

```

REM *****
REM
REDIM FVERT(T + 1) AS DOUBLE, Q(T + 1) AS DOUBLE
REDIM INERZ(T + 1) AS DOUBLE, LOCSTIF(T + 1) AS DOUBLE
REDIM S1(T - 2) AS DOUBLE, S2(T - 2) AS DOUBLE, INCL(T + 1) AS DOUBLE
REDIM INDK(T - 2) AS DOUBLE, K(TT) AS DOUBLE
REDIM B(TT) AS DOUBLE, ML(TT) AS DOUBLE, MC(T + 1) AS DOUBLE
REDIM EIG(T - 2) AS DOUBLE, VECT(T - 2, NVEC) AS DOUBLE
REDIM SPOS(T + 1, T - 2) AS DOUBLE, SBAN(T + 1, T - 2) AS DOUBLE
REDIM ERRO(NVEC), USUB(T - 2, NVEC), VSUB(T - 2, NVEC)
REM -----
REM CHIAMA LA SUBROUTINE PER L'INGRESSO DEI DATI IN ARRAY
REM -----
CALL IngressoDatiArray
REM -----
REM CHIAMA LA SUBROUTINE PER RIEMPIRE ALCUNI ARRAY
REM -----
CALL Geometria
REM -----
REM CHIAMA LA SUBROUTINE PER STAMPARE I DATI DI INGRESSO
REM -----
TIME2 = TIMER - TIME1
CALL UscitaDati
TIME1 = TIMER
REM -----
REM CHIAMA LA SUBROUTINE PER RIEMPIRE LA MATRICE
REM DELLE RIGIDENZE LOCALI DELLE CELLE
REM -----
CALL RigidezzeLocali
REM -----
REM CHIAMA LA SUBROUTINE PER ASSEMBLARE LA MATRICE
REM DI RIGIDENZA GLOBALE
REM -----
CALL RigidezzeGlobali
REM -----
REM CHIAMA LA SUBROUTINE PER CALCOLARE LA MATRICE
REM DI RIGIDENZA GEOMETRICA
REM -----
CALL RigidezzeGeometriche
REM -----
REM CHIAMA LA SUBROUTINE PER CALCOLARE LA MATRICE
REM RIDOTTA DELLE RIGIDENZE

```

```

REM -----
CALL RigidezzeRidotte
REM -----
REM CHIAMA LA SUBROUTINE PER CALCOLARE LA MATRICE
REM DELLE MASSE
REM -----
CALL MatriceDelleMasse
REM -----
REM CHIAMA LA SUBROUTINE PER IL CALCOLO DEGLI AUTOVALORI
REM -----
CALL sivib2(ML(), INDK(), K(), INDK(), EIG(), ERRO(), USUB(), VSUB(),
VECT(), TT, TT, T - 2, NVEC, NEIG, NOI, TOLVEC)
REM -----
REM CHIAMA LA SUBROUTINE PER IL CALCOLO DEGLI SPOSTAMENTI
REM -----
CALL AutoSpostamenti
REM -----
REM CHIAMA LA SUBROUTINE PER LA STAMPA DEI RISULTATI
REM -----
TIME2 = TIME2 + TIMER - TIME1
CALL UscitaRisultati
REM -----
REM CHIUDE TUTTI I BUFFER TEMPORANEI
REM -----
CLOSE
END

SUB IngressoDati
REM
REM *****
REM *
REM *           I n g r e s s o D a t i
REM *
REM *****
REM
REM *****
REM *
REM *           Questa subroutine si occupa dell'ingresso dei dati
REM *           non organizzati in array
REM *
REM *****
REM *****

```

```

REM *
REM * LEGGE IL TITOLO DELLA STRUTTURA *
REM * IL FLAG DI STAMPA PER LE VARIABILI DI INGRESSO *
REM * IL FLAG DI STAMPA PER LE VARIABILI DI USCITA *
REM * - CARTA → I DATI VENGONO STAMPATI *
REM * - VIDEO → I DATI VENGONO INVIATI A VIDEO *
REM * - FILES → I DATI VENGONO MEMORIZZATI SU *
REM * FILES DA SPECIFICARE *
REM * - VIDEOFILES → I DATI VENGONO INVIATI A *
REM * VIDEO E MEMORIZZATI SU FILES *
REM * - CARTAFILES → I DATI VENGONO STAMPATI *
REM * CARTA E MEMORIZZATI SU FILES *
REM * - TUTTO → I DATI VENGONO INVIATI A VIDEO, *
REM * STAMPATI SU CARTA E MEMORIZZATI SU FILES *
REM *
REM *****
REM
INPUT #7, TITLE$
INPUT #7, STAMPA$
INPUT #7, STAMPARIS$
IF STAMPA$ = "FILES" OR STAMPA$ = "VIDEOFILES" OR STAMPA$ = "CAR-
TATAFILES" OR STAMPA$ = "TUTTO" THEN
  INPUT #7, FILEINGR$
END IF
IF STAMPARIS$ = "FILES" OR STAMPARIS$ = "VIDEOFILES" OR STAM-
PARIS$ = "CARTATAFILES" OR STAMPARIS$ = "TUTTO" THEN
  INPUT #7, FILEUSC$
END IF
CALL OpenFiles
REM
REM *****
REM *
REM * LEGGE: *
REM * LUCE Luce dell'arco *
REM * T Numero di coordinate lagrangiane *
REM * E Modulo di Young *
REM * CS Cedibilità angolare a sinistra *
REM * CD Cedibilità angolare a destra *
REM * MULT Moltiplicatore del peso proprio *
REM * NEIG Numero di autovalori da calcolare *
REM * NVEC Numero di vettori per condurre *
REM * l'iterazione sul sottospazio (NVEC > NEIG) *

```

```

REM *      NOI      Numero massimo di iterazioni                *
REM *      TOLVEC   Errore ammesso sugli autovettori           *
REM *      ASSE$    Se l'asse dell'arco è descritto da una     *
REM *              funzione, ASSE$ = FUNZ, se invece è dato   *
REM *              per punti, allora ASSE$ = DATI             *
REM *      VINCOLO$ Può assumere i seguenti valori:            *
REM *              INCASTRO                                     *
REM *              1 CERNIERA                                   *
REM *              2 CERNIERE                                   *
REM *              3 CERNIERE                                   *
REM *      SEZIONE$ Se la variazione del momento di inerzia   *
REM *              della sezione retta è data per punti,    *
REM *              allora SEZIONE$ = DATI. Altrimenti sono   *
REM *              disponibili le seguenti opzioni:            *
REM *              = PARABOLA se le inerzie variano           *
REM *              con legge parabolica                        *
REM *              = COSENOW se le inerzie variano            *
REM *              con la legge del coseno                     *
REM *      CARICHI$ Se i carichi sull'arco sono descritti da  *
REM *              funzione, CARICHI$ = FUNZ, se invece       *
REM *              sono dati per punti, allora CARICHI$ = DATI *
REM *      MASSE$   VERTICALE se le masse sono spalmate      *
REM *              lungo l'impalcato                          *
REM *              ARCO se le masse sono spalmate             *
REM *              lungo l'asse                                 *
REM *      Se l'asse è definito da funzione:                   *
REM *      FREC     Freccia dell'arco                          *
REM *      RIAL     Rialzo a destra                             *
REM *      Se l'inerzia della sezione varia secondo PARABOLA: *
REM *      INERZS   Momento di inerzia a sinistra              *
REM *      INERZC   Momento di inerzia al centro                *
REM *      INERZD   Momento di inerzia a destra                *
REM *      Se l'inerzia della sezione varia secondo COSENOW: *
REM *      W        Esponente del coseno                       *
REM *      INERZC   Momento di inerzia al centro                *
REM *      Se i carichi sono descritti da funzione:           *
REM *      QUNIF   Carico verticale uniforme                   *
REM *
REM *****
REM
INPUT #7, LUCE, T, E, CS, CD, MULT, NEIG, NVEC, NOI, TOLVEC
INPUT #7, ASSE$, VINCOLO$, SEZIONE$, CARICHI$, MASSE$

```

```

IF ASSE$ = "FUNC" THEN
  INPUT #7, FREQ, RIAL
END IF
SELECT CASE VINCOLO$
CASE "INCASTRO"
  CERN = 0
CASE "1 CERNIERA"
  CERN = 1
CASE "2 CERNIERE"
  CERN = 2
CASE "3 CERNIERE"
  CERN = 3
CASE ELSE
  CLS
  PRINT "OPZIONE NON VALIDA PER LA STRINGA VINCOLO$"
END SELECT
FOR I = 1 TO CERN
  INPUT #7, DIVCERN(I)
NEXT I
IF SEZIONE$ = "PARABOLA" THEN INPUT #7, INERZS, INERZC, INERZD
IF SEZIONE$ = "COSENOW" THEN INPUT #7, W, INERZC
IF CARICHI$ = "FUNC" THEN
  INPUT #7, QUNIF
END IF
END SUB

```

```

SUB MatriceDelleMasse
REM
REM *****
REM *
REM *           M a t r i c e D e l l e M a s s e
REM *
REM *****
REM
REM *****
REM *
REM *           C A L C O L A   L A   M A T R I C E   G L O B A L E   D E L L E   M A S S E
REM *
REM *****
REM
REM COSTRUZIONE DELL'ARRAY DELLE MASSE, CALCOLATE
REM COME RAPPORTI TRA PESO PROPRIO E GRAVITA',

```

```

REM OPPURE COME MASSE DISTRIBUITE LUNGO L'ARCO
REM _____
REM
IF MASSE$ = "VERTICALE" THEN
  FOR I = 1 TO T + 1
    MC(I) = FVERT(I) / GRAV
  NEXT I
ELSE
  FOR I = 1 TO T + 1
    MC(I) = FVERT(I) / GRAV / COS(INCL(I))
  NEXT I
END IF
REM _____
REM ASSEMBLAGGIO DELLA MATRICE LAGRANGIANA DELLE MASSE
REM SECONDO LA FORMULA 7.12
REM _____
FOR I = 1 TO T - 2
  FOR J = I TO T - 2
    IK = J * (J + 1) / 2 - J + I
    M = 0
    FOR K = J + 1 TO T - 1
      M = M + MC(K)
    NEXT K
    ML(IK) = LTRAT ^ 2 * (M + (1 + S1(I)) * (1 + S1(J)) * MC(T))
    ML(IK) = ML(IK) + M * (Q(I + 1) - Q(I)) * (Q(J + 1) - Q(J))
    MAUX = (Q(I + 1) - Q(I) + (Q(T) - Q(T - 1)) * S1(I))
    MAUX1 = (Q(J + 1) - Q(J) + (Q(T) - Q(T - 1)) * S1(J))
    ML(IK) = ML(IK) + MAUX * MAUX1 * MC(T)
  NEXT J
NEXT I
END SUB

SUB RigidezzeRidotte
REM
REM *****
REM *
REM *           R i g i d e z z e R i d o t t e
REM *
REM *****
REM
REM *****
REM *

```

```

REM *          CALCOLA LA MATRICE RIDOTTA DELLE RIGIDENZE          *
REM *                                                                 *
REM *****
REM
FOR IK = 1 TO TT
  K(IK) = K(IK) - MULT * B(IK)
NEXT IK
END SUB

SUB UscitaDati
REM
REM *****
REM *                                                                 *
REM *          U s c i t a D a t i          *
REM *                                                                 *
REM *****
REM
REM *****
REM *                                                                 *
REM *          Questa subroutine si occupa dell'uscita (su video, su stampa *
REM *          e su file) dei dati di ingresso *
REM *                                                                 *
REM *****
REM
FOR ITER = 1 TO ITER1
  PRINT #ITER, " ===== "
  PRINT #ITER, "PROGRAMMA ADIN1 - ANALISI DINAMICA
  PRINT #ITER, TITLE$
  PRINT #ITER, DI UN ARCO DA PONTE"
  PRINT #ITER, " ===== "
  PRINT #ITER,
  IF CERN = 0 THEN PRINT #ITER, "ARCO INCASTRATO"
  IF CERN = 1 THEN PRINT #ITER, "ARCO AD 1 CERNIERA"
  IF CERN = 2 THEN PRINT #ITER, "ARCO A 2 CERNIERE"
  IF CERN = 3 THEN PRINT #ITER, "ARCO A 3 CERNIERE"
  PRINT #ITER, "FILE DEI DATI ="; FILEINPUT$
  PRINT #ITER, "LUCE DELL'ARCO = "; LUCE
  PRINT #ITER, "FRECCIA DELL'ARCO = "; FREC
  PRINT #ITER,
  PRINT #ITER, "FILE DEI DATI ="; FILEINPUT$
  PRINT #ITER, "LUCE DELL'ARCO = "; LUCE
  PRINT #ITER, "FRECCIA DELL'ARCO = "; FREC

```

```

PRINT #ITER, "RIALZO A DESTRA ="; RIAL
PRINT #ITER, "MODULO DI YOUNG ="; E
PRINT #ITER, "CEDIBILITA' ANGOLARE A SINISTRA = "; CS
PRINT #ITER, "CEDIBILITA' ANGOLARE A DESTRA = "; CD
PRINT #ITER, "MOLTIPLICATORE DEL PESO PROPRIO = "; MULT
PRINT #ITER, "NUMERO DI AUTOVALORI DA CALCOLARE = "; NEIG
PRINT #ITER, "ERRORE AMMESSO SUGLI AUTOVETTORI = "; TOLVEC
PRINT #ITER, "DIMENSIONE DEL SOTTOSPAZIO = "; NVEC
PRINT #ITER, "NUMERO MASSIMO DI ITERAZIONI = "; NOI
PRINT #ITER, "NUMERO DI TRATTI = "; T
IF CERN >= 1 THEN
  PRINT #ITER, "PRIMA CERNIERA ALLA DIVIDENTE "; DIVCERN(1)
END IF
IF CERN >= 2 THEN
  PRINT #ITER, "SECONDA CERNIERA ALLA DIVIDENTE "; DIVCERN(2)
END IF
IF CERN = 3 THEN
  PRINT #ITER, "TERZA CERNIERA ALLA DIVIDENTE "; DIVCERN(3)
END IF
IF SEZIONE$ = "PARABOLA" THEN
  PRINT #ITER, "INERZIA DELLA SEZIONE RETTA DELL'ARCO A SINIS-
    TRA = "; INERZS
  PRINT #ITER, "INERZIA DELLA SEZIONE RETTA DELL'ARCO AL CEN-
    TRO = "; INERZC
  PRINT #ITER, "INERZIA DELLA SEZIONE RETTA DELL'ARCO A DE-
    STRA = "; INERZD
END IF
IF SEZIONE$ = "COSENOW" THEN
  PRINT #ITER, "INERZIA DELLA SEZIONE RETTA DELL'ARCO AL CEN-
    TRO = "; INERZC
END IF
PRINT #ITER,
IF CARICHI$ = "FUNC" THEN
  PRINT #ITER, "CARICO VERTICALE UNIFORME = "; QUNIF
END IF
IF MASSE$ = "VERTICALE" THEN
  PRINT #ITER, "MASSE DISTRIBUITE SULL'IMPALCATO"
ELSE
  PRINT #ITER, "MASSE DISTRIBUITE LUNGO L'ARCO"
END IF
REM -----
REM STAMPA LE QUOTE DELL'ARCO PER CIASCUNA DIVIDENTE

```



```

REM -----
IF ASSE$ = "DATI" THEN
PRINT #ITER,
  PRINT #ITER, " ===== "
  PRINT #ITER, " = "
  PRINT #ITER, " =          TABELLA DELLE QUOTE DELL'ARCO          = "
  PRINT #ITER, " = "
  PRINT #ITER, " ===== "
  PRINT #ITER,
  PRINT #ITER, " ----- "
  PRINT #ITER, " DIVIDENTE          QUOTA          "
  PRINT #ITER, " ----- "
  PRINT #ITER,
  FOR I = 1 TO T + 1
    PRINT #ITER, TAB(1); I; TAB(34); Q(I)
  NEXT I
END IF
IF CARICHI$ = "DATI" THEN
REM -----
REM STAMPA I CARICHI PER CIASCUNA DIVIDENTE
REM -----
PRINT #ITER, "SPINTA ORIZZONTALE = "; HSPINTA
PRINT #ITER,
  PRINT #ITER, " ===== "
  PRINT #ITER, " = "
  PRINT #ITER, " =          CARICHI VERTICALI SULL'ARCO          = "
  PRINT #ITER, " = "
  PRINT #ITER, " ===== "
  PRINT #ITER,
  PRINT #ITER, " ----- "
  PRINT #ITER, " DIVIDENTE          CARICO VERTICALE          "
  PRINT #ITER, " ----- "
  PRINT #ITER,
  FOR I = 1 TO T + 1
    PRINT #ITER, TAB(2); I; TAB(20); FVERT(I)
  NEXT I
END IF
IF SEZIONE$ = "DATI" THEN
REM -----
REM STAMPA I MOMENTI DI INERZIA PER CIASCUNA DIVIDENTE
REM -----
PRINT #ITER, " ===== "

```

```

PRINT #ITER, " =                                     ="
PRINT #ITER, " = MOMENTI DI INERZIA DELLA SEZIONE RETTA ="
PRINT #ITER, " =                                     ="
PRINT #ITER, " ====="
PRINT #ITER,
PRINT #ITER, " _____"
PRINT #ITER, " DIVIDENTE          MOMENTO DI INERZIA      "
PRINT #ITER, " _____"
PRINT #ITER,
FOR I = 1 TO T + 1
  PRINT #ITER, TAB(2); I; TAB(20); INERZ(I)
NEXT I
END IF
NEXT ITER
END SUB

SUB UscitaRisultati
REM
REM *****
REM *
REM *          U s c i t a R i s u l t a t i
REM *
REM *****
REM
REM *****
REM *
REM *          Questa subroutine si occupa dell'uscita (su video, su stampa
REM *          e su file) dei risultati
REM *
REM *****
PI = 4 * ATN(1)
FOR ITER = ITER1 + 1 TO ITER2
  PRINT #ITER, "          FREQUENZE CIRCOLARI    PERIODO"
  PRINT #ITER,
  FOR I = 1 TO NEIG
    PRINT #ITER, TAB(2); I; TAB(7); EIG(I); TAB(30); 2 * PI / SQR(EIG(I))
  NEXT I
  PRINT #ITER,
  FOR I = 1 TO NEIG
    PRINT #ITER,
    PRINT #ITER, "AUTOVETTORE N. "; I
    PRINT #ITER,

```

```
    FOR K = 1 TO T - 2
      PRINT #ITER, TAB(2); K; TAB(15); VECT(K, I)
    NEXT K
  NEXT I
  PRINT #ITER,
  FOR I = 1 TO NEIG
    PRINT #ITER,
    PRINT #ITER, "SPOSTAMENTI VERTICALI CONNESSI ALL'AUTOVET-
      TORE N. "; I
    PRINT #ITER,
    FOR K = 1 TO T + 1
      PRINT #ITER, TAB(2); K; TAB(15); SPOS(K, I)
    NEXT K
  NEXT I
  PRINT #ITER,
  PRINT #ITER,
  FOR I = 1 TO NEIG
    PRINT #ITER,
    PRINT #ITER, "SPOSTAMENTI ORIZZONTALI CONNESSI ALL'AUTO-
      VETTORE N. "; I
    PRINT #ITER,
    FOR K = 1 TO T + 1
      PRINT #ITER, TAB(2); K; TAB(15); SBAN(K, I)
    NEXT K
  NEXT I
  PRINT #ITER,
  PRINT #ITER, "TEMPO DI ESECUZIONE = "; TIME2; " SEC."
  PRINT #ITER, "RUN DEL "; DATE$; " ORE "; TIME$
NEXT ITER
END SUB
```

Appendice 7.2. Il programma ADIN2

```
DEFDBL A-Z
DECLARE SUB mult11rc (A#(), B#(), RA#, CA#, C#())
DECLARE SUB mult1cc (A(), B(), RA, CA, C())
DECLARE SUB mult1rc (A(), B(), RA, CA, C())
DECLARE SUB dot (A(), B(), C, N)
DECLARE SUB Trasferimento ()
DECLARE SUB ForzeSismiche ()
DECLARE SUB RigidezzeRidotte ()
DECLARE SUB MatriceDelleMasse ()
DECLARE SUB sivib2 (G(), GD(), L(), LD(), BD(), ERRO(), U(), V(), W(), NA,
    NK, N, M, NRQD, NOI, TOLVEC)
DECLARE SUB errore (U(), W(), BD(), ERRO(), N, M, NRQD, LCK, TOLVEC,
    LT)
DECLARE SUB decouple (U(), V(), W(), BD(), N, M, LCK, TOLVEC)
DECLARE SUB orthog (W(), N, M, LCK)
DECLARE SUB premult (U(), V(), L(), LD(), NA, N, M, LCK, IRENT)
DECLARE SUB rand (U(), N, M, IRENT)
DECLARE SUB reduce (L(), LD(), NK, N)
DECLARE SUB forsub (U(), V(), L(), LD(), NK, N, M, LCK)
DECLARE SUB backsub (U(), V(), L(), LD(), NK, N, M, LCK)
DECLARE SUB RigidezzeGeometriche ()
DECLARE SUB RigidezzeGlobali ()
DECLARE SUB RigidezzeLocali ()
DECLARE SUB Geometria ()
DECLARE SUB UscitaDati ()
DECLARE SUB UscitaRisultati ()
DECLARE SUB IngressoDati ()
DECLARE SUB IngressoDatiArray ()
DECLARE SUB OpenFiles ()
REM *****
REM *
REM *          P R O G R A M M A      A D I N 2          *
```

```

REM *
REM *
REM *
REM *          V I N C E N Z O   F R A N C I O S I
REM *
REM *          Release 1.0.0. 3 Agosto 1987 (in BASIC HP)
REM *          Release 3.0.0. Ottobre 1994 (in Microsoft QBasic)
REM *
REM *****
REM
REM
REM *****
REM *
REM *          Questo programma esegue il calcolo delle forze sismiche agenti
REM *          su di un arco, a causa di uno scuotimento sincrono
REM *
REM *****
OPTION BASE 1
REM
REM *****
REM *
REM *          C O M M O N
REM *
REM *****
REM
COMMON SHARED LUCE, T, E, CS, CD, FREC, RIAL
COMMON SHARED INERZS, INERZC, INERZD, WEXP, QUNIF, HSPINTA
COMMON SHARED LTRAT, CERN, GRAV, MULT, TT, AMPL, T0
COMMON SHARED NVEC, NEIG, NOI, TOLVEC
COMMON SHARED ITER1, ITER2, TIME2
COMMON SHARED FVERT() AS DOUBLE, Q() AS DOUBLE
COMMON SHARED INERZ() AS DOUBLE, INCL() AS DOUBLE
COMMON SHARED LOCSTIF() AS DOUBLE, DIVCERN() AS DOUBLE
COMMON SHARED S1() AS DOUBLE, S2() AS DOUBLE, S3() AS DOUBLE
COMMON SHARED K() AS DOUBLE, B() AS DOUBLE, INDK() AS DOUBLE
COMMON SHARED EIG() AS DOUBLE, VECT() AS DOUBLE
COMMON SHARED VT() AS DOUBLE, WT() AS DOUBLE
COMMON SHARED SV() AS DOUBLE, SW() AS DOUBLE
COMMON SHARED MC() AS DOUBLE, ML() AS DOUBLE
COMMON SHARED PART() AS DOUBLE, MOM() AS DOUBLE
COMMON SHARED SISV() AS DOUBLE, SISO() AS DOUBLE
COMMON SHARED TITLE$, STAMPA$, STAMPARIS$, FILEINGR$, FILEUSC$

```

```

COMMON SHARED ASSE$, VINCOLO$, SEZIONE$, CARICHI$, FILEINPUT$
COMMON SHARED SCUOTIMENTO$, MASSE$
REM
REM *****
REM *
REM *          DIMENSIONA GLI ARRAY STATICI          *
REM *
REM *****
REM
DIM TITLE$(100), TITLE1$(100), STAMPA$(10), STAMPARIS$(10)
DIM FILEINGR$(10), FILEUSC$(10), DIVCERN(3) AS DOUBLE
REM
REM *****
REM *
REM *          DEFINISCE ALCUNE LEGGI DI VARIAZIONE          *
REM *
REM *****
DEF FNQ (Z) = RIAL * Z / LUCE + 4 * FREC / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNINERZ1 (Z)
  FAUX = INERZS * (LUCE - Z) / LUCE + INERZD * Z / LUCE
  FAUX = FAUX - 2 * (INERZS + INERZD - 2 * INERZC) / LUCE ^ 2 * Z *
    (LUCE - Z)
  FNINERZ1 = FAUX
END DEF
DEF FNA (Z) = 4 * FREC / LUCE ^ 2 * (LUCE - 2 * Z) + RIAL / LUCE
DEF FNINERZ2 (Z) = INERZC * COS(ATN(FNA(Z))) ^ WEXP
CLS
REM
REM *****
REM *
REM *          LEGGE IL NOME DEL FILE DI INPUT DATI          *
REM *
REM *****
REM
PRINT "NOME DEL FILE DI INPUT"
INPUT FILEINPUT$
TIME1 = TIMER
OPEN FILEINPUT$ FOR INPUT AS #7
REM -----
REM CHIAMA LA SUBROUTINE PER L'INGRESSO DEI DATI
REM -----
CALL IngressoDati

```

```

TT = (T - 2) * (T - 1) / 2
LTRAT = LUCE / T
GRAV = 9.81#
REM
REM *****
REM *
REM *          REDIMENSIONA GLI ARRAY DINAMICI          *
REM *
REM *****
REDIM FVERT(T + 1) AS DOUBLE, Q(T + 1) AS DOUBLE
REDIM INERZ(T + 1) AS DOUBLE, LOCSTIF(T + 1) AS DOUBLE
REDIM S1(T - 2) AS DOUBLE, S2(T - 2) AS DOUBLE, S3(T - 2) AS DOUBLE
REDIM INDK(T - 2) AS DOUBLE, K(TT) AS DOUBLE, INCL(T + 1) AS DOUBLE
REDIM B(TT) AS DOUBLE, ML(TT) AS DOUBLE, MC(T + 1) AS DOUBLE
REDIM EIG(T - 2) AS DOUBLE, VECT(T - 2, NVEC) AS DOUBLE
REDIM VT(T + 1, T - 2) AS DOUBLE, WT(T + 1, T - 2) AS DOUBLE
REDIM PART(NEIG) AS DOUBLE, MOM(T + 1) AS DOUBLE
REDIM SISV(T + 1) AS DOUBLE, SISO(T + 1) AS DOUBLE
REDIM SV(T + 1) AS DOUBLE, SW(T + 1) AS DOUBLE
REDIM ERRO(NVEC) AS DOUBLE, USUB(T - 2, NVEC) AS DOUBLE
REDIM VSUB(T - 2, NVEC) AS DOUBLE
REM -----
REM CHIAMA LA SUBROUTINE PER L'INGRESSO DEI DATI IN ARRAY
REM -----
CALL IngressoDatiArray
REM -----
REM CHIAMA LA SUBROUTINE PER RIEMPIRE ALCUNI ARRAY
REM -----
CALL Geometria
REM -----
REM CHIAMA LA SUBROUTINE PER STAMPARE I DATI DI INGRESSO
REM -----
TIME2 = TIMER - TIME1
CALL UscitaDati
TIME1 = TIMER
REM -----
REM CHIAMA LA SUBROUTINE PER RIEMPIRE LA MATRICE
REM DELLE RIGIDENZE LOCALI DELLE CELLE
REM -----
CALL RigidezzeLocali
REM -----
REM CHIAMA LA SUBROUTINE PER ASSEMBLARE LA MATRICE

```

```

REM DI RIGIDEZZA GLOBALE
REM -----
CALL RigidezzeGlobali
REM -----
REM CHIAMA LA SUBROUTINE PER CALCOLARE LA MATRICE
REM DI RIGIDEZZA GEOMETRICA
REM -----
CALL RigidezzeGeometriche
REM -----
REM CHIAMA LA SUBROUTINE PER CALCOLARE LA MATRICE
REM RIDOTTA DELLE RIGIDEZZE
REM -----
CALL RigidezzeRidotte
REM -----
REM CHIAMA LA SUBROUTINE PER CALCOLARE LA MATRICE
REM GLOBALE DELLE MASSE
REM -----
CALL MatriceDelleMasse
REM -----
REM CHIAMA LA SUBROUTINE PER IL CALCOLO DEGLI AUTOVALORI
REM -----
CALL sivib2(ML(), INDK(), K(), INDK(), EIG(), ERRO(), USUB(), VSUB(),
    VECT(), TT, TT, T - 2, NVEC, NEIG, NOI, TOLVEC)
REM -----
REM CHIAMA LA SUBROUTINE PER IL CALCOLO
REM DELLE MATRICI DI TRASFERIMENTO
REM -----
CALL Trasferimento
REM -----
REM CHIAMA LA SUBROUTINE PER IL CALCOLO DELLEFORZE SISMICHE
REM -----
CALL ForzeSismiche
REM -----
REM CHIAMA LA SUBROUTINE PER LA STAMPA DEI RISULTATI
REM -----
TIME2 = TIME2 + TIMER - TIME1
CALL UscitaRisultati
REM -----
REM CHIUDE TUTTI I BUFFER TEMPORANEI
REM -----
CLOSE
END

```



```

SUB ForzeSismiche REM
REM *****
REM *
REM *           F o r z e S i s m i c h e
REM *
REM *****
REM
REM *****
REM *
REM * QUESTA SUBROUTINE CALCOLA LE FORZE SISMICHE AGENTI
REM * SU DI UN ARCO, A CAUSA DI UNO SCUOTIMENTO SISMICO
REM * DI TIPO SINCRONO, ORIZZONTALE O VERTICALE,
REM * TRAMITE SOVRAPPOSIZIONE MODALE DEI PRIMI
REM * NRQD AUTOVETTORI
REM *
REM *****
REM -----
REM Calcolo dei coefficienti di partecipazione
REM -----
IF SCUOTIMENTO$ = "ORIZZONTALE" THEN
  CALL mult1cc(WT(), MC(), T + 1, T - 2, S3())
ELSE
  CALL mult1cc(VT(), MC(), T + 1, T - 2, S3())
END IF
FOR I = 1 TO NEIG
  FOR J = 1 TO T - 2
    S1(J) = VECT(J, I)
  NEXT J
  CALL mult11rc(ML(), S1(), T - 2, T - 2, S2())
  CALL dot(S1(), S2(), D1, T - 2)
  CALL dot(S1(), S3(), N1, T - 2)
  PART(I) = N1 / D1
NEXT I
REM -----
REM Calcolo degli spostamenti
REM -----
PI = 4 * ATN(1)
AMPL = AMPL * (2 * PI / T0) ^ 2
OMFOR = 2 * PI / T0
FOR I = 1 TO T - 2
  D = 0
  FOR J = 1 TO NEIG

```

'cfr.7.32

```

      D = D - VECT(I, J) * PART(J) / (EIG(J) - OMFOR ^ 2)
    NEXT J
    S1(I) = D
  NEXT I
  CALL mult1rc(VT(), S1(), T + 1, T - 2, SV())
  CALL mult1rc(WT(), S1(), T + 1, T - 2, SW())
  FOR I = 1 TO T + 1
    SV(I) = AMPL * SV(I)
    SW(I) = AMPL * SW(I)
  NEXT I
  REM -----
  REM Calcolo dei momenti
  REM -----
  FOR I = 2 TO T
    MOM(I) = LOCSTIF(I) * (-SV(I + 1) + 2 * SV(I) - SV(I - 1)) / LTRAT
  NEXT I
  MOM(1) = -LOCSTIF(1) * SV(2) / LTRAT
  MOM(T + 1) = -LOCSTIF(T + 1) * SV(T) / LTRAT
  REM -----
  REM Calcolo delle forze orizzontali e verticali
  REM -----
  FOR I = 1 TO T - 2
    S1(I) = 0
    FOR J = 1 TO NEIG
      S1(I) = S1(I) - EIG(J) / (EIG(J) - OMFOR ^ 2) * PART(J) * VECT(I, J)
    NEXT J
  NEXT I
  CALL mult1rc(VT(), S1(), T + 1, T - 2, SISV())
  CALL mult1rc(WT(), S1(), T + 1, T - 2, SISO())
  FOR I = 1 TO T + 1
    SISV(I) = SISV(I) * MC(I) * AMPL           'cfr. 7.61
    SISO(I) = SISO(I) * MC(I) * AMPL           'cfr. 7.64
  NEXT I
  END SUB

SUB IngressoDati
REM
REM *****
REM *
REM *           I n g r e s s o D a t i
REM *
REM *****

```

```

REM
REM *****
REM *
REM *          Questa subroutine si occupa dell'ingresso dei dati          *
REM *          non organizzati in array                                     *
REM *
REM *****
REM *****
REM *
REM * LEGGE IL TITOLO DELLA STRUTTURA                                     *
REM * IL FLAG DI STAMPA PER LE VARIABILI DI INGRESSO                     *
REM * IL FLAG DI STAMPA PER LE VARIABILI DI USCITA                       *
REM *   - CARTA → I DATI VENGONO STAMPATI                                 *
REM *   - VIDEO → I DATI VENGONO INVIATI A VIDEO                         *
REM *   - FILES → I DATI VENGONO MEMORIZZATI SU                           *
REM *     FILES DA SPECIFICARE                                           *
REM *   - VIDEOFILES → I DATI VENGONO INVIATI A                          *
REM *     VIDEO E MEMORIZZATI SU FILES                                    *
REM *   - CARTAFILES → I DATI VENGONO STAMPATI                            *
REM *     CARTA E MEMORIZZATI SU FILES                                    *
REM *   - TUTTO → I DATI VENGONO INVIATI A VIDEO,                         *
REM *     STAMPATI SU CARTA E MEMORIZZATI SU FILES                       *
REM *
REM *****
REM
INPUT #7, TITLE$
INPUT #7, STAMPA$
INPUT #7, STAMPARIS$
IF STAMPA$ = "FILES" OR STAMPA$ = "VIDEOFILES" OR STAMPA$ = "CAR-
  TFILES" OR STAMPA$ = "TUTTO" THEN
  INPUT #7, FILEINGR$
END IF
IF STAMPARIS$ = "FILES" OR STAMPARIS$ = "VIDEOFILES" OR STAM-
  PARIS$ = "CARTFILES" OR STAMPARIS$ = "TUTTO" THEN
  INPUT #7, FILEUSC$
END IF
CALL OpenFiles
REM
REM *****
REM *
REM * LEGGE:
REM *   LUCE   Luce dell'arco

```

REM *	T	Numero di coordinate lagrangiane	*
REM *	E	Modulo di Young	*
REM *	CS	Cedibilità angolare a sinistra	*
REM *	CD	Cedibilità angolare a destra	*
REM *	MULT	Moltiplicatore del peso proprio	*
REM *	AMPL	Ampiezza dello scuotimento	*
REM *	T0	Periodo dello scuotimento	*
REM *	NEIG	Numero di autovalori da calcolare	*
REM *	NVEC	Numero di vettori per condurre l'iterazione sul sottospazio (NVEC > NEIG)	*
REM *	NOI	Numero massimo di iterazioni	*
REM *	TOLVEC	Errore ammesso sugli autovettori	*
REM *	ASSE\$	Se l'asse dell'arco è descritto da una funzione, ASSE\$ = FUNZ, se invece è dato per punti, allora ASSE\$ = DATI	*
REM *	VINCOLO\$	Può assumere i seguenti valori:	*
REM *		INCASTRO	*
REM *		1 CERNIERA	*
REM *		2 CERNIERE	*
REM *		3 CERNIERE	*
REM *	SEZIONE\$	Se la variazione del momento di inerzia della sezione retta è data per punti, allora SEZIONE\$ = DATI. Altrimenti sono disponibili le seguenti opzioni:	*
REM *		= PARABOLA se le inerzie variano con legge parabolica	*
REM *		= COSENOW se le inerzie variano con la legge del coseno	*
REM *	CARICHI\$	Se i carichi sull'arco sono descritti da funzione, CARICHI\$ = FUNZ, se invece sono dati per punti, allora CARICHI\$ = DATI	*
REM *	MASSE\$	VERTICALE se le masse sono spalmate lungo l'impalcato	*
REM *		ARCO se le masse sono spalmate lungo l'asse	*
REM *	SCUOTIMENTO\$	VERTICALE se il sismo è sussultorio	*
REM *		ORIZZONTALE se il sismo è ondulatorio	*
REM *		Se l'asse è definito da funzione:	*
REM *	FREC	Freccia dell'arco	*
REM *	RIAL	Rialzo a destra	*
REM *		Se l'inerzia della sezione varia secondo PARABOLA:	*
REM *	INERZS	Momento di inerzia a sinistra	*
REM *	INERZC	Momento di inerzia al centro	*

```

REM *      INERZD   Momento di inerzia a destra          *
REM *      Se l'inerzia della sezione varia secondo COSENOW:      *
REM *      WEXP     Esponente del coseno                  *
REM *      INERZC   Momento di inerzia al centro          *
REM *      Se i carichi sono descritti da funzione:              *
REM *      QUNIF    Carico verticale uniforme              *
REM *
REM *****
REM
INPUT #7, LUCE, T
INPUT #7, E, CS, CD, MULT, AMPL, T0, NEIG, NVEC, NOI, TOLVEC
INPUT #7, ASSE$, VINCOLO$, SEZIONE$, CARICHI$
INPUT #7, MASSE$, SCUOTIMENTO$
IF ASSE$ = "FUNC" THEN
  INPUT #7, FREC, RIAL
END IF
SELECT CASE VINCOLO$
CASE "INCASTRO"
  CERN = 0
CASE "1 CERNIERA"
  CERN = 1
CASE "2 CERNIERE"
  CERN = 2
CASE "3 CERNIERE"
  CERN = 3
CASE ELSE
  CLS
  PRINT "OPZIONE NON VALIDA PER LA STRINGA VINCOLO$"
END SELECT
FOR I = 1 TO CERN
  INPUT #7, DIVCERN(I)
NEXT I
IF SEZIONE$ = "PARABOLA" THEN INPUT #7, INERZS, INERZC, INERZD
IF SEZIONE$ = "COSENOW" THEN INPUT #7, WEXP, INERZC
IF CARICHI$ = "FUNC" THEN
  INPUT #7, QUNIF
END IF
END SUB

SUB mult11rc (A(), B(), RA, CA, C())
REM
REM *****

```

```

REM *
REM *
REM *
REM *****
REM *****
REM *****
REM *
REM * OPERA IL PRODOTTO MATRICIALE, RIGA PER COLONNA, TRA LA *
REM * MATRICE A, AD RA RIGHE E CA COLONNE, ED IL VETTORE B *
REM * AD RB RIGHE, OTTENENDO IL VETTORE C, A CA RIGHE *
REM * LA MATRICE A() E' MEMORIZZATA IN BANDA MOBILE *
REM * 1. IL PRODOTTO PUO' EFFETTUARSI SOLO SE CA = RB *
REM *
REM *****
FOR I = 1 TO CA
  C(I) = 0
  FOR K = 1 TO RA
    IK = K * (K + 1) / 2 - K + I
    C(I) = C(I) + A(IK) * B(K)
  NEXT K
NEXT I
END SUB

SUB multlcc (A(), B(), RA, CA, C())
REM *****
REM *
REM *
REM *
REM *****
REM *****
REM *****
REM *
REM * OPERA IL PRODOTTO MATRICIALE, COLONNA PER COLONNA, *
REM * TRA LA MATRICE A, AD RA RIGHE E CA COLONNE, *
REM * ED IL VETTORE B AD RB RIGHE, *
REM * OTTENENDO IL VETTORE C, A CA RIGHE *
REM * 1. IL PRODOTTO PUO' EFFETTUARSI SOLO SE RA = RB *
REM * 2. IL PRODOTTO EQUIVALE ALL'USUALE PRODOTTO *
REM * TRA LA TRASPOSTA DI A E B. *
REM *
REM *****

```

400 *Le strutture ad arco*

```
REM
FOR I = 1 TO CA
  C(I) = 0
  FOR K = 1 TO RA
    C(I) = C(I) + A(K, I) * B(K)
  NEXT K
NEXT I
END SUB
```

```
SUB mult1rc (A(), B(), RA, CA, C())
REM
REM *****
REM *
REM *           m u l t 1 r c
REM *
REM *****
REM
REM *****
REM *
REM * OPERA IL PRODOTTO MATRICIALE, RIGA PER COLONNA,
REM * TRA LA MATRICE A, AD RA RIGHE E CA COLONNE,
REM * ED IL VETTORE B AD RB RIGHE, OTTENENDO
REM * IL VETTORE C, A CA RIGHE
REM * 1. IL PRODOTTO PUO' EFFETTUARSI SOLO SE CA = RB
REM *
REM *****
FOR I = 1 TO RA
  C(I) = 0
  FOR K = 1 TO CA
    C(I) = C(I) + A(I, K) * B(K)
  NEXT K
NEXT I
END SUB
```

```
SUB Trasferimento
REM
REM *****
REM *
REM *           T r a s f e r i m e n t o
REM *
REM *****
REM
* REM
```

```

REM *****
REM *
REM *          COSTRUISCE LE MATRICI DI TRASFERIMENTO
REM *          PER GLI SPOSTAMENTI
REM *
REM *****
FOR I = 2 TO T - 1
  FOR J = 1 TO I - 1
    VT(I, J) = -LTRAT
    WT(I, J) = -(Q(J + 1) - Q(J))
  NEXT J
NEXT I
FOR J = 1 TO T - 2
  VT(T, J) = -LTRAT * (1 + S1(J))
  VT(T + 1, J) = -LTRAT * (1 + S1(J) + S2(J))
  WT(T, J) = -(Q(J + 1) - Q(J) + (Q(T) - Q(T - 1)) * S1(J))
  WT(T + 1, J) = WT(T, J) - (Q(T + 1) - Q(T)) * S2(J)
NEXT J
END SUB

SUB UscitaDati
REM *****
REM *
REM *          U s c i t a D a t i
REM *
REM *****
REM *****
REM *
REM *          Questa subroutine si occupa dell'uscita (su video, su stampa
REM *          e su file) dei dati di ingresso
REM *
REM *****
REM *****
FOR ITER = 1 TO ITER1
  PRINT #ITER, " ===== "
  PRINT #ITER, "PROGRAMMA ADIN2 - CALCOLO DELLE FORZE
  PRINT #ITER, TITLE$
  PRINT #ITER, SISMICHE PER SCUOTIMENTO "; SCUOTIMENTO$
  PRINT #ITER, " ===== "
  PRINT #ITER,

```



```

IF CERN = 0 THEN PRINT #ITER, "ARCO INCASTRATO"
IF CERN = 1 THEN PRINT #ITER, "ARCO AD 1 CERNIERA"
IF CERN = 2 THEN PRINT #ITER, "ARCO A 2 CERNIERE"
IF CERN = 3 THEN PRINT #ITER, "ARCO A 3 CERNIERE"
PRINT #ITER, "FILE DEI DATI ="; FILEINPUT$
PRINT #ITER, "LUCE DELL'ARCO = "; LUCE
PRINT #ITER, "FRECCIA DELL'ARCO = "; FREC
PRINT #ITER,
PRINT #ITER, "FILE DEI DATI ="; FILEINPUT$
PRINT #ITER, "LUCE DELL'ARCO = "; LUCE
PRINT #ITER, "FRECCIA DELL'ARCO = "; FREC
PRINT #ITER, "RIALZO A DESTRA ="; RIAL
PRINT #ITER, "MODULO DI YOUNG ="; E
PRINT #ITER, "CEDIBILITA' ANGOLARE A SINISTRA = "; CS
PRINT #ITER, "CEDIBILITA' ANGOLARE A DESTRA = "; CD
PRINT #ITER, "MOLTIPLICATORE DEL PESO PROPRIO = "; MULT
PRINT #ITER, "AMPIEZZA DELLO SCUOTIMENTO = "; AMPL
PRINT #ITER, "PERIODO DELLO SCUOTIMENTO = "; T0
PRINT #ITER, "NUMERO DI AUTOVALORI DA CALCOLARE = "; NEIG
PRINT #ITER, "ERRORE AMMESSO SUGLI AUTOVETTORI = "; TOLVEC
PRINT #ITER, "DIMENSIONE DEL SOTTOSPAZIO = "; NVEC
PRINT #ITER, "NUMERO MASSIMO DI ITERAZIONI = "; NOI
PRINT #ITER, "NUMERO DI TRATTI = "; T
IF CERN >= 1 THEN
  PRINT #ITER, "PRIMA CERNIERA ALLA DIVIDENTE "; DIVCERN(1)
END IF
IF CERN >= 2 THEN
  PRINT #ITER, "SECONDA CERNIERA ALLA DIVIDENTE "; DIVCERN(2)
END IF
IF CERN = 3 THEN
  PRINT #ITER, "TERZA CERNIERA ALLA DIVIDENTE "; DIVCERN(3)
END IF
IF SEZIONE$ = "PARABOLA" THEN
  PRINT #ITER, "INERZIA DELLA SEZIONE RETTA DELL'ARCO A SINIS-
    TRA = "; INERZS
  PRINT #ITER, "INERZIA DELLA SEZIONE RETTA DELL'ARCO AL CEN-
    TRO = "; INERZC
  PRINT #ITER, "INERZIA DELLA SEZIONE RETTA DELL'ARCO A DE-
    STRA = "; INERZD
END IF
IF SEZIONE$ = "COSENOW" THEN

```

```

PRINT #ITER, "INERZIA DELLA SEZIONE RETTA DELL'ARCO AL CEN-
TRO = "; INERZC
END IF
PRINT #ITER,
IF CARICHI$ = "FUNC" THEN
  PRINT #ITER, "CARICO VERTICALE UNIFORME = "; QUNIF
END IF
IF MASSE$ = "VERTICALE" THEN
  PRINT #ITER, "MASSE DISTRIBUITE SULL'IMPALCATO"
ELSE
  PRINT #ITER, "MASSE DISTRIBUITE LUNGO L'ARCO"
END IF
REM -----
REM STAMPA LE QUOTE DELL'ARCO PER CIASCUNA DIVIDENTE
REM -----
IF ASSE$ = "DATI" THEN
PRINT #ITER,
  PRINT #ITER, "===== "
  PRINT #ITER, " = "
  PRINT #ITER, " =          TABELLA DELLE QUOTE DELL'ARCO          = "
  PRINT #ITER, " = "
  PRINT #ITER, "===== "
  PRINT #ITER,
  PRINT #ITER, " _____ "
  PRINT #ITER, " DIVIDENTE          QUOTA          "
  PRINT #ITER, " _____ "
  PRINT #ITER,
  FOR I = 1 TO T + 1
    PRINT #ITER, TAB(1); I; TAB(34); Q(I)
  NEXT I
END IF
IF CARICHI$ = "DATI" THEN
REM -----
REM STAMPA I CARICHI PER CIASCUNA DIVIDENTE
REM -----
PRINT #ITER, "SPINTA ORIZZONTALE = "; HSPINTA
PRINT #ITER,
PRINT #ITER, "===== "
PRINT #ITER, " = "
PRINT #ITER, " =          CARICHI VERTICALI SULL'ARCO          = "
PRINT #ITER, " = "
PRINT #ITER, "===== "

```

```

PRINT #ITER,
PRINT #ITER, " _____"
PRINT #ITER, " DIVIDENTE          CARICO VERTICALE          "
PRINT #ITER, " _____"
PRINT #ITER,
FOR I = 1 TO T + 1
  PRINT #ITER, TAB(2); I; TAB(20); FVERT(I)
NEXT I
END IF
IF SEZIONE$ = "DATI" THEN
  REM _____
  REM STAMPA I MOMENTI DI INERZIA PER CIASCUNA DIVIDENTE
  REM _____
  PRINT #ITER, " ====="
  PRINT #ITER, " = "
  PRINT #ITER, " = MOMENTI DI INERZIA DELLA SEZIONE RETTA = "
  PRINT #ITER, " = "
  PRINT #ITER, " ====="
  PRINT #ITER,
  PRINT #ITER, " _____"
  PRINT #ITER, " DIVIDENTE          MOMENTO DI INERZIA          "
  PRINT #ITER, " _____"
  PRINT #ITER,
  FOR I = 1 TO T + 1
    PRINT #ITER, TAB(2); I; TAB(20); INERZ(I)
  NEXT I
END IF
NEXT ITER
END SUB

SUB UscitaRisultati
REM
REM *****
REM *
REM *          U s c i t a R i s u l t a t i          *
REM *
REM *****
REM
REM *****
REM *
REM *          Questa subroutine si occupa dell'uscita (su video, su stampa
REM *          e su file) dei risultati          *

```

```

REM *
REM *****
PI = 4 * ATN(1)
FOR ITER = ITER1 + 1 TO ITER2
  PRINT #ITER, "    FREQUENZE CIRCOLARI    PERIODO"
  PRINT #ITER,
  FOR I = 1 TO NEIG
    PRINT #ITER, TAB(2); I; TAB(7); EIG(I); TAB(30); 2 * PI / SQR(EIG(I))
  NEXT I
  PRINT #ITER,
  FOR I = 1 TO NEIG
    PRINT #ITER,
    PRINT #ITER, "AUTOVETTORE N. "; I
    PRINT #ITER,
    FOR K = 1 TO T - 2
      PRINT #ITER, TAB(2); K; TAB(15); VECT(K, I)
    NEXT K
  NEXT I
  PRINT #ITER,
  PRINT #ITER, "COEFFICIENTI DI PARTECIPAZIONE"
  PRINT #ITER,
  FOR K = 1 TO NEIG
    PRINT #ITER, TAB(2); K; TAB(15); PART(K)
  NEXT K
  PRINT #ITER,
  PRINT #ITER, "I  SPOST. VERTICALI  SPOST. ORIZZONTALI"
  PRINT #ITER,
  FOR K = 1 TO T + 1
    PRINT #ITER, TAB(2); K; TAB(7); SV(K); TAB(35); SW(K)
  NEXT K
  PRINT #ITER,
  PRINT #ITER, "I  FORZE VERTICALI  FORZE ORIZZONTALI"
  PRINT #ITER,
  FOR K = 1 TO T + 1
    PRINT #ITER, TAB(2); K; TAB(7); SISV(K); TAB(35); SISO(K)
  NEXT K
  PRINT #ITER,
  PRINT #ITER, "I  MOMENTI FLETTENTI"
  PRINT #ITER,
  FOR K = 1 TO T + 1
    PRINT #ITER, TAB(2); K; TAB(10); MOM(K)
  NEXT K

```

406 *Le strutture ad arco*

```
PRINT #ITER,  
PRINT #ITER, "TEMPO DI ESECUZIONE = "; TIME2; " SEC."  
PRINT #ITER, "RUN DEL "; DATE$; " ORE "; TIME$  
NEXT ITER  
END SUB
```