the pipeline, σ_L (psi).

The PDROP program calculates the length of the unsupported portion of the pipeline, for the given lowering depth. The axial stress induced by the lengthening of the pipeline is taken into account in the program and is added to the existing stress to determine the total axial stress. The program iterates on the solution until the lengthening stress from two successive iterations is equal.

The program was designed to be run interactively. The output from the program is displayed on the terminal screen. An additional copy of the output may be written to the file FORØØ1.DAT, if desired. The output consists of:

1. An echo *of* the input parameters
2. The length of unsupported pipeline and maximum stress.

A computer listing of the program (VAX-11/780 version) is given below.