Lampiran I. Listing program PGV

```
REM PROGRAM UNTUK MENGHITUNG RESPON LINIER dan RESPON NON LINIER
COMMON FILE$
CLS
INPUT "NAMA FILE INPUT DATA = ", FILE$
OPEN "I", #1, FILE$
'OPEN "I", #1, "centro-1.dat"
OPEN "O", #2, "RESPON.DAT"
PRINT "JUMLAH LAPISAN TANAH                    NE = "; : INPUT #1, NE
PRINT "MASSA BANGUNAN(KG)                      MB = "; : INPUT #1, MB
FOR I = 1 TO NE
PRINT "TEBAL LAPISAN TANAH  (cm)  TL("; I; ") = "; : INPUT #1, TL(I)
NEXT I
FOR I = 1 TO NE
PRINT "INDEX PLASTISITAS  PI("; I; ") = "; : INPUT #1, PI(I)
NEXT I
FOR I = 1 TO NE
PRINT "SUDUT GESER DALAM  FI("; I; ") = "; : INPUT #1, FI(I)
NEXT I
FOR I = 1 TO NE
PRINT "ANGKA PORI TANAH  E("; I; ")= "; : INPUT #1, E(I)
NEXT I
FOR I = 1 TO NE
PRINT "BERAT JENIS TANAH  GS("; I; ")= "; : INPUT #1, GS(I)
NEXT I
FOR I = 1 TO NE
PRINT "BERAT VOLUME TANAH  Gw("; I; ") = "; : INPUT #1, Gw(I)
NEXT I
PRINT "KONDISI LAPISAN : PILIH SALAH SATU NOMOR DIBAWAH"
PRINT "1. UNTUK PASIR TIDAK TERENDAM AIR."
PRINT "2. UNTUK LEMPUNG TIDAK TERENDAM AIR."
PRINT "3. UNTUK PASIR TERENDAM AIR."
PRINT "4. UNTUK LEMPUNG TERENDAM AIR."
FOR I = 1 TO NE
PRINT "TANAH TERENDAM/TIDAK TERENDAM KT("; I; ") = "; : INPUT #1, KT(I)
NEXT I
PRINT "PERCEPATAN GRAVITASI ( cm/dt2 )  G = ": INPUT #1, GRAV
PRINT "SELISIH WAKTU GEMPA dt = ": INPUT #1, DT
PRINT "JARAK EPICENTRUM = ": INPUT #1, DEL
PRINT "MAGNITUDE = ": INPUT #1, MAG
FOR I = 1 TO NE
PRINT "FAKTOR REDUKSI MODULUS GESER LINIER ELASTIS = ": INPUT #1, F(I)
NEXT I
REGA = (.894 * (10 ^ (.548 * MAG)) * ((DEL + 30) ^ -.774) * (10 ^ -6)) *
100
CLS
PRINT #2, "A. INPUT DATA LAPISAN TANAH DAN KONDISINYA "
PRINT #2,
PRINT #2, "Lps  Tebal  PI(%)     FI        e        Gs        Gw      ";
PRINT #2, "Geff        Ko        k  ";
PRINT #2,
FOR I = 1 TO NE
IF KT(I) = 1 THEN GE(I) = (GS(I) * Gw(I)) / (1 + E(I))
IF KT(I) = 2 THEN GE(I) = (GS(I) * Gw(I)) / (1 + E(I))
IF KT(I) = 3 THEN GE(I) = ((GS(I) - 1) * Gw(I)) / (1 + E(I))
```

```
IF KT(I) = 4 THEN GE(I) = ((GS(I) - 1) * Gw(I)) / (1 + E(I))
CONST PHI = 3.141592654#
SELECT CASE PI(I)
CASE IS = 0
   Ko(I) = 1 - SIN(FI(I) * PHI / 180)
CASE 0 TO 40
   Ko(I) = .4 + .007 * PI(I)
CASE IS > 40
   Ko(I) = .68 + .001 * PI(I)
END SELECT
SELECT CASE PI(I)
CASE 0 TO 20
   KK(I) = (.18 - 0) * (PI(I) - 0) / (20 - 0) + 0
CASE 20 TO 40
   KK(I) = (.3 - .18) * (PI(I) - 20) / (40 - 20) + .18
CASE 40 TO 60
   KK(I) = (.41 - .3) * (PI(I) - 40) / (60 - 40) + .3
CASE 60 TO 80
   KK(I) = (.48 - .41) * (PI(I) - 60) / (80 - 60) + .41
CASE 80 TO 100
   KK(I) = (.5 - .48) * (PI(I) - 80) / (100 - 80) + .48
CASE IS >= 100
   KK(I) = .5
END SELECT
LP(I) = I
PRINT #2, USING "# ####.##  ##.##  ##.##"; LP(I); TL(I); PI(I); FI(I);
PRINT #2, USING "###.#### "; E(I); GS(I); Gw(I); GE(I); Ko(I); KK(I)
NEXT I
PRINT #2,
PRINT #2, "B. HASIL PERHITUNGAN G MAKSIMUN, MASSA DAN KEKAKUAN"
PRINT #2,
PRINT #2, "Lps  Elevasi   SIGMA1   SIGMA2  SIGMA0    G MAKS  ";
PRINT #2, "       G     MASSA    KEKAKUAN"
PRINT #2, "                         =SIGMA3"
FOR I = 1 TO NE
KED(1) = TL(1)
KE(1) = 0
FOR B = 2 TO NE
IF I = B THEN KED(B) = KED(B - 1) + TL(I)
IF I = B THEN KE(I) = KED(B - 1)
NEXT B
S(I) = GE(I) * TL(I) / 1000
SA(1) = 0
SB(1) = S(1)
FOR J = 2 TO NE
IF I = J THEN SA(J) = SB(J - 1)
IF I = J THEN SB(J) = SB(J - 1) + S(I)
NEXT J
MBA(I + 1) = MB / ((6000 + KED(I)) * (1800 + KED(I)))
MBA(1) = 0
MBB(I + 1) = MB / ((6000 + KED(I)) * (1800 + KED(I)))
MBB(I) = MBA(I + 1)
SIGA(I) = SA(I) + MBA(I)
SIGB(I) = SB(I) + MBB(I)
SIGA2(I) = Ko(I) * SIGA(I)
SIGB2(I) = Ko(I) * SIGB(I)
SIGA0(I) = (SIGA(I) + SIGA2(I) + SIGA2(I)) / 3
```

```
      SIGB0(I) = (SIGB(I) + SIGB2(I) + SIGB2(I)) / 3
      OCR = 1
      IF KT(I) = 1 OR KT(I) = 3 THEN
      GAM(I) = 700 * (2.17 - E(I)) ^ 2 * SIGA0(I) ^ .5 / (1 + E(I))
      GBM(I) = 700 * (2.17 - E(I)) ^ 2 * SIGB0(I) ^ .5 / (1 + E(I))
      ELSE
      GAM(I) = 331 * (OCR ^ KK(I)) * (2.973 - E(I)) ^ 2 * SIGA0(I) ^ .5 / (1 +
E(I))
      GBM(I) = 331 * (OCR ^ KK(I)) * (2.973 - E(I)) ^ 2 * SIGB0(I) ^ .5 / (1 +
E(I))
      END IF

      SELECT CASE PI(I)
         CASE 0 TO 10
             REGH(I) = .04
         CASE 10 TO 20
             REGH(I) = .07
         CASE 20 TO 40
             REGH(I) = .12
         CASE 40 TO 80
             REGH(I) = .2
         CASE IS >= 80
             REGH(I) = .38
      END SELECT
      SELECT CASE PI(I)
         CASE IS < 20
             ALFA(I) = 1
         CASE 20 TO 40
             ALFA(I) = .95
         CASE 40 TO 80
             ALFA(I) = .87
         CASE IS >= 80
             ALFA(I) = .75
      END SELECT
      B = NE - I + 1
      'G(I) = GBM(I) / (1 + (ALFA(I) * ((REGA ^ ALFA(I)) / REGH(I))))
      G(I) = F(B) * GBM(I) / (1 + (ALFA(I) * ((REGA ^ ALFA(I)) / REGH(I))))
      MAS(I) = (GE(I - 1) * TL(I - 1) / 2 + GE(I) * TL(I) / 2) / GRAV
      KEK(I) = G(I) * 1000 / TL(I)
      NEXT I
      FOR I = 1 TO NE
      PRINT #2, USING "## -######.## ###.#### ###.####"; LP(I); KE(I); SIGA(I);
      PRINT  #2,  USING  "###.####  ###.####  ####.####";  SIGA2(I);  SIGA0(I);
GAM(I)
      PRINT #2, USING "   -######.## ###.#### "; KED(I); SIGB(I);
      PRINT  #2,  USING  "###.####  ###.####  ####.####  ";  SIGB2(I);  SIGB0(I);
GBM(I);
      PRINT #2, USING "####.#### ####.#### ######.####"; G(I); MAS(I); KEK(I)
      NEXT I
      PRINT "MASSA SETIAP LAPIS TANAH"
      FOR I = 1 TO NE
      B = NE - I + 1
      M(I) = MAS(B)
      SIG0(I) = SIGB0(B)
      TTN(I) = TL(B)
      ALF(I) = ALFA(B)
      RGH(I) = REGH(B)
```

```
      GM(I) = GBM(B)
      GT(I) = G(B)
      NEXT I
      FOR J = 1 TO NE - 1
      FOR I = 1 TO NE - J
      IF MAS(I) < MAS(I + 1) THEN 200
      T = MAS(I)
      MAS(I) = MAS(I + 1)
      MAS(I + 1) = T
  200 NEXT I
      NEXT J
      UM = MAS(1)
      FOR I = NE TO 1 STEP -1
      PRINT "   Massa Lapis Tanah (kg.dt2/cm)  M("; I; ") = "; M(I)
      NEXT I
      PRINT "UNIT MASS OF ALL STORY UM = "; UM
      PRINT
      PRINT "KEKAKUAN SETIAP LAPIS TANAH"
      FOR I = 1 TO NE
      RMM#(I, I) = M(I) / UM
      NEXT I
      FOR I = 1 TO NE
      B = NE - I + 1
      K(I) = KEK(B)
      NEXT I
      FOR J = 1 TO NE - 1
      FOR I = 1 TO NE - J
      IF KEK(I) < KEK(I + 1) THEN 300
      T = KEK(I)
      KEK(I) = KEK(I + 1)
      KEK(I + 1) = T
  300 NEXT I
      NEXT J
      UK = KEK(1)
      FOR I = NE TO 1 STEP -1
      PRINT "   Kekakuan Lapis Tanah (kg/cm)  K("; I; ") = "; K(I)
      NEXT I
      PRINT "UNIT STIFNESS OF ALL STORY UK = "; UK
      PRINT
      PRINT "SEDANG MEMPROSES RESPON UNTUK INPUT DATA FILE : "; FILE$
      FOR I = 1 TO NE
      RK(I) = K(I) / UK
      NEXT I
      PRINT #2,
      PRINT #2, "C. HASIL RELATIVE-MASS MATRIX"
      FOR I = 1 TO NE
      FOR J = 1 TO NE
      PRINT #2, USING "####.######"; RMM#(I, J);
      NEXT J
      PRINT #2,
      NEXT I
      PRINT #2,
      PRINT #2, "D. HASIL MATRIX KEKAKUAN RELATIV"
      FOR I = 1 TO (NE - 1)
      IF I = 0 THEN RKM#(I, I) = RK(I) + RK(I + 1) + KA(I): GOTO 650
      RKM#(I, I) = RK(I) + RK(I + 1)
  650 RKM#(I, I + 1) = -RK(I + 1): RKM#(I + 1, I) = RKM#(I, I + 1)
```

```
NEXT I
RKM#(NE, NE) = RK(NE)
FOR I = 1 TO NE
FOR J = 1 TO NE
PRINT #2, USING "  #########.######"; RKM#(I, J);
NEXT J
PRINT #2,
NEXT I
PRINT #2,
FOR I = 1 TO NE
FOR J = 1 TO NE
AAA#(I, J) = RKM#(I, J) / RMM#(I, I)
NEXT J
NEXT I
FOR I = 1 TO NE
FOR J = 1 TO NE
NEXT J
NEXT I
FOR I = 1 TO NE
FOR J = 1 TO NE
BB#(I, J) = AAA#(I, J)
NEXT J
NEXT I
ME = NE - 1
FOR KE = 1 TO ME
TRACE# = 0
FOR I = 1 TO NE
TRACE# = TRACE# + BB#(I, I)
NEXT I
AK = KE
P#(KE) = TRACE# / AK
FOR I = 1 TO NE
BB#(I, I) = BB#(I, I) - P#(KE)
NEXT I
FOR J = 1 TO NE
FOR I = 1 TO NE
COLB#(I) = BB#(I, J)
NEXT I
FOR I = 1 TO NE
BB#(I, J) = 0
FOR LE = 1 TO NE
BB#(I, J) = BB#(I, J) + AAA#(I, LE) * COLB#(LE)
NEXT LE
NEXT I
NEXT J
NEXT KE
P#(NE) = BB#(1, 1)
R = (-1)
FOR KE = 1 TO NE
P#(KE) = R * (P#(KE))
NEXT KE
FOR I = 1 TO NE
AA#(I) = P#(I)
NEXT I
AA#(0) = 1
N = NE
```

```
     E = .00001
     K = 1
     R = 0
     IF N = 2 THEN 1810
1430 U#(K) = AA#(N - 1) / AA#(N - 2)
     V#(K) = AA#(N) / AA#(N - 2)
1450 B#(0) = AA#(0)
     B#(1) = AA#(1) - U#(K)
     FOR I = 2 TO N
     B#(I) = AA#(I) - B#(I - 1) * U#(K) - B#(I - 2) * V#(K)
     NEXT I
     C#(0) = B#(0)
     C#(1) = B#(1) - U#(K)
     FOR I = 2 TO N - 1
     C#(I) = B#(I) - C#(I - 1) * U#(K) - C#(I - 2) * V#(K)
     NEXT I
     DU#(K) = (B#(N - 1) * C#(N - 2) - B#(N) * C#(N - 3)) / (C#(N - 2) ^ 2 -
C#(N - 1) * C#(N - 3))
     DV#(K) = (C#(N - 2) * B#(N) - C#(N - 1) * B#(N - 1)) / (C#(N - 2) ^ 2 -
C#(N - 1) * C#(N - 3))
     U#(K) = U#(K) + DU#(K)
     V#(K) = V#(K) + DV#(K)
     IF ABS(DU#(K)) + ABS(DV#(K)) <= E THEN 1610
     GOTO 1450
1610 X#(K) = (-U#(K) + (U#(K) ^ 2 - 4 * 1 * V#(K)) ^ .5) / 2
     X#(K + 1) = (-U#(K) - (U#(K) ^ 2 - 4 * 1 * V#(K)) ^ .5) / 2
     D#(K + R) = X#(K): D#(K + 1 + R) = X#(K + 1)
     N = N - 2
     FOR S = 0 TO N
     AA#(S) = B#(S)
     NEXT S
     IF N = 2 THEN 1810
     IF N < 2 THEN 1780
     K = K + 1
     R = R + 1
     GOTO 1430
1780 D#(2 * K + 1) = -AA#(1)
     GOTO 1960
1810 X#(K) = (-AA#(1) + (AA#(1) ^ 2 - 4 * 1 * AA#(2)) ^ .5) / 2
     X#(K + 1) = (-AA#(1) - (AA#(1) ^ 2 - 4 * 1 * AA#(2)) ^ .5) / 2
     D#(2 * K + 1) = X#(K): D#(2 * K + 2) = X#(K + 1)
     GOTO 1930
1870 X#(K) = (-AA#(1) + (AA#(1) ^ 2 - 4 * 1 * AA#(2)) ^ .5) / 2
     X#(K + 1) = (-AA#(1) - (AA#(1) ^ 2 - 4 * 1 * AA#(2)) ^ .5) / 2
     D#(K) = X#(K): D#(K + 1) = X#(K + 1)
1930 REM
1960 FOR K = 1 TO NE: D(K) = D#(K): NEXT K
     PRINT #2, "E. HASIL INITIAL EIGENVALUE (LAMDA)"
     FOR J = 1 TO NE - 1
     FOR I = 1 TO NE - J
     IF D(I) < D(I + 1) THEN 2100
     T = D(I)
     D(I) = D(I + 1)
     D(I + 1) = T
2100 NEXT I
     NEXT J
     FOR K = 1 TO NE
```

```
     PRINT #2, USING "   ##  #######.######"; K; D(K)
     NEXT K
     FOR J = 1 TO NE
     W(J) = (D(J) * UK / UM) ^ .5
     WKW(J) = (D(J) * UK / UM)
     T(J) = 1 / ((2 * 3.14159) / W(J))
     NEXT J
     PRINT #2,
     PRINT #2, "F. HASIL FREKWENSI ALAM (rad/sec) DAN FREKWENSI NATURAL
(cps)"
     FOR I = 1 TO NE
     PRINT #2, USING "   ## ######.####  #######.####"; I; W(I); T(I)
     NEXT I
     MVC = NE
     PRINT #2,
     PRINT #2, "G. HASIL MODE SHAPE"
     IF NE > 2 THEN 2470
     FOR J = 1 TO NE: U(1, J) = 1: NEXT J
2430 FOR J = 1 TO NE
     U(2, J) = ((K(1) + K(2) - W(J) ^ 2 * M(1))) / K(2)
     NEXT J
     GOTO 2570
2470 FOR J = 1 TO MVC
     U(1, J) = 1
     NEXT J
     FOR J = 1 TO MVC
     FOR I = 3 TO NE
     U(2, J) = ((K(1) + K(2) - W(J) ^ 2 * M(1))) / K(2)
     U(I, J) = ((-U(I - 2, J)) * K(I - 1) + ((K(I - 1) + K(I) - W(J) ^ 2 *
M(I - 1)) * U(I - 1, J))) / K(I)
     NEXT I
     NEXT J
2570 FOR I = 1 TO NE
     FOR J = 1 TO MVC
     PRINT #2, USING " ###########.####"; U(I, J);
     NEXT J
     PRINT #2,
     NEXT I
     PRINT #2,
     PRINT #2, "H. PARTISIPASI SETIAP MODE"
     FOR I = 1 TO MVC
     P(I) = 0
     FOR J = 1 TO NE
     UT(I, J) = U(J, I)
     P(I) = P(I) + UT(I, J) * M(J)
     NEXT J
     NEXT I
     FOR I = 1 TO NE
     MS(I, I) = M(I)
     NEXT I
     FOR I = 1 TO MVC
     PM(I, J) = 0
     FOR J = 1 TO NE
     PM(I, J) = PM(I, J) + UT(I, J) * MS(J, J)
     NEXT J
     NEXT I
     FOR I = 1 TO MVC
```

```
      MM(I) = 0
      FOR J = 1 TO NE
      MM(I) = MM(I) + PM(I, J) * U(J, I)
      PA(I) = P(I) / MM(I)
      NEXT J
      PRINT #2, "    PA("; I; ") = "; PA(I)
      NEXT I
      PRINT #2,
      FOR I = 1 TO NE
      REM DAMPING RATIO = DR
      PD = .0387 * (EXP(1) ^ (.7876 * MAG))
      IF KT(I) = 1 THEN DRM(I) = (33 - (1.5 * LOG(PD)))
      IF KT(I) = 2 THEN DRM(I) = (31 - (3 + .03 * T(I)) * SIG0(I) ^ .5 + 1.5 *
T(I) ^ .5 - 1.5 * (LOG(PD)))
      IF KT(I) = 3 THEN DRM(I) = (28 - (1.5 * LOG(PD)))
      IF KT(I) = 4 THEN DRM(I) = (31 - (3 + .03 * T(I)) * SIG0(I) ^ .5 + 1.5 *
T(I) ^ .5 - 1.5 * (LOG(PD)))
      DR(I) = DRM(I) * (1 - (GT(I) / GM(I))) / 100
      'PRINT DR(I);
      NEXT I
      FOR I = 1 TO NE
      R(I) = WKW(I) - (2 / (DT ^ 2))
      Z(I) = (1 / DT ^ 2) - (2 * DR(I) * W(I) / (2 * DT))
      F(I) = (1 / DT ^ 2) + (2 * DR(I) * W(I) / (2 * DT))
      NEXT I
      PRINT #2, "I. KOEFISIEN I, a, b, k, & DAMPING RATIO  = "
      PRINT #2,
      PRINT #2, "    Lapis    a     b     k     DR"
      FOR I = 1 TO NE
      PRINT #2, USING "  ##"; I; TAB(8);
      PRINT #2, USING "  ########.####"; R(I); Z(I); F(I); DR(I)
      NEXT I
      INPUT #1, T
      NJ = T / .01
      DIM TT(NJ), YY(NJ)
      FOR I = 1 TO NJ
      INPUT #1, TT(I), YY(I)
      NEXT I
      DIM GD(NJ, NE), ZD(NJ, NE)
      FOR I = 1 TO NJ - 1
      FOR J = 1 TO NE
      GD(I + 1, J) = (-YY(I) - R(J) * GD(I, J) - Z(J) * GD(I - 1, J)) / F(J)
      ZD(I, J) = GD(I, J) * PA(J)
      NEXT J
      NEXT I
      ERASE GD
      DIM YD(NJ, NE)
      FOR I = 1 TO NJ - 1
      FOR J = 1 TO NE
      YD(I, J) = 0
      FOR K = 1 TO MVC
      YD(I, J) = YD(I, J) + U(J, K) * ZD(I, K)
      NEXT K
      NEXT J
      NEXT I
      ERASE ZD
```

```
      PRINT #2,
      PRINT #2, "J. SIMPANGAN TANAH LINIER ELASTIS TIAP LAPIS"
      PRINT #2,
      PRINT #2, "   No.      Waktu      Percepatan  ";
      FOR J = 1 TO NE
      PRINT #2, ""; J; "";
      NEXT J
      FOR J = 1 TO NJ - 1
      PRINT #2,
      PRINT #2, USING " ######   #####.##"; J; TT(J);
      PRINT #2, USING " #####.#### "; YY(J);
      FOR I = 1 TO NE
      PRINT #2, USING "  #####.#######"; YD(J, I);
      NEXT I
      NEXT J
      DIM REGA(NJ, NE)
      FOR J = 1 TO NJ - 1
      FOR I = 1 TO NE
      REGA(J, I) = (YD(J, I) - YD(J, I - 1)) * 100 / (TTN(I))
      NEXT I
      NEXT J
      ERASE YD
      DIM GN(NJ, NE)
      FOR J = 1 TO NJ - 1
      FOR I = 1 TO NE
      GN(J, I) = GM(I) / (1 + (ALF(I) * ((ABS(REGA(J, I)) ^ ALF(I)) /
RGH(I))))
      NEXT I
      NEXT J
      ERASE REGA
      DIM KN(NJ, NE)
      FOR J = 1 TO NJ - 1
      FOR I = 1 TO NE
      KN(J, I) = GN(J, I) * 1000 / TTN(I)
      NEXT I
      NEXT J
      ERASE GN
      PRINT #2,
      PRINT #2,
      PRINT #2, "K. RESPON TANAH NON LINIER ELASTIS TIAP LAPIS"
      PRINT #2,
      PRINT #2, "  Mode Shape, Partisipasi Mode, Damping Ratio, a, b, k,";
      PRINT #2, " dan Frekwensi Natural"
      PRINT
      PRINT "SEDANG MENGANALISIS STEP INTEGRASI KE : "
      DO WHILE M < NJ - 1
      M = M + 1
      ERASE U, UT, PM, K, R, Z, F, W, WKW, PA
      PRINT M;
      PRINT #2,
      FOR I = 1 TO NE
      K(I) = KN(M, I)
      NEXT I
      FOR J = 1 TO NE - 1
      FOR I = 1 TO NE - J
      IF KN(M, I) < KN(M, I + 1) THEN 2300
      T = KN(M, I)
```

```
      KN(M, I) = KN(M, I + 1)
      KN(M, I + 1) = T
2300  NEXT I
      NEXT J
      UK = KN(M, 1)
      FOR I = 1 TO NE
      RK(I) = K(I) / UK
      NEXT I
      FOR I = 1 TO (NE - 1)
      IF I = 0 THEN RKM#(I, I) = RK(I) + RK(I + 1) + KA(I): GOTO 2650
      RKM#(I, I) = RK(I) + RK(I + 1)
2650  RKM#(I, I + 1) = -RK(I + 1): RKM#(I + 1, I) = RKM#(I, I + 1)
      NEXT I
      RKM#(NE, NE) = RK(NE)
      FOR I = 1 TO NE
      FOR J = 1 TO NE
      AAA#(I, J) = RKM#(I, J) / RMM#(I, I)
      NEXT J
      NEXT I
      FOR I = 1 TO NE
      FOR J = 1 TO NE
      BB#(I, J) = AAA#(I, J)
      NEXT J
      NEXT I
      ME = NE - 1
      FOR KE = 1 TO ME
      TRACE# = 0
      FOR I = 1 TO NE
      TRACE# = TRACE# + BB#(I, I)
      NEXT I
      AK = KE
      P#(KE) = TRACE# / AK
      FOR I = 1 TO NE
      BB#(I, I) = BB#(I, I) - P#(KE)
      NEXT I
      FOR J = 1 TO NE
      FOR I = 1 TO NE
      COLB#(I) = BB#(I, J)
      NEXT I
      FOR I = 1 TO NE
      BB#(I, J) = 0
      FOR LE = 1 TO NE
      BB#(I, J) = BB#(I, J) + AAA#(I, LE) * COLB#(LE)
      NEXT LE
      NEXT I
      NEXT J
      NEXT KE
      P#(NE) = BB#(1, 1)
      R = (-1)
      FOR KE = 1 TO NE
      P#(KE) = R * (P#(KE))
      NEXT KE
      FOR I = 1 TO NE
      AA#(I) = P#(I)
      NEXT I
      AA#(0) = 1
      N = NE
```

```
     IF N = 2 THEN K = 1: GOTO 3870
     E = .00001
     K = 1
     R = 0
     IF N = 2 THEN 3810
3430 U#(K) = AA#(N - 1) / AA#(N - 2)
     V#(K) = AA#(N) / AA#(N - 2)
3450 B#(0) = AA#(0)
     B#(1) = AA#(1) - U#(K)
     FOR I = 2 TO N
     B#(I) = AA#(I) - B#(I - 1) * U#(K) - B#(I - 2) * V#(K)
     NEXT I
     C#(0) = B#(0)
     C#(1) = B#(1) - U#(K)
     FOR I = 2 TO N - 1
     C#(I) = B#(I) - C#(I - 1) * U#(K) - C#(I - 2) * V#(K)
     NEXT I
     DU#(K) = (B#(N - 1) * C#(N - 2) - B#(N) * C#(N - 3)) / (C#(N - 2) ^ 2 -
C#(N - 1) * C#(N - 3))
     DV#(K) = (C#(N - 2) * B#(N) - C#(N - 1) * B#(N - 1)) / (C#(N - 2) ^ 2 -
C#(N - 1) * C#(N - 3))
     U#(K) = U#(K) + DU#(K)
     V#(K) = V#(K) + DV#(K)
     IF ABS(DU#(K)) + ABS(DV#(K)) <= E THEN 3610
     GOTO 3450
3610 X#(K) = (-U#(K) + (U#(K) ^ 2 - 4 * 1 * V#(K)) ^ .5) / 2
     X#(K + 1) = (-U#(K) - (U#(K) ^ 2 - 4 * 1 * V#(K)) ^ .5) / 2
     D#(K + R) = X#(K): D#(K + 1 + R) = X#(K + 1)
     N = N - 2
     FOR S = 0 TO N
     AA#(S) = B#(S)
     NEXT S
     IF N = 2 THEN 3810
     IF N < 2 THEN 3780
     K = K + 1
     R = R + 1
     GOTO 3430
3780 D#(2 * K + 1) = -AA#(1)
     GOTO 3960
3810 X#(K) = (-AA#(1) + (AA#(1) ^ 2 - 4 * 1 * AA#(2)) ^ .5) / 2
     X#(K + 1) = (-AA#(1) - (AA#(1) ^ 2 - 4 * 1 * AA#(2)) ^ .5) / 2
     D#(2 * K + 1) = X#(K): D#(2 * K + 2) = X#(K + 1)
     GOTO 3930
3870 X#(K) = (-AA#(1) + (AA#(1) ^ 2 - 4 * 1 * AA#(2)) ^ .5) / 2
     X#(K + 1) = (-AA#(1) - (AA#(1) ^ 2 - 4 * 1 * AA#(2)) ^ .5) / 2
     D#(K) = X#(K): D#(K + 1) = X#(K + 1)
3930 REM
3960 FOR K = 1 TO NE: D(K) = D#(K): NEXT K
     FOR J = 1 TO NE - 1
     FOR I = 1 TO NE - J
     IF D(I) < D(I + 1) THEN 4100
     T = D(I)
     D(I) = D(I + 1)
     D(I + 1) = T
4100 NEXT I
     NEXT J
     FOR J = 1 TO NE
```

```
      W(J) = (D(J) * UK / UM) ^ .5
      WKW(J) = (D(J) * UK / UM)
      NEXT J
      MVC = NE
      IF NE > 2 THEN 4470
      FOR J = 1 TO NE: U(1, J) = 1: NEXT J
4430  FOR J = 1 TO NE
      U(2, J) = ((K(1) + K(2) - WKW(J) * M(1))) / K(2)
      NEXT J
      GOTO 4570
4470  FOR J = 1 TO MVC
      U(1, J) = 1
      NEXT J
      FOR J = 1 TO MVC
      FOR I = 3 TO NE
      U(2, J) = ((K(1) + K(2) - W(J) ^ 2 * M(1))) / K(2)
      U(I, J) = ((-U(I - 2, J)) * K(I - 1) + ((K(I - 1) + K(I) - W(J) ^ 2 *
M(I - 1)) * U(I - 1, J))) / K(I)
      NEXT I
      NEXT J
4570  FOR R = 1 TO NE
      FOR I = 1 TO MVC
      P(I) = 0
      FOR J = 1 TO NE
      UT(I, J) = U(J, I)
      P(I) = P(I) + UT(I, J) * M(J)
      NEXT J
      PRINT #2, USING "  ########.#######"; U(R, I);
      NEXT I
      NEXT R
      FOR I = 1 TO NE
      MS(I, I) = M(I)
      NEXT I
      FOR I = 1 TO MVC
      PM(I, J) = 0
      FOR J = 1 TO NE
      PM(I, J) = PM(I, J) + UT(I, J) * MS(J, J)
      NEXT J
      NEXT I
      FOR I = 1 TO MVC
      MM(I) = 0
      FOR J = 1 TO NE
      MM(I) = MM(I) + PM(I, J) * U(J, I)
      PA(I) = P(I) / MM(I)
      NEXT J
      PRINT #2, USING "  ########.#######"; PA(I);
      NEXT I
      FOR I = 1 TO NE
      REM DAMPING RATIO = DR
      PD = .0387 * (EXP(1) ^ (.7876 * MAG))
      T(I) = 1 / ((2 * 3.14159) / W(I))
      IF KT(I) = 1 THEN DRM(I) = (33 - (1.5 * LOG(PD)))
      IF KT(I) = 2 THEN DRM(I) = (31 - (3 + .03 * T(I)) * SIG0(I) ^ .5 + 1.5 *
T(I) ^ .5 - 1.5 * (LOG(PD)))
      IF KT(I) = 3 THEN DRM(I) = (28 - (1.5 * LOG(PD)))
      IF KT(I) = 4 THEN DRM(I) = (31 - (3 + .03 * T(I)) * SIG0(I) ^ .5 + 1.5 *
T(I) ^ .5 - 1.5 * (LOG(PD)))
```

```
DR(I) = DRM(I) * (1 - (GT(I) / GM(I))) / 100
PRINT #2, USING "  ########.#######"; DR(I);
NEXT I
FOR I = 1 TO NE
R(I) = WKW(I) - (2 / (DT ^ 2))
Z(I) = (1 / DT ^ 2) - (2 * DR(I) * W(I) / (2 * DT))
F(I) = (1 / DT ^ 2) + (2 * DR(I) * W(I) / (2 * DT))
PRINT #2, USING "  ########.#######"; R(I);
NEXT I
FOR I = 1 TO NE
PRINT #2, USING "  ########.#######"; Z(I);
NEXT I
FOR I = 1 TO NE
PRINT #2, USING "  ########.#######"; F(I);
NEXT I
FOR I = 1 TO NE
PRINT #2, USING "  ########.#######"; T(I);
NEXT I
LOOP
PRINT #2,
ERASE GN, KN
DIM GD(NJ, NE)
FOR I = 1 TO NJ - 1
FOR J = 1 TO NE
GD(I + 1, J) = (-YY(I) - R(J) * GD(I, J) - Z(J) * GD(I - 1, J)) / F(J)
NEXT J
NEXT I
DIM GV(NJ, NE)
FOR I = 1 TO NJ - 1
FOR J = 1 TO NE
GV(I, J) = (GD(I + 1, J) - GD(I - 1, J)) / (2 * DT)
NEXT J
NEXT I
DIM ZD(NJ, NE)
FOR I = 1 TO NJ - 1
FOR J = 1 TO NE
ZD(I, J) = GD(I + 1, J) * PA(J)
NEXT J
NEXT I
ERASE GD
DIM YD(NJ, NE)
FOR I = 1 TO NJ - 1
FOR J = 1 TO NE
YD(I, J) = 0
FOR K = 1 TO MVC
YD(I, J) = YD(I, J) + U(J, K) * ZD(I, K)
NEXT K
NEXT J
NEXT I
ERASE ZD
DIM ZV(NJ, NE)
FOR I = 1 TO NJ - 1
FOR J = 1 TO NE
ZV(I, J) = GV(I, J) * PA(J)
NEXT J
NEXT I
ERASE GV
```

```
DIM YV(NJ, NE)
FOR I = 1 TO NJ - 1
FOR J = 1 TO NE
YV(I, J) = 0
FOR K = 1 TO MVC
YV(I, J) = YV(I, J) + U(J, K) * ZV(I, K)
NEXT K
NEXT J
NEXT I
ERASE ZV, YD
PRINT #2,
PRINT #2, "L. KECEPATAN TANAH LINIER ELASTIS TIAP LAPIS"
PRINT #2,
PRINT #2, "   No.      ti        Yti  ";
FOR I = 1 TO NE
PRINT #2, ""; I; "";
NEXT I
FOR J = 1 TO NJ - 1
PRINT #2,
PRINT #2, USING " #####    ####.##"; J; TT(J);
PRINT #2, USING " #####.#### "; YY(J);
FOR I = 1 TO NE
PRINT #2, USING " #####.#######"; YV(J, I);
NEXT I
NEXT J
PRINT #2,
ERASE YV
DIM GD(NJ, NE), GA(NJ, NE)
FOR I = 1 TO NJ - 1
FOR J = 1 TO NE
GD(I + 1, J) = (-YY(I) - R(J) * GD(I, J) - Z(J) * GD(I - 1, J)) / F(J)
GA(I, J) = (GD(I + 1, J) - (2 * GD(I, J)) + GD(I - 1, J)) / (DT ^ 2)
NEXT J
NEXT I
DIM ZD(NJ, NE)
FOR I = 1 TO NJ - 1
FOR J = 1 TO NE
ZD(I, J) = GD(I, J) * PA(J)
NEXT J
NEXT I
ERASE GD
DIM YD(NJ, NE)
FOR I = 1 TO NJ - 1
FOR J = 1 TO NE
YD(I, J) = 0
FOR K = 1 TO MVC
YD(I, J) = YD(I, J) + U(J, K) * ZD(I, K)
NEXT K
NEXT J
NEXT I
ERASE ZD
DIM ZA(NJ, NE)
FOR I = 1 TO NJ - 1
FOR J = 1 TO NE
ZA(I, J) = GA(I, J) * PA(J)
NEXT J
```

```
NEXT I
ERASE GA
DIM YA(NJ, NE)
FOR I = 1 TO NJ - 1
FOR J = 1 TO NE
YA(I, J) = 0
FOR K = 1 TO MVC
YA(I, J) = YA(I, J) + U(J, K) * ZA(I, K)
NEXT K
NEXT J
NEXT I
ERASE ZA
PRINT #2,
PRINT #2, "M. PERCEPATAN TANAH LINIER ELASTIS TIAP LAPIS"
PRINT #2,
PRINT #2, "   No.      ti        Yti  ";
FOR I = 1 TO NE
PRINT #2, "  "; I; "";
NEXT I
FOR J = 1 TO NJ - 1
PRINT #2,
PRINT #2, USING " #####   ####.##   ####.#####"; J; TT(J); YY(J);
FOR I = 1 TO NE
PRINT #2, USING " #####.#######"; YA(J, I);
NEXT I
NEXT J
ERASE YA
CLOSE
PRINT
PRINT
PRINT "HASIL SIMPANGAN LINIER ELASTIS DAN RESPON NON LINIER ELASTIS"
PRINT "SELESAI... HASIL DAPAT DIBUKA PADA FILE : RESPON.DAT "
END
```

Lampiran II. Input Data PGV, dengan Gempa Bucharest.

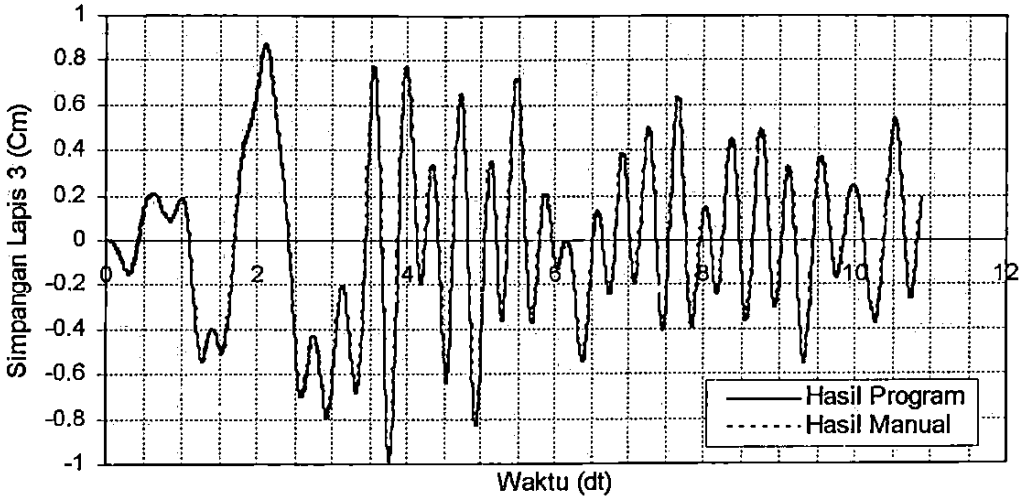
3 0
500 500 1000
15 20 25
0 0 0
0.55 0.8 0.85
2.68 2.71 2.72
1.47 1.58 1.6
2 4 4
980 0.01
140 7.1
0.80038 0.72595 0.88868
10
0 0
0.01 0.954434783
0.02 1.908869565
0.03 2.863304348
0.04 3.81773913
0.05 4.772173913
0.06 5.726608696
0.07 6.681043478
0.08 7.635478261
0.09 8.589913043
0.1 9.544347826
0.11 10.49878261
0.12 11.45321739
0.13 12.40765217
0.14 13.36208696
0.15 14.31652174
0.16 15.27095652
0.17 16.2253913
0.18 17.17982609
0.19 18.13426087
0.2 19.08869565
0.21 20.04313043
dan seterusnya sampai dengan selesai.

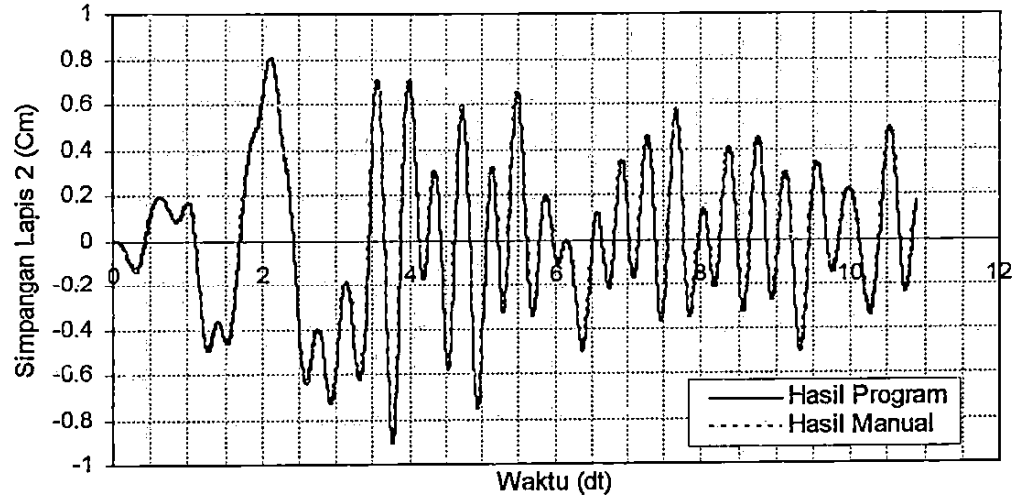
Lampiran III. Hasil Analisis Simpangan.

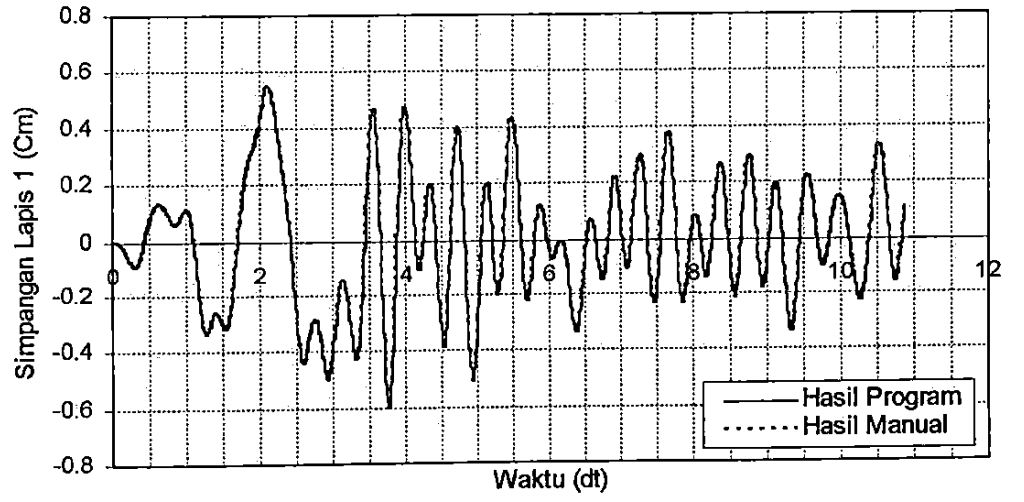
J. SIMPANGAN TANAH LINIER ELASTIS TIAP LAPIS

No.	Waktu	Percepatan	1	2	3
1	0.00	0.0000	0.0000000	0.0000000	0.0000000
2	0.01	0.9544	0.0000000	0.0000000	0.0000000
3	0.02	1.9089	-0.0000934	-0.0000947	-0.0000950
4	0.03	2.8633	-0.0003624	-0.0003770	-0.0003791
5	0.04	3.8177	-0.0008741	-0.0009368	-0.0009446
6	0.05	4.7722	-0.0016785	-0.0018586	-0.0018810
7	0.06	5.7266	-0.0028093	-0.0032178	-0.0032732
8	0.07	6.6810	-0.0042872	-0.0050780	-0.0051986
9	0.08	7.6355	-0.0061225	-0.0074884	-0.0077234
10	0.09	8.5899	-0.0083194	-0.0104823	-0.0108978
11	0.10	9.5443	-0.0108781	-0.0140773	-0.0147525
12	0.11	10.4988	-0.0137964	-0.0182757	-0.0192961
13	0.12	11.4532	-0.0170695	-0.0230653	-0.0245139
14	0.13	12.4077	-0.0206897	-0.0284205	-0.0303702
15	0.14	13.3621	-0.0246446	-0.0343026	-0.0368117
16	0.15	14.3165	-0.0289148	-0.0406611	-0.0437717
17	0.16	15.2710	-0.0334729	-0.0474348	-0.0511747
18	0.17	16.2254	-0.0382821	-0.0545536	-0.0589403
19	0.18	17.1798	-0.0432963	-0.0619412	-0.0669854
20	0.19	18.1343	-0.0484609	-0.0695172	-0.0752259
21	0.20	19.0887	-0.0537152	-0.0771997	-0.0835773
22	0.21	20.0431	-0.0589951	-0.0849074	-0.0919553
23	0.22	20.9976	-0.0642367	-0.0925608	-0.1002765
24	0.23	21.9520	-0.0693794	-0.1000835	-0.1084597
25	0.24	22.9064	-0.0743692	-0.1074032	-0.1164270
26	0.25	23.8609	-0.0791599	-0.1144526	-0.1241054
27	0.26	22.3696	-0.0837147	-0.1211716	-0.1314285
28	0.27	20.8783	-0.0877665	-0.1272654	-0.1380946
29	0.28	19.3870	-0.0910847	-0.1324542	-0.1438144
30	0.29	17.8957	-0.0934853	-0.1364770	-0.1483142
31	0.30	16.4044	-0.0948342	-0.1390991	-0.1513404

dan seterusnya sampai dengan selesai


Simpangan Lapis 3 (Cm)
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
2
4
6
8
10
12
Hasil Program
Hasil Manual
Waktu (dt)


Simpangan Lapis 2 (Cm)
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
2
4
6
8
10
12
Hasil Program
Hasil Manual
Waktu (dt)

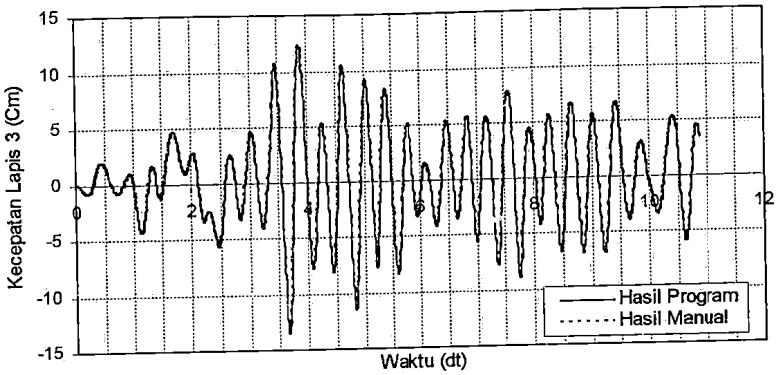

Simpangan Lapis 1 (Cm)
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
0
2
4
6
8
10
12
Hasil Program
Hasil Manual
Waktu (dt)

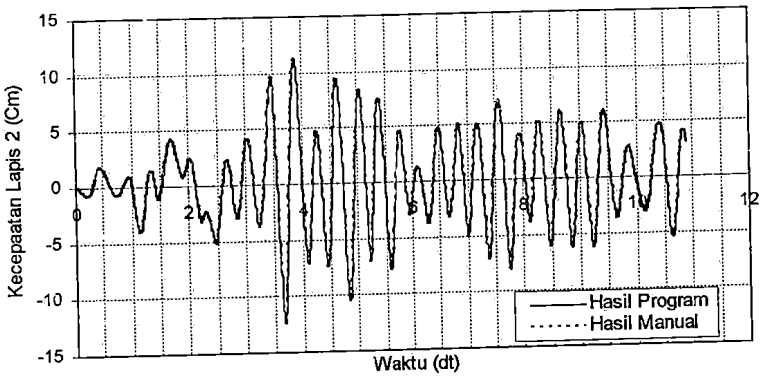
Lampiran IV. Hasil Analisis Kecepatan.

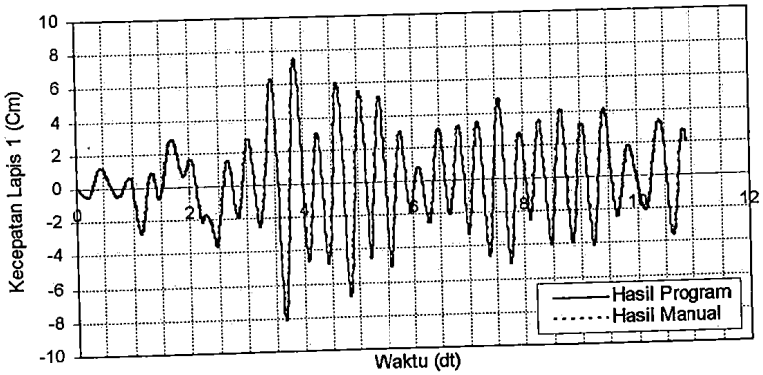
K. HASIL KECEPATAN TANAH LINIER ELASTIS TIAP LAPIS

No.	Waktu	Percepatan	1	2	3
1	0.00	0.0000	0.0000000	0.0000000	0.0000000
2	0.01	0.9544	-0.0046677	-0.0047338	-0.0047523
3	0.02	1.9089	-0.0181203	-0.0188489	-0.0189570
4	0.03	2.8633	-0.0390384	-0.0421070	-0.0424780
5	0.04	3.8177	-0.0658036	-0.0740792	-0.0750921
6	0.05	4.7722	-0.0967608	-0.1140483	-0.1164289
7	0.06	5.7266	-0.1304343	-0.1609735	-0.1658819
8	0.07	6.6810	-0.1656582	-0.2135316	-0.2225087
9	0.08	7.6355	-0.2016135	-0.2702151	-0.2849570
10	0.09	8.5899	-0.2377819	-0.3294461	-0.3514588
11	0.10	9.5443	-0.2738460	-0.3896691	-0.4199151
12	0.11	10.4988	-0.3095669	-0.4494003	-0.4880666
13	0.12	11.4532	-0.3446694	-0.5072392	-0.5537079
14	0.13	12.4077	-0.3787543	-0.5618628	-0.6148924
15	0.14	13.3621	-0.4112510	-0.6120300	-0.6700729
16	0.15	14.3165	-0.4414142	-0.6566090	-0.7181500
17	0.16	15.2710	-0.4683653	-0.6946267	-0.7584316
18	0.17	16.2254	-0.4911700	-0.7253214	-0.7905363
19	0.18	17.1798	-0.5089406	-0.7481768	-0.8142813
20	0.19	18.1343	-0.5209465	-0.7629238	-0.8295937
21	0.20	19.0887	-0.5267105	-0.7695106	-0.8364667
22	0.21	20.0431	-0.5260742	-0.7680561	-0.8349592
23	0.22	20.9976	-0.5192170	-0.7588074	-0.8252210
24	0.23	21.9520	-0.5066261	-0.7421187	-0.8075234
25	0.24	22.9064	-0.4890249	-0.7184556	-0.7822838
26	0.25	23.8609	-0.4672779	-0.6884198	-0.7500778
27	0.26	22.3696	-0.4303303	-0.6406401	-0.6994637
28	0.27	20.8783	-0.3684987	-0.5641319	-0.6192946
29	0.28	19.3870	-0.2859406	-0.4605756	-0.5109786
30	0.29	17.8957	-0.1874732	-0.3322420	-0.3763004
31	0.30	16.4044	-0.0779137	-0.1821865	-0.2175547

dan seterusnya sampai dengan selesai


Kecepatan Lapis 3 (Cm)
15
10
5
0
-5
-10
-15
0
2
4
6
8
10
12
Hasil Program
Hasil Manual
Waktu (dt)


Kecepaatan Lapis 2 (Cm)
15
10
5
0
-5
-10
-15
0
2
4
6
8
10
12
Hasil Program
Hasil Manual
Waktu (dt)

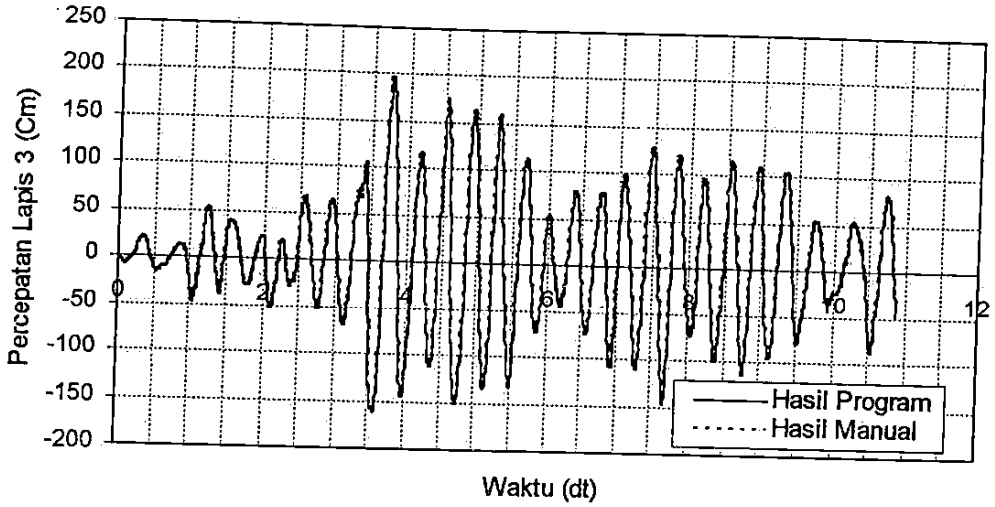

Kecepatan Lapis 1 (Cm)
10
8
6
4
2
0
-2
-4
-6
-8
-10
0
2
4
6
8
10
12
Hasil Program
Hasil Manual
Waktu (dt)

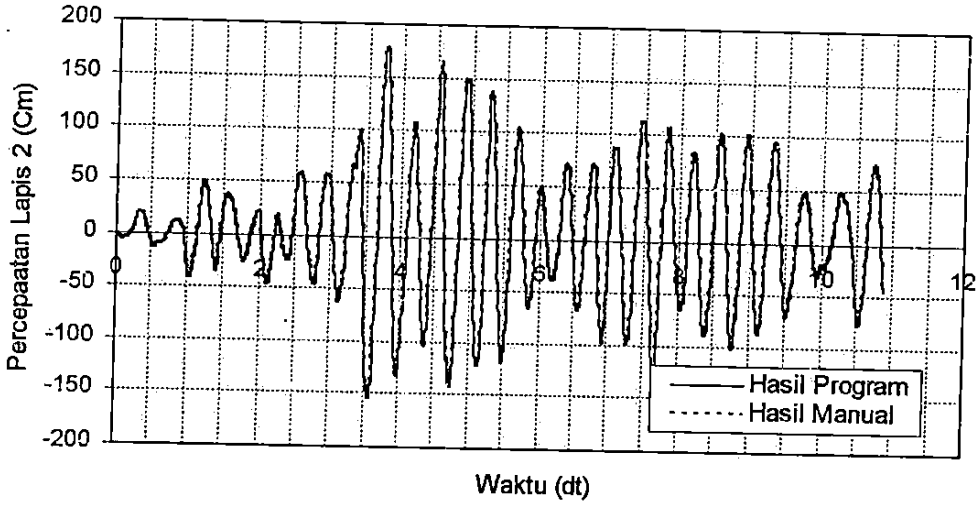
Lampiran V. Hasil Analisis Percepatan.

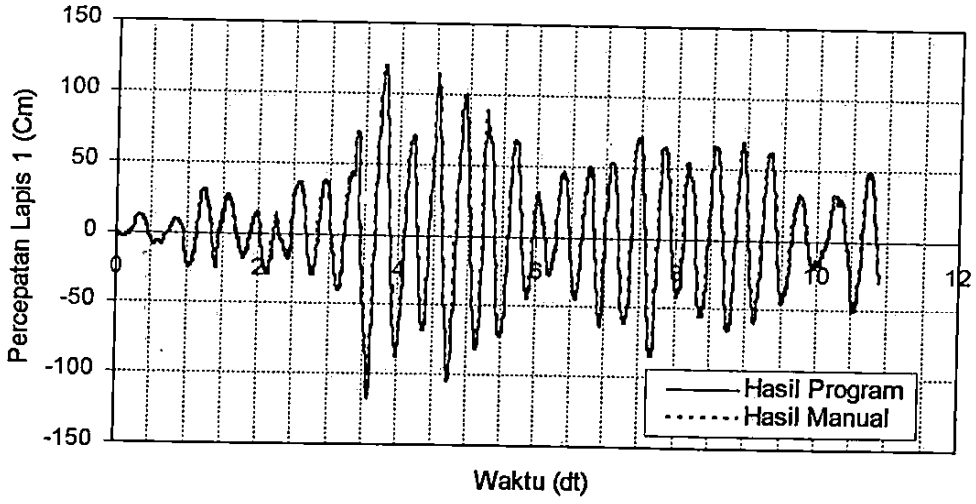
L. HASIL PERCEPATAN TANAH LINIER ELASTIS TIAP LAPIS

No.	Waktu	Percepatan	1	2	3
1	0.00	0.00000	0.0000000	0.0000000	0.0000000
2	0.01	0.95443	-0.9335454	-0.9467666	-0.9504559
3	0.02	1.90887	-1.7569661	-1.8762403	-1.8904904
4	0.03	2.86330	-2.4266536	-2.7753825	-2.8136997
5	0.04	3.81774	-2.9263816	-3.6190710	-3.7091224
6	0.05	4.77217	-3.2650762	-4.3747463	-4.5582418
7	0.06	5.72661	-3.4696124	-5.0102801	-5.3323650
8	0.07	6.68104	-3.5751684	-5.5013437	-5.9929886
9	0.08	7.63548	-3.6158919	-5.8353539	-6.4966803
10	0.09	8.58991	-3.6177974	-6.0108542	-6.8036656
11	0.10	9.54435	-3.5950153	-6.0337386	-6.8876028
12	0.11	10.49878	-3.5491691	-5.9125109	-6.7426853
13	0.12	11.45322	-3.4713299	-5.6552558	-6.3855696
14	0.13	12.40765	-3.3456542	-5.2694688	-5.8513422
15	0.14	13.36209	-3.1536784	-4.7639780	-5.1847739
16	0.15	14.31652	-2.8789668	-4.1518245	-4.4306173
17	0.16	15.27096	-2.5112534	-3.4517109	-3.6257014
18	0.17	16.22539	-2.0496769	-2.6872189	-2.7952676
19	0.18	17.17983	-1.5044572	-1.8838573	-1.9537255
20	0.19	18.13426	-0.8967198	-1.0655541	-1.1087298
21	0.20	19.08870	-0.2560822	-0.2518232	-0.2658846
22	0.21	20.04313	0.3833525	0.5427266	0.5673658
23	0.22	20.99756	0.9880954	1.3070157	1.3802973
24	0.23	21.95200	1.5300891	2.0307529	2.1592238
25	0.24	22.90644	1.9901339	2.7018421	2.8886762
26	0.25	23.86087	2.3592646	3.3053253	3.5525305
27	0.26	22.36960	5.0302558	6.2506142	6.5703058
28	0.27	20.87826	7.3360624	9.0510225	9.4635029
29	0.28	19.38699	9.1755714	11.6602545	12.1996984
30	0.29	17.89565	10.5179014	14.0064449	14.7359428
31	0.30	16.40438	11.3939991	16.0046654	17.0131950

dan seterusnya sampai dengan selesai


Percepatan Lapis 3 (Cm)
250
200
150
100
50
0
-50
-100
-150
-200
0
2
4
6
8
10
12
Hasil Program
Hasil Manual
Waktu (dt)


Percepaatan Lapis 2 (Cm)
200
150
100
50
0
-50
-100
-150
-200
0
2
4
6
8
10
12
Hasil Program
Hasil Manual
Waktu (dt)


Percepatan Lapis 1 (Cm)
150
100
50
0
-50
-100
-150
0
2
4
6
8
10
12
Hasil Program
Hasil Manual
Waktu (dt)

LAMPIRAN VI. RIWAYAT HIDUP PENELITI

I. DATA PRIBADI

Nama Lengkap	:	Ir. As'at Pujianto, MT.
Tempat, Tanggal Lahir	:	Temanggung, 14 April 1966.
Jenis Kelamin	:	Laki-laki.
Agama	:	Islam.
Pekerjaan	:	Dosen Tetap Fakultas Teknik Universitas Muhammadiyah Yogyakarta.
Alamat Rumah	:	Kembang Tamantirto Kasihan Bantul Yogyakarta.
Alamat Kantor	:	Jln. HOS. Cokroaminoto No. 17 Yogyakarta. Telp. (0274) 618053 Fax. (0274) 618166.

II. RIWAYAT PENDIDIKAN.

Tahun 1977 Tamat dari Sekolah Dasar di Temanggung Jawa Tengah

Tahun 1981 Tamat dari Sekolah Menengah Pertama di Temanggung Jawa Tengah.

Tahun 1984 Tamat dari Sekolah Menengah Atas di Temanggung Jawa Tengah.

Tahun 1990 Mendapat Ijazah Lokal dari Jurusan Teknik Sipil FT. UMY.

Tahun 1993 Mendapat Ijazah Negara dari Jurusan Teknik Sipil
Fakultas Teknik Universitas Muhammadiyah Yogyakarta.

Tahun 2003 Mendapat Ijazah Pascasarjana Magister Teknik Sipil
Universitas Islam IndonesiaYogyakarta.

III. RIWAYAT PEKERJAAN.

Tahun 1991-1992 PT. WIJAYA KARYA Sudirman Square Project (BRI II) Jakarta.

Tahun 1992-1993 PT. PERENTJANA DJAJA, Proyek Jalan & Jembatan di Maluku.

Tahun 1993-1994 PT. Sinca Mataram, Proyek Gedung Rektoriat IAIN Yogyakarta.

Tahun 1993-Sekarang DOSEN TETAP Jurusan Teknik Sipil Fakultas Teknik UMY.

IV. PENGALAMAN PENELITIAN.

1. Program Komputer dan Analisis Grid Dengan Metoda Kekakuan.
2. *Respon Seismik Lapisan Tanah Linier dan Non Linier Elastis Akibat Beban Gempa.*
3. *Validasi Parameter Percepatan Tanah dan Efek Frekuensi Gempa Terhadap Respon Struktur Bangunan Bertingkat.*