LAMPIRAN

LAMPIRAN I

LISTING PROGRAM RESPON STRUKTUR MDOF

```
REM  PROGRAM RESPON STRUKTUR MDOF
     CLS
     OPEN "I", #1, "bucha.DAT"
     OPEN "O", #2, "OUT.DAT"
     GRAV = 980:          'Gaya Gravitasi (cm/dt2)
     DT = .01:            'Step Integrasi (dt)
     INPUT #1, NE:        'JUMLAH TINGKAT BANGUNAN
     INPUT #1, NC:        'JUMLAH KOLOM UNTUK SATU PORTAL
     INPUT #1, E:         'MODULUS ELASTIS (kg/cm2)
     INPUT #1, PB:        'PANJANG BALOK (cm)
     INPUT #1, DR:        'RATIO REDAMAN (%)
     FOR I = 1 TO NE
     INPUT #1, Q(I):      'Beban Bangunan (kg/cm)
     NEXT I
     FOR I = 1 TO NE
     INPUT #1, TL(I):     'TINGGI TINGKAT BANGUNAN(cm)
     NEXT I
     FOR I = 1 TO NE
     INPUT #1, LKX(I):    'LEBAR KOLOM B
     NEXT I
     FOR I = 1 TO NE
     INPUT #1, LKY(I):    'LEBAR KOLOM H
     NEXT I

     PRINT #2, "A. INPUT DATA BANGUNAN BERTINGKAT"
     PRINT #2,
     PRINT #2, "   Tingkat  Lebar-B Lebar-H  Kekakuan     Massa  "
     PRINT #2,
     FOR I = 1 TO NE
     IK(I) = LKX(I) * (LKY(I) ^ 3) / 12
     KEK(I) = 12 * E * IK(I) / (TL(I) ^ 3)
     K(I) = NC * KEK(I)
     NEXT I
     FOR I = 1 TO NE
     MAS(I) = Q(I) * PB / GRAV
     NEXT I
     FOR I = 1 TO NE
     LP(I) = I
     PRINT #2, USING "    #        ###      ###    ########.## ####.####";
LP(I); LKX(I); LKY(I); K(I); MAS(I)
     NEXT I
     FOR I = 1 TO NE
     M(I) = MAS(I)
     K(I) = KEK(I)
     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)
      PRINT "MASSA SETIAP LANTAI"
      FOR I = NE TO 1 STEP -1
      PRINT "   Massa Lantai (kg.dt2/cm)  M("; I; ") = "; M(I)
      NEXT I
      PRINT "UNIT MASS OF ALL STORY UM = "; UM
      PRINT
      PRINT "KEKAKUAN SETIAP TINGKAT"
      FOR I = 1 TO NE
      RMM#(I, I) = M(I) / UM
      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 Tingkat (kg/cm)  K("; I; ") = "; K(I)
      NEXT I
      PRINT "UNIT STIFNESS OF ALL STORY UK = "; UK
      PRINT
      FOR I = 1 TO NE
      RK(I) = K(I) / UK
      NEXT I
      PRINT #2,
      PRINT #2, "B. HASIL RELATIVE-MASS MATRIX"
      PRINT #2,
      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, "C. HASIL MATRIX KEKAKUAN RELATIV"
      PRINT #2,
      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
      PRINT #2, "D. HASIL INITIAL EIGENPROBLEM MATRIX"
      PRINT #2,
      FOR I = 1 TO NE
      FOR J = 1 TO NE
      PRINT #2, USING "  ######.#######"; AAA#(I, J);
      NEXT J
      PRINT #2,
      NEXT I
      PRINT #2,
      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 1870
      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)"
      PRINT #2,
      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) / 6.3))
```

```
      PG(J) = (2 * 3.14159) / (W(J) / 6.3)
      NEXT J
      PRINT #2,
      PRINT #2, "F. FREK NATURAL (cps), PERIODE GETAR (dt) "
      PRINT #2,
      FOR I = 1 TO NE
      PRINT #2, USING "   ## #####.####  ###.#####"; I; T(I); PG(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  PRINT #2,
      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
      FOR I = 1 TO MVC
      FOR J = 1 TO NE
      UT(I, J) = U(J, I)
      NEXT J
      NEXT I
      FOR I = 1 TO MVC
      P(I) = 0
      FOR J = 1 TO NE
      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)
NEXT J
NEXT I
FOR I = 1 TO MVC
PA(I) = P(I) / MM(I)
NEXT I
PRINT #2,
PRINT #2, "H. PARTISIPASI SETIAP MODE"
PRINT #2,
FOR I = 1 TO MVC
PRINT #2, "   PA("; I; ")="; PA(I)
NEXT I
PRINT #2,
FOR I = 1 TO NE
DRM(I) = (W(1) * DR / W(I)) / 100
R(I) = WKW(I) - (2 / (DT ^ 2))
Z(I) = (1 / DT ^ 2) - (2 * DRM(I) * W(I) / (2 * DT))
F(I) = (1 / DT ^ 2) + (2 * DRM(I) * W(I) / (2 * DT))
NEXT I
PRINT #2, "I. KOEFISIEN I, a, b, k, Damping Rasio"
PRINT #2,
PRINT #2, "    Lapis     a      b      k       Damping Rasio"
FOR I = 1 TO NE
PRINT #2, USING "  ##"; I; TAB(8);
PRINT #2, USING "  ########.####"; R(I); Z(I); F(I); DRM(I)
NEXT I
PRINT #2,

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)
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 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
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
ERASE YD

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
ERASE GD
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
PRINT #2,
PRINT #2,
PRINT #2, "K. 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
ERASE YV

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 GA(NJ, NE)
FOR I = 1 TO NJ - 1
FOR J = 1 TO NE
GA(I, J) = (GD(I + 1,J) - (2 * GD(I,J)) + GD(I - 1, J)) / (DT ^ 2)
NEXT J
NEXT I
ERASE GD
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,
PRINT #2, "L. 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
```

LAMPIRAN II

INPUT DATA KOLOM UKURAN 60/60 CM AKIBAT GEMPA KOYNA

5	4	2100000	2160	5
25	25	25	25	20
380	380	380	380	380
60	60	60	60	60
60	60	60	60	60
10				
0	0			
0.01	-1.96			
0.02	-3.92			
0.03	-5.88			
0.04	-7.84			
0.05	-9.8			
0.06	-11.76			
0.07	-13.72			
0.08	-15.68			
0.09	-17.64			
0.1	-19.6			
0.11	-17.64			
0.12	-15.68			
0.13	-13.72			
0.14	-11.76			
0.15	-9.8			
0.16	-7.84			
0.17	-5.88			
0.18	-3.92			
0.19	-1.96			
0.2	0			
0.21	-4.9			
0.22	-9.8			
0.23	-14.7			
0.24	-19.6			
0.25	-24.5			
0.26	-29.4			
0.27	-34.3			
0.28	-39.2			
0.29	-44.1			
0.3	-49			
0.31	-39.2			
0.32	-29.4			
0.33	-19.6			
0.34	-9.8			
0.35	0			
0.36	9.8			
0.37	19.6			
0.38	29.4			
0.39	39.2			
0.4	49			
0.41	24.5			

LAMPIRAN III

INPUT DATA KOLOM UKURAN 60/60 CM AKIBAT GEMPA BUCHAREST

```
5       4       2100000 2160    5
25      25      25      25      20
380     380     380     380     380
60      60      60      60      60
60      60      60      60      60
10.9
0       0
0.01    1.372
0.02    2.744
0.03    4.116
0.04    5.488
0.05    6.86
0.06    8.232
0.07    9.604
0.08    10.976
0.09    12.348
0.1     13.72
0.11    15.092
0.12    16.464
0.13    17.836
0.14    19.208
0.15    20.58
0.16    21.952
0.17    23.324
0.18    24.696
0.19    26.068
0.2     27.44
0.21    28.812
0.22    30.184
0.23    31.556
0.24    32.928
0.25    34.3
0.26    32.1563
0.27    30.0125
0.28    27.8688
0.29    25.725
0.3     23.5813
0.31    21.4375
0.32    19.2938
0.33    17.15
0.34    15.0062
0.35    12.8625
0.36    10.7188
0.37    8.575
0.38    6.4313
0.39    4.2875
0.4     2.1437
0.41    0
0.42    -2.5789
```

dan seterusnya sampai dengan selesainya data gempa.