

RUN DEL 10-06-1994 ORE 08:34:45

Minime modifiche permettono di ottenere il diagramma del taglio per effetto del carico accidentale, ed i diagrammi del momento e del taglio per effetto del treno viaggiante. Si ha quindi, ad esempio:

DIAGRAMMA DEI MASSIMI E MINIMI DEL TAGLIO
STRUTTURA PERCORSATA DA STESA DI CARICO UNIFORME

ASCISSA	MASSIMI	MINIMI
0	129.507739048598	-329.71234096461
5	154.750053155301	-293.769450534378
10	197.950626282189	-239.075127096871
15	250.940952019381	-181.398768631202
20	315.866994715612	-121.956340241701
24.9995	395.386922371767	-64.4952724528559
25.0005	1.75444959748857	-313.957154808054
30	37.2849547362104	-231.722989288302
35	104.585949754331	-168.68465567014
40	206.996382245838	-129.573816649515
45	322.164593390286	-96.4189428113572
49.9995	441.332314425357	-66.3563144276022
50.0005	66.3518502789789	-441.346341799976
55	96.4138907142536	-322.17790143083
60	129.568164279993	-207.008711670354
65	168.675276500844	-104.593970577309
70	231.71145553203	-37.2900949443243
74.9995	313.942524387651	-1.75567561417576
75.0005	64.488275303518	-395.395781491616
80	121.948261053963	-315.873926597878
85	181.390505989627	-250.946851480547
90	239.066815697079	-197.955657471589
95	293.761184724794	-154.754354827353
100	329.704146182062	-129.511603198255

TEMPO DI ESECUZIONE = 9.01171875 SEC.

RUN DEL 10-06-1994 ORE 16:31:02

In Figura 5.17 sono riportati i quattro diagrammi citati.

Un secondo esempio è relativo all'arco di 400 metri di luce, già studiato nei capitoli precedenti. Nella Figura 5.18 sono riportati i diagrammi dei momenti (baricentrici, intradossali ed estradossali) e dei tagli per effetto del carico accidentale, in Figura 5.19

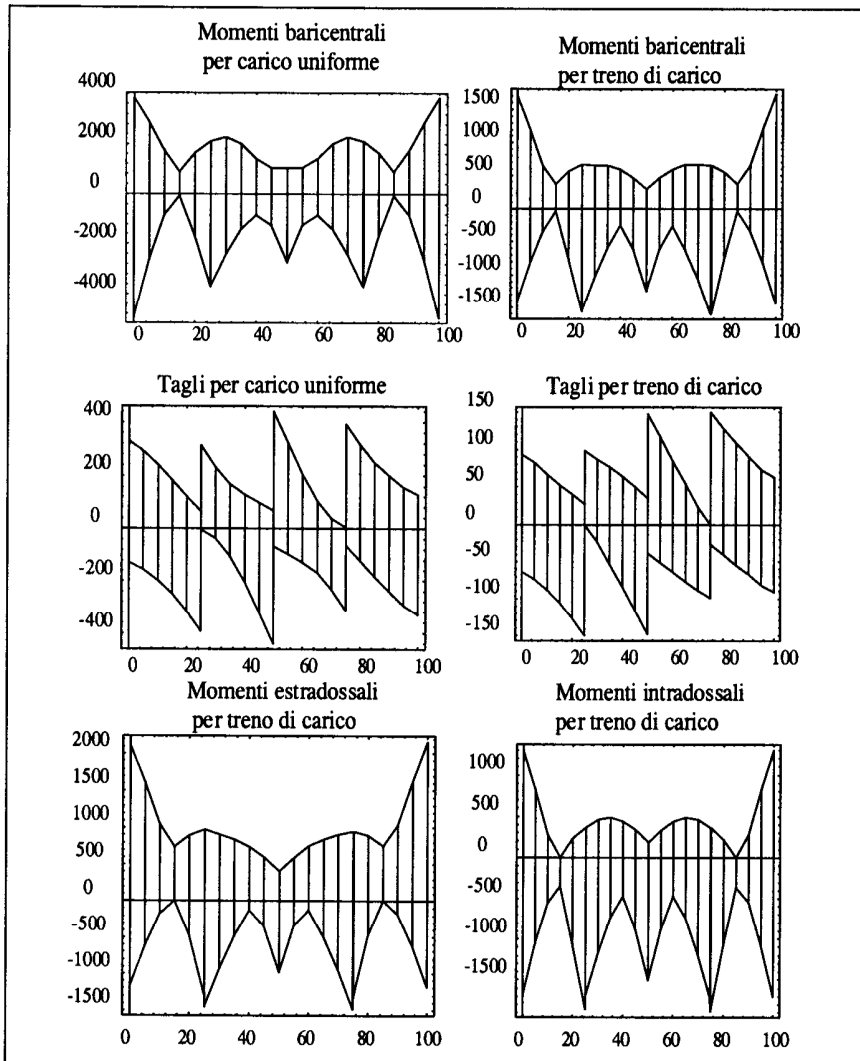


Figura 5.17- I diagrammi degli estremi per l'arco di prova

5. I carichi mobili 221

gli stessi grafici per treno viaggiante. I valori numerici — non riportati per brevità — possono comunque essere ritrovati nei files .USC allegati in disco.



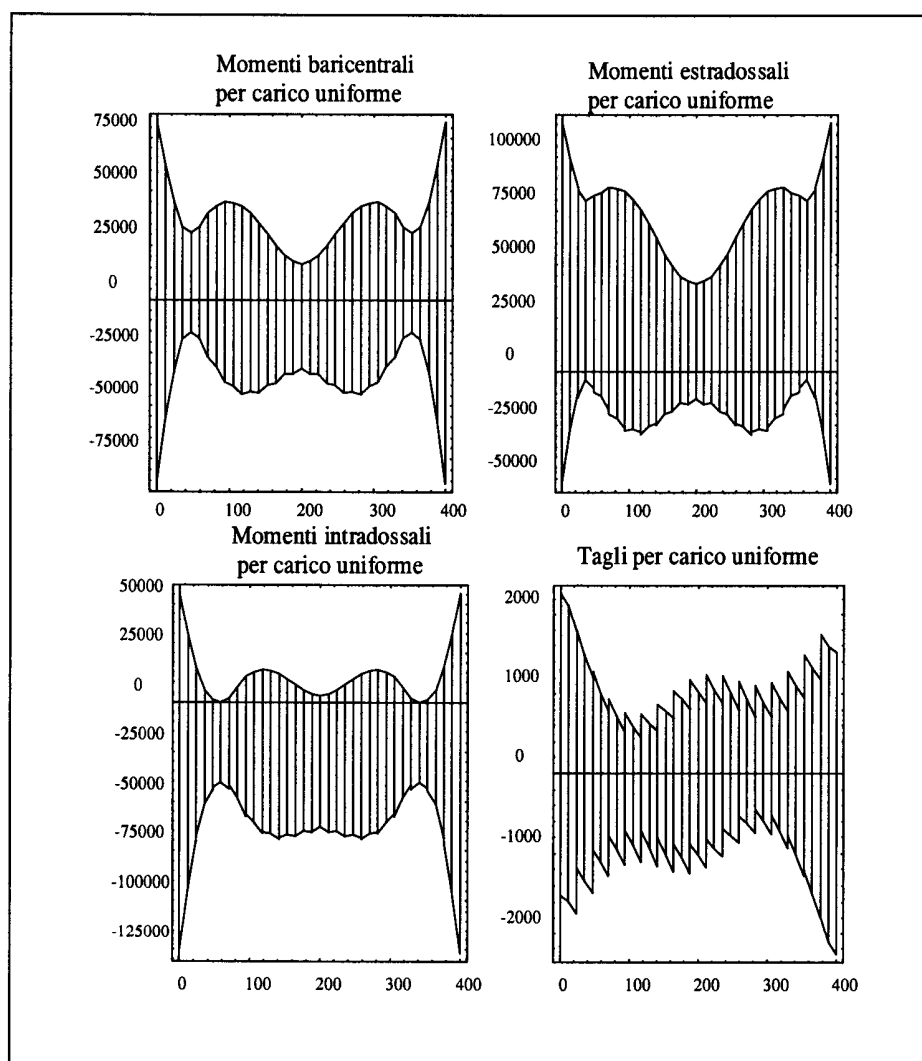


Figura 5.18- I diagrammi degli estremi per l'arco di Krk
Il caso del carico uniforme

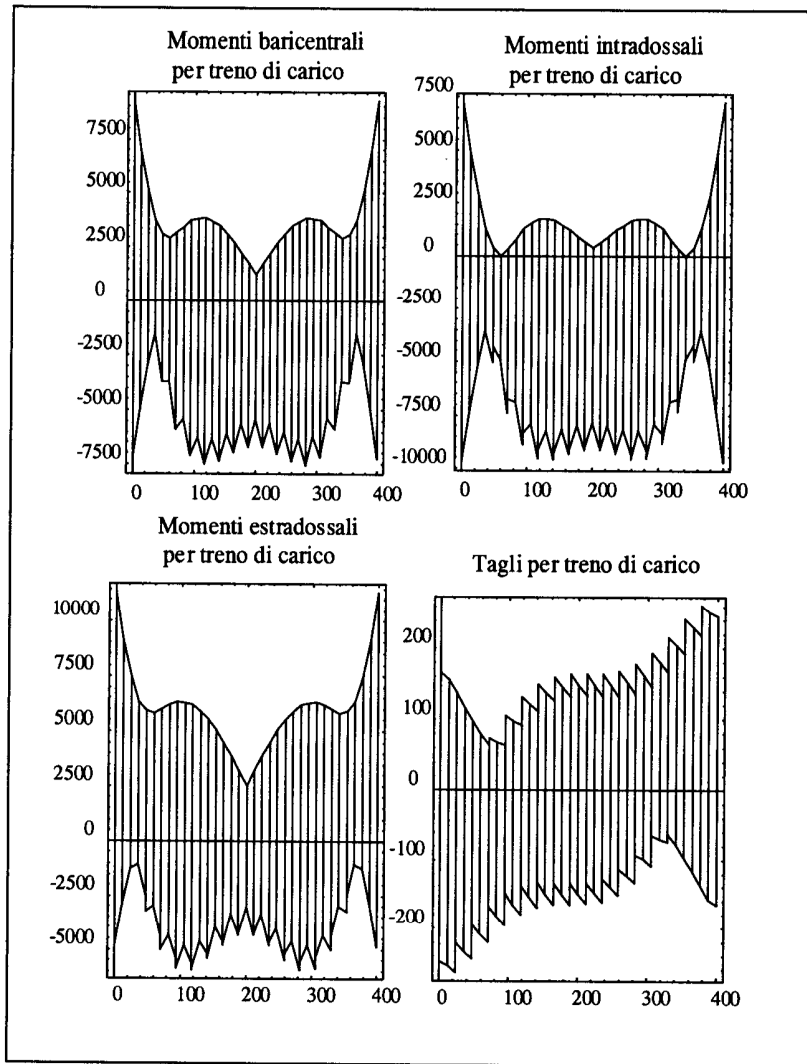


Figura 5.19- I diagrammi degli estremi per l'arco di Krk
Il caso del treno di carico

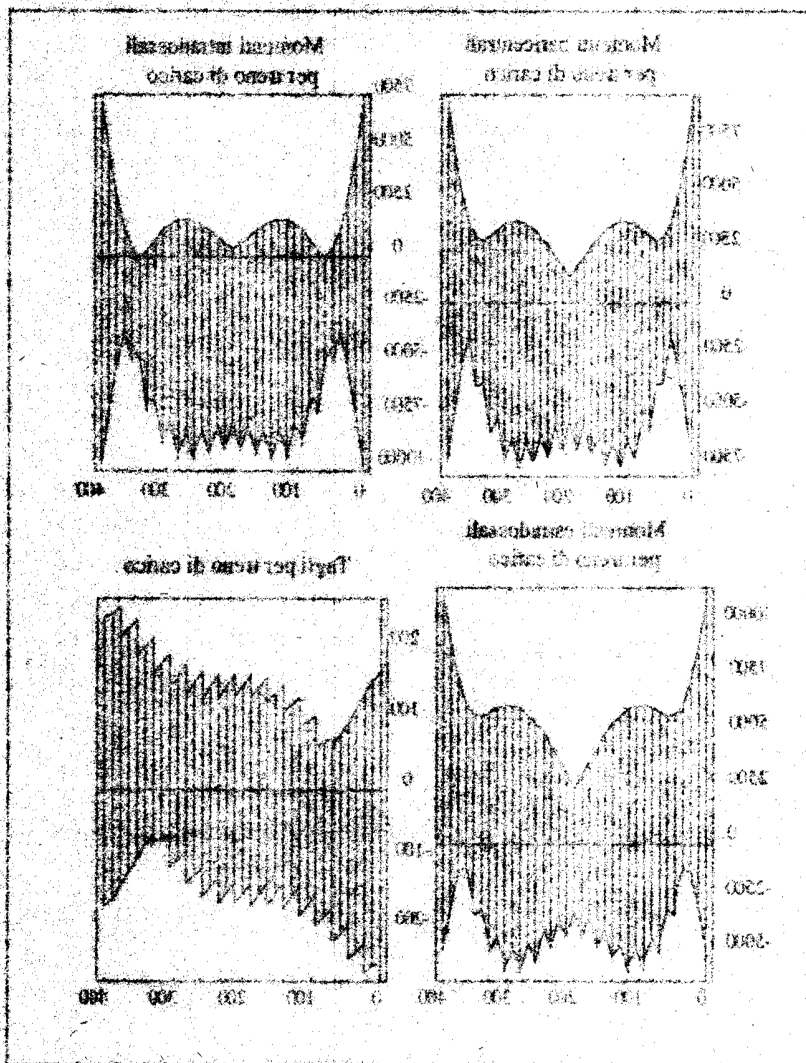


Fig. 1. Sound pressure level versus time for different conditions.

Appendice 5.1. Il programma AVL

```
DEFDBL A-Z
DECLARE SUB Aree ()
DECLARE SUB Estremi ()
DECLARE SUB LineaDiInfluenza ()
DECLARE SUB Caratteristiche ()
DECLARE SUB Cedibilita ()
DECLARE SUB TerminiNoti ()
DECLARE SUB Geometria ()
DECLARE SUB Spostamenti ()
DECLARE SUB Iperstatiche ()
DECLARE SUB slsim (A#(), B#(), X#(), N#)
DECLARE SUB Imposte (A#(), B#(), RA#(), RB#(), PA#(), PB#(), C#)
DECLARE SUB product (A#(), B#(), C#(), N#)
DECLARE SUB dot (A#(), B#(), C#, N#)
DECLARE SUB Plv (MA#(), MB#(), NA#(), NB#(), TA#(), TB#(), C#)
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 V L          *
REM *
REM *                      d i                      *
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. Gennaio 1988 (in BASIC HP) *
REM *          Release 3.0.0. Agosto 1994 (in Microsoft QBasic) *
REM *
REM *****
```

```

REM
REM
REM *****
REM *
REM *      Questo programma studia l'arco in fase elastica tracciando le
REM *      linee di influenza dei momenti, baricentrici, estradossali ed
REM *      intradossali, dello sforzo normale e del taglio, per effetto
REM *      di un treno di forze verticali viaggianti
REM *
REM *****
OPTION BASE 1
REM
REM *****
REM *
REM *      C O M M O N
REM *
REM *****
REM
COMMON SHARED LUCE, T, E, NU, G, FREQ, RIAL, TRAT, CONC
COMMON SHARED NFOR, KRAP, ZINF
COMMON SHARED LARS, LARC, LARD, ALTS, ALTC, ALTD
COMMON SHARED SPES, SPEC, SPED, SPTS, SPTC, SPTD
COMMON SHARED LMIN, LMAX, ZMIN, ZMAX, DELTA, SMALLDELTA
COMMON SHARED BIGDELTA, ITER1, ITER2, TIME2, APOS, ANEG
COMMON SHARED M() AS DOUBLE, M1() AS DOUBLE
COMMON SHARED M2() AS DOUBLE, M3() AS DOUBLE
COMMON SHARED TG() AS DOUBLE, T1() AS DOUBLE
COMMON SHARED T2() AS DOUBLE, T3() AS DOUBLE
COMMON SHARED N() AS DOUBLE, N1() AS DOUBLE
COMMON SHARED N2() AS DOUBLE, N3() AS DOUBLE
COMMON SHARED X() AS DOUBLE, INCL() AS DOUBLE
COMMON SHARED R0() AS DOUBLE, P0() AS DOUBLE
COMMON SHARED R() AS DOUBLE, P() AS DOUBLE
COMMON SHARED R1() AS DOUBLE, R2() AS DOUBLE, R3() AS DOUBLE
COMMON SHARED P1() AS DOUBLE, P2() AS DOUBLE, P3() AS DOUBLE
COMMON SHARED C() AS DOUBLE, AREA() AS DOUBLE
COMMON SHARED CHI() AS DOUBLE, Q() AS DOUBLE
COMMON SHARED TREN() AS DOUBLE, DIST() AS DOUBLE
COMMON SHARED A() AS DOUBLE, B() AS DOUBLE, BB() AS DOUBLE
COMMON SHARED V() AS DOUBLE, PHI() AS DOUBLE
COMMON SHARED LDI() AS DOUBLE, INERZ() AS DOUBLE
COMMON SHARED TITLE$, STAMPA$, STAMPARIS$, FILEINGR$, FILEUSC$

```



```

COMMON SHARED ASSE$, FILEINPUT$, LINF$
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), LINF$(30)
DIM R0(3) AS DOUBLE, R1(3) AS DOUBLE, R2(3) AS DOUBLE
DIM P0(3) AS DOUBLE, P1(3) AS DOUBLE, P2(3) AS DOUBLE
DIM P(3) AS DOUBLE, R(3) AS DOUBLE, R3(3) AS DOUBLE, P3(3) AS DOUBLE
DIM C(3, 3) AS DOUBLE, BB(3) AS DOUBLE, X(3) AS DOUBLE
DIM A(3, 3) AS DOUBLE, B(3, 3) AS DOUBLE REM
REM *****
REM *
REM *          DEFINISCE ALCUNE LEGGI DI VARIAZIONE          *
REM *
REM *****
REM
DEF FNA (Z) = LARS + (LARD - LARS) / LUCE * Z - 4 * (LARS + (LARD -
    LARS) / 2 - LARC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNB (Z) = ALTS + (ALTD - ALTS) / LUCE * Z - 4 * (ALTS + (ALTD -
    ALTS) / 2 - ALTC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNC (Z) = SPTS + (SPTD - SPTS) / LUCE * Z - 4 * (SPTS + (SPTD - SPTS)
    / 2 - SPTC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNS (Z) = SPES + (SPED - SPES) / LUCE * Z - 4 * (SPES + (SPED - SPES)
    / 2 - SPEC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNQ (Z) = RIAL * Z / LUCE + 4 * FREC / LUCE ^ 2 * Z * (LUCE - Z)
CLS
REM
REM *****
REM *
REM *          LEGGE IL NOME DEL FILE DI INPUT DATI          *
REM *
REM *****
REM
PRINT "NOME DEL FILE DI INPUT"
INPUT FILEINPUT$
TIME1 = TIMER
OPEN FILEINPUT$ FOR INPUT AS #7

```

```

REM -----
REM CHIAMA LA SUBROUTINE PER L'INGRESSO DEI DATI
REM -----
CALL IngressoDati
T = TRAT * CONC
DELTA = LUCE / T
BIGDELTA = LUCE / TRAT
SMALLDELTA = BIGDELTA / KRAP                                     'cfr. 5.63
G = E / (2 * (1 + NU))
REM
REM *****
REM *
REM *          REDIMENSIONA GLI ARRAY DINAMICI
REM *
REM *****
REM
REDIM M(T + 1) AS DOUBLE, M1(T + 1) AS DOUBLE
REDIM M2(T + 1) AS DOUBLE, M3(T + 1) AS DOUBLE
REDIM TG(T + 1) AS DOUBLE, T1(T + 1) AS DOUBLE
REDIM T2(T + 1) AS DOUBLE, T3(T + 1) AS DOUBLE
REDIM N(T + 1) AS DOUBLE, N1(T + 1) AS DOUBLE
REDIM N2(T + 1) AS DOUBLE, N3(T + 1) AS DOUBLE
REDIM TREN(NFOR) AS DOUBLE, DIST(NFOR) AS DOUBLE
REDIM AREA(T + 1) AS DOUBLE, INERZ(T + 1) AS DOUBLE
REDIM CHI(T + 1) AS DOUBLE, Q(T + 1) AS DOUBLE
REDIM V(T + 1) AS DOUBLE, PHI(T + 1) AS DOUBLE
REDIM LDI(TRAT + 1) AS DOUBLE, INCL(T + 1) 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
REM LA MATRICE DELLE FLESSIBILITA'
REM _____
CALL Cedibilita
REM _____
REM CHIAMA LA SUBROUTINE PER CALCOLARE
REM I TERMINI NOTI DELLE EQUAZIONI DI CONGRUENZA
REM _____
CALL TerminiNoti
REM _____
REM CHIAMA LA SUBROUTINE PER CALCOLARE LE IPERSTATICHE
REM _____
CALL Iperstatiche
REM _____
REM CHIAMA LA SUBROUTINE PER CALCOLARE LE CARATTERISTICHE
REM _____
CALL Caratteristiche
REM _____
REM CHIAMA LA SUBROUTINE PER CALCOLARE GLI SPOSTAMENTI
REM _____
CALL Spostamenti
REM _____
REM CHIAMA LE SUBROUTINE PER CALCOLARE LE LINEE DI INFLUENZA,
REM LE AREE POSITIVE E NEGATIVE, I PUNTI DI MASSIMO E
REM MINIMO CON LE RISPETTIVE ASCISSE
REM _____
CALL LineaDiInfluenza
CALL Aree
CALL Estremi
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 Aree
REM
```

```

REM *****
REM *
REM *           A r e e
REM *
REM *****
REM
REM
REM *****
REM *
REM * QUESTA SUBROUTINE CALCOLA LE AREE POSITIVE E NEGATIVE *
REM *           SOTTESE DALLA LINEA DI INFLUENZA
REM *
REM *****
REM
APOS = 0
ANEG = 0
FOR I = 1 TO TRAT
  IF LDI(I) * LDI(I + 1) >= 0 THEN
    IF LDI(I) <= 0 AND LDI(I + 1) <= 0 THEN
      ANEG = ANEG + (LDI(I + 1) + LDI(I)) / 2 * BIGDELTA ' Area negativa
    ELSE
      APOS = APOS + (LDI(I + 1) + LDI(I)) / 2 * BIGDELTA ' Area positiva
    END IF
  ELSE
    ZNUL = BIGDELTA * LDI(I) / (LDI(I) - LDI(I + 1)) ' Punto di nullo
    IF LDI(I) >= 0 THEN
      APOS = APOS + LDI(I) * ZNUL / 2
      ANEG = ANEG + LDI(I + 1) * (BIGDELTA - ZNUL) / 2
    ELSE
      ANEG = ANEG + LDI(I) * ZNUL / 2
      APOS = APOS + LDI(I + 1) * (BIGDELTA - ZNUL) / 2
    END IF
  END IF
END IF
NEXT I
END SUB

SUB Caratteristiche
REM
REM *****
REM *
REM *           C a r a t t e r i s t i c h e
REM *
REM *

```

```

REM *****
REM
REM *****
REM *
REM *      QUESTA SUBROUTINE CALCOLA LE CARATTERISTICHE, *
REM *      NOTE LE INCOGNITE IPERSTATICHE *
REM *
REM *****
REM
FOR I = 1 TO T + 1
  M(I) = X(1) * M1(I) + X(2) * M2(I) + X(3) * M3(I)
  N(I) = X(1) * N1(I) + X(2) * N2(I) + X(3) * N3(I)
  TG(I) = X(1) * T1(I) + X(2) * T2(I) + X(3) * T3(I)
NEXT I
END SUB

SUB Estremi
REM
REM *****
REM *
REM *      E s t r e m i *
REM *
REM *****
REM
REM *****
REM *
REM *      QUESTA SUBROUTINE CALCOLA I VALORI ESTREMI *
REM *      DELLA LINEA DI INFLUENZA *
REM *
REM *****
REM
LMAX = 0
LMIN = 0
ZMAX = 0
ZMIN = 0
N0 = INT(DIST(NFOR) / SMALLDELTA) + 1 ' cfr. 5.68
FOR I = 1 TO KRAP * TRAT + N0
  W = 0
  FOR J = 1 TO NFOR
    Z = I * SMALLDELTA - DIST(J)
    IF Z >= 0 AND Z <= LUCE THEN
      H = INT(Z / BIGDELTA) + 1 ' cfr. 5.60

```

```

        AUX = LDI(H) + (LDI(H + 1) - LDI(H)) / BIGDELTA * (Z - (H - 1) *
        BIGDELTA)
        W = W + AUX * TREN(J)                                     ' cfr. 5.61
    END IF
NEXT J
W1 = LMAX
W2 = LMIN
IF W > LMAX THEN LMAX = W
IF W > W1 THEN ZMAX = I
IF W < LMIN THEN LMIN = W
IF W < W2 THEN ZMIN = I
NEXT I
ZMIN = ZMIN * SMALLDELTA
ZMAX = ZMAX * SMALLDELTA
END SUB

SUB Geometria
REM
REM *****
REM *
REM *
REM *
REM *
REM *
REM *****
REM
REM
REM -----
REM Riempimento degli array delle aree, delle inerzie e dei fattori di taglio
REM -----
FOR I = 1 TO T + 1
    Z = LUCE / T * (I - 1)
    AREA(I) = FNA(Z) * FNB(Z) - (FNA(Z) - FNC(Z)) * (FNB(Z) - 2 * FNS(Z))
    INERZ(I) = FNA(Z) * FNB(Z) ^ 3 / 12 - (FNA(Z) - FNC(Z)) * (FNB(Z) - 2 *
    FNS(Z)) ^ 3 / 12
    CHI(I) = AREA(I) / (FNC(Z) * (FNB(Z) - 2 * FNS(Z)))
NEXT I
REM -----
REM Riempimento dell'array delle quote, se descritte da funzioni
REM -----
IF ASSE$ = "FUNC" THEN
    FOR I = 1 TO T + 1
        Z = LUCE / T * (I - 1)
        Q(I) = FNQ(Z)
    NEXT I

```

```

END IF
REM -----
REM Riempimento degli array delle inclinazioni dei tratti
REM -----
INCL(1) = ATN(Q(2) / LUCE * T)
FOR I = 2 TO T
  INCL(I) = ATN((Q(I + 1) - Q(I - 1)) / LUCE / 2 * T)
NEXT I
INCL(T + 1) = ATN((Q(T + 1) - Q(T)) / LUCE * T)
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 non
REM *          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-
  TAFILES" OR STAMPA$ = "TUTTO" THEN
  INPUT #7, FILEINGR$
END IF
IF STAMPARIS$ = "FILES" OR STAMPARIS$ = "VIDEOFILES" OR STAM-
  PARIS$ = "CARTAFILES" OR STAMPARIS$ = "TUTTO" THEN
  INPUT #7, FILEUSC$
END IF
CALL OpenFiles
REM
REM *****
REM *
REM * LEGGE:
REM * LUCE Luce dell'arco
REM * TRAT Numero di tratti tra le stilate
REM * CONC Numero dei conci per tratto
REM * NFOR Numero delle forze del treno
REM * KRAP Rapporto tra la luce del tratto tra le
REM * stilate ed il passo di transito del treno
REM * E Modulo di Young
REM * NU Modulo di Poisson
REM * LARS Larghezza della sezione retta dell'arco a sinistra
REM * LARC Larghezza della sezione retta dell'arco al centro
REM * LARD Larghezza della sezione retta dell'arco a destra
REM * ALTS Altezza della sezione retta dell'arco a sinistra
REM * ALTC Altezza della sezione retta dell'arco al centro
REM * ALTD Altezza della sezione retta dell'arco a destra
REM * SPES Spessore della soletta superiore o
REM * inferiore della sezione retta a sinistra
REM * SPEC Spessore della soletta superiore o
REM * inferiore della sezione retta al centro.
REM * SPED Spessore della soletta superiore o
REM * inferiore della sezione retta a destra
REM * SPTS Spessore complessivo dei timpani dell'arco a sinistra
REM * SPTC Spessore complessivo dei timpani dell'arco al centro
REM * SPTD Spessore complessivo dei timpani dell'arco a destra
REM * A(3,3) Matrice (3,3) dei cedimenti elastici della
REM * imposta di sinistra

```



```

REM *      B(3,3)  Matrice (3,3) dei cedimenti elastici della          *
REM *      imposta di destra                                          *
REM *      ZINF   Ascissa della sezione in cui si vuole la linea di influenza *
REM *      INFL$  Può assumere i seguenti valori:                    *
REM *      MOMENTO BARICENTRICO                                       *
REM *      MOMENTO INTRADOSSALE                                       *
REM *      MOMENTO ESTRADOSSALE                                       *
REM *      SFORZO NORMALE                                             *
REM *      TAGLIO                                                       *
REM *      ASSE$  Se l'asse dell'arco e' descritto da una            *
REM *      funzione, ASSE$ = FUNZ, se invece e' dato                  *
REM *      per punti, allora ASSE$ = DATI                             *
REM *      Se l'asse e' definito da funzione:                          *
REM *      FREC   Freccia dell'arco                                    *
REM *      RIAL   Rialzo a destra                                     *
REM *                                                                 *
REM *****
REM
INPUT #7, LUCE, TRAT, CONC, NFOR, KRAP, E, NU
INPUT #7, LARS, LARC, LARD, ALTS, ALTC, ALTD
INPUT #7, SPES, SPEC, SPED, SPTS, SPTC, SPTD
FOR I = 1 TO 3
  FOR J = 1 TO 3
    INPUT #7, A(I, J)
  NEXT J
NEXT I
FOR I = 1 TO 3
  FOR J = 1 TO 3
    INPUT #7, B(I, J)
  NEXT J
NEXT I
INPUT #7, ZINF, LINF$
INPUT #7, ASSE$
IF ASSE$ = "FUNC" THEN
  INPUT #7, FREC, RIAL
END IF
END SUB

SUB IngressoDatiArray
REM
REM *****
REM *

```

```

REM *           I n g r e s s o D a t i A r r a y           *
REM *
REM *****
REM
REM *****
REM *
REM *           Questa subroutine si occupa dell'ingresso dei dati           *
REM *           organizzati in array                                         *
REM *
REM *****
REM
REM _____
REM Legge le forze del treno e la loro distanza da destra
REM _____
FOR I = 1 TO NFOR
  INPUT #7, TREN(I)
NEXT I
FOR I = 1 TO NFOR
  INPUT #7, DIST(I)
NEXT I
REM _____
REM Legge le quote dell'arco
REM _____
IF ASSE$ = "DATI" THEN
  FOR I = 1 TO T + 1
    INPUT #7, Q(I)
  NEXT I
  RIAL = Q(T + 1)
  FREC = Q(T / 2 + 1) - RIAL / 2
END IF
REM
REM *****
REM *
REM *           Legge i carichi verticali ed orizzontali           *
REM *
REM *****
END SUB

SUB LineaDiInfluenza
REM
REM *****
REM *

```

```

REM *                               L i n e a D i I n f l u e n z a *
REM *                               *
REM *****
REM
REM *****
REM *                               *
REM *   QUESTA SUBROUTINE CALCOLA LA LINEA DI INFLUENZA *
REM *                               D E S I D E R A T A *
REM *                               *
REM *****
REM Completamento della formula 5.29
IF ZINF = LUCE THEN
  INCL = INCL(T + 1)
ELSE
  U = INT(ZINF / DELTA) + 1
  ZAUX = (ZINF - (U - 1) * DELTA) / DELTA
  INCL = INCL(U) + (INCL(U + 1) - INCL(U)) * ZAUX
END IF
B = FNB(ZINF)
SELECT CASE LINF$
CASE "MOMENTO BARICENTRICO"
  FOR I = 1 TO T + 1
    IF ZINF < (I - 1) * DELTA THEN
      V(I) = V(I) - (DELTA * (I - 1) - ZINF)
    END IF
  NEXT I
CASE "MOMENTO INTRADOSSALE"
  FOR I = 1 TO T + 1
    IF ZINF < (I - 1) * DELTA THEN
      V(I) = V(I) - (DELTA * (I - 1) - ZINF - B / 2 * SIN(INCL))
    END IF
  NEXT I
CASE "MOMENTO ESTRADOSSALE"
  FOR I = 1 TO T + 1
    IF ZINF < (I - 1) * DELTA THEN
      V(I) = V(I) - (DELTA * (I - 1) - ZINF + B / 2 * SIN(INCL))
    END IF
  NEXT I
CASE "SFORZO NORMALE"
  FOR I = 1 TO T + 1
    IF ZINF < (I - 1) * DELTA THEN
      V(I) = V(I) - SIN(INCL)
    END IF
  NEXT I

```

```

      END IF
    NEXT I
CASE "TAGLIO"
  FOR I = 1 TO T + 1
    IF ZINF < (I - 1) * DELTA THEN
      V(I) = V(I) + COS(INCL)
    END IF
  NEXT I
CASE ELSE
  CLS
  PRINT "SCELTA NON VALIDA PER LA LINEA DI INFLUENZA"
  STOP
END SELECT
LDI(1) = 0
LDI(TRAT + 1) = 0
FOR I = 2 TO TRAT
  H = (I - 1) * CONC + 1
  LDI(I) = V(H)
NEXT I
END SUB

```

SUB Spostamenti

```

REM
REM *****
REM *
REM *           S p o s t a m e n t i
REM *
REM *****
REM
REM *****
REM *
REM *   QUESTA SUBROUTINE CALCOLA SPOSTAMENTI E ROTAZIONI
REM *
REM *****
REM
DIM U(3), S3(3)
FOR I = 1 TO T
  M = (M(I) + M(I + 1)) / 2
  N = (N(I) + N(I + 1)) / 2
  TG = (TG(I) + TG(I + 1)) / 2
  AREA = (AREA(I) + AREA(I + 1)) / 2
  INERZ = (INERZ(I) + INERZ(I + 1)) / 2

```

```

PEND = (INCL(I) + INCL(I + 1)) / 2
HH = (FNB(I + 1) + FNB(I)) / 2
QQ = Q(I + 1) - Q(I)
CHI = (CHI(I) + CHI(I + 1)) / 2
LUNGH = SQR((Q(I + 1) - Q(I)) ^ 2 + (LUCE / T) ^ 2)
REM Cfr. (5.30)
V(I + 1) = V(I) - M / E / INERZ * LUNGH * LUCE / T / 2
V(I + 1) = V(I + 1) - N / E / AREA * LUNGH * SIN(PEND)
V(I + 1) = V(I + 1) + CHI * TG / G / AREA * LUNGH * COS(PEND)
V(I + 1) = V(I + 1) - PHI(I) * LUCE / T
REM Cfr. (5.31)
PHI(I + 1) = PHI(I) + M / E / INERZ * LUNGH
NEXT I
REM Cfr. (4.52)
FOR I = 1 TO 3
  R(I) = R0(I) + X(1) * R1(I) + X(2) * R2(I) + X(3) * R3(I)
  P(I) = P0(I) + X(1) * P1(I) + X(2) * P2(I) + X(3) * P3(I)
NEXT I
CALL product(A(), R(), U(), 3)
FOR I = 1 TO 3: S3(I) = SA(I) - U(I): NEXT I
REM Cfr. (4.53)
FOR I = 1 TO T + 1
  Z = LUCE / T * (I - 1)
  V(I) = V(I) + S3(1) - S3(3) * Z
  PHI(I) = PHI(I) + S3(3)
NEXT I
END SUB

```

```

SUB TerminiNoti
REM
REM *****
REM *
REM *           T e r m i n i N o t i
REM *
REM *****
REM
REM *****
REM *
REM *           Q U E S T A   S U B R O U T I N E   C A L C O L A   I   T E R M I N I   N O T I
REM *           D E L L E   E Q U A Z I O N I   D I   C O N G R U E N Z A
REM *
REM *****

```

```

REM -----
REM Estrazione delle caratteristiche geometriche in ZINF
REM -----
IF ZINF = LUCE THEN
  U = T + 1
  Q = Q(T + 1)
  INCL = INCL(T + 1)
ELSE
  U = INT(ZINF / DELTA) + 1
  ZAUX = (ZINF - (U - 1) * DELTA) / DELTA
  Q = Q(U) + (Q(U + 1) - Q(U)) * ZAUX
  INCL = INCL(U) + (INCL(U + 1) - INCL(U)) * ZAUX
END IF
B = FNB(ZINF)
REM -----
REM Riempe l'array dei termini noti con i casi opportuni
REM -----
SELECT CASE LINF$
CASE "MOMENTO BARICENTRICO"
  REM Formule 5.22, 5.23, 5.24
  BB(1) = (RIAL / 2 / FREC / LUCE - 1 / LUCE) * ZINF - 1 / 2 / FREC * Q + 1
  BB(2) = -RIAL / FREC / LUCE * ZINF + Q / FREC
  BB(3) = (RIAL / 2 / FREC + 1 / LUCE) * ZINF - Q / 2 / FREC
CASE "MOMENTO INTRADOSSALE"
  REM Formula 5.53
  BB(1) = (RIAL / 2 / FREC / LUCE - 1 / LUCE) * ZINF - 1 / 2 / FREC * Q + 1
  BB(1) = BB(1) - B / 2 * (1 / LUCE - RIAL / 2 / FREC / LUCE) * SIN(INCL)
  BB(1) = BB(1) + B / 2 * COS(INCL) / 2 / FREC
  BB(2) = -RIAL / FREC / LUCE * ZINF + Q / FREC
  BB(2) = BB(2) - B / 2 * RIAL / FREC / LUCE * SIN(INCL)
  BB(2) = BB(2) - B / 2 * COS(INCL) / FREC
  BB(3) = (RIAL / 2 / FREC + 1 / LUCE) * ZINF - Q / 2 / FREC
  BB(3) = BB(3) - B / 2 * (-(RIAL / 2 / FREC / LUCE + 1 / LUCE) * SIN(INCL))
  BB(3) = BB(3) + B / 2 * COS(INCL) / 2 / FREC
CASE "MOMENTO ESTRADOSSALE"
  REM Formula 5.52
  BB(1) = (RIAL / 2 / FREC / LUCE - 1 / LUCE) * ZINF - 1 / 2 / FREC * Q + 1
  BB(1) = BB(1) + B / 2 * (1 / LUCE - RIAL / 2 / FREC / LUCE) * SIN(INCL)
  BB(1) = BB(1) - B / 2 * COS(INCL) / 2 / FREC
  BB(2) = -RIAL / FREC / LUCE * ZINF + Q / FREC
  BB(2) = BB(2) + B / 2 * RIAL / FREC / LUCE * SIN(INCL)
  BB(2) = BB(2) + B / 2 * COS(INCL) / FREC

```

```

BB(3) = (RIAL / 2 / FREC + 1 / LUCE) * ZINF - Q / 2 / FREC
BB(3) = BB(3) + B / 2 * (-RIAL / 2 / FREC / LUCE + 1 / LUCE) * SIN(INCL))
BB(3) = BB(3) - B / 2 * COS(INCL) / 2 / FREC
CASE "SFORZO NORMALE"
  REM Formula 5.39
  BB(1) = (1 / LUCE - RIAL / 2 / FREC / LUCE) * SIN(INCL)
  BB(1) = BB(1) - COS(INCL) / 2 / FREC
  BB(2) = RIAL / FREC / LUCE * SIN(INCL)
  BB(2) = BB(2) + COS(INCL) / FREC
  BB(3) = (-RIAL / 2 / FREC / LUCE + 1 / LUCE) * SIN(INCL))
  BB(3) = BB(3) - COS(INCL) / 2 / FREC
CASE "TAGLIO"
  REM Formula 5.44
  BB(1) = (RIAL / 2 / FREC / LUCE - 1 / LUCE) * COS(INCL)
  BB(1) = BB(1) - SIN(INCL) / 2 / FREC
  BB(2) = -RIAL / FREC / LUCE * COS(INCL)
  BB(2) = BB(2) + SIN(INCL) / FREC
  BB(3) = ((RIAL / 2 / FREC / LUCE + 1 / LUCE) * COS(INCL))
  BB(3) = BB(3) - SIN(INCL) / 2 / FREC
CASE ELSE
  CLS
  PRINT "SCELTA NON VALIDA PER LA LINEA DI INFLUENZA"
  STOP
END SELECT
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 *****
FOR ITER = 1 TO ITER1
  PRINT #ITER, " ===== "

```

```

PRINT #ITER, "PROGRAMMA AVL - LINEE DI INFLUENZA"
PRINT #ITER, TITLE$
PRINT #ITER, "LINEA DI INFLUENZA DEL "; LINF$
PRINT #ITER, " ====="
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, "MODULO DI POISSON ="; NU
PRINT #ITER, "NUMERO DI FORZE DEL TRENO ="; NFOR
PRINT #ITER, "RAPPORTO TRA PASSO DELLE STILATE E PASSO DEL
TRENO ="; KRAP
PRINT #ITER, "NUMERO DI TRATTI TRA LE STILATE = "; TRAT
PRINT #ITER, "NUMERO DI CONCI PER OGNI TRATTO = "; CONC
PRINT #ITER, "NUMERO DI TRATTI COMPLESSIVO = "; T
PRINT #ITER, "LARGHEZZA DELLA SEZIONE RETTA DELL'ARCO A SIN-
ISTRA = "; LARS
PRINT #ITER, "LARGHEZZA DELLA SEZIONE RETTA DELL'ARCO IN CHI-
AVE = "; LARC
PRINT #ITER, "LARGHEZZA DELLA SEZIONE RETTA DELL'ARCO A DE-
STRA = "; LARD
PRINT #ITER, "ALTEZZA DELLA SEZIONE RETTA DELL'ARCO A SINIS-
TRA = "; ALTS
PRINT #ITER, "ALTEZZA DELLA SEZIONE RETTA DELL'ARCO IN CHIAVE
= "; ALTC
PRINT #ITER, "ALTEZZA DELLA SEZIONE RETTA DELL'ARCO A DESTRA
= "; ALTD
PRINT #ITER, "SPESSORE DELLA SOLETTA A SINISTRA = "; SPES
PRINT #ITER, "SPESSORE DELLA SOLETTA AL CENTRO = "; SPEC
PRINT #ITER, "SPESSORE DELLA SOLETTA A DESTRA = "; SPED
PRINT #ITER, "SPESSORE COMPLESSIVO DELLE NERVATURE A SINIS-
TRA = "; SPTS
PRINT #ITER, "SPESSORE COMPLESSIVO DELLE NERVATURE AL CEN-
TRO = "; SPTC
PRINT #ITER, "SPESSORE COMPLESSIVO DELLE NERVATURE A DESTRA
= "; SPTD
PRINT #ITER,
PRINT #ITER, "MATRICE DELLE CEDIBILITA' ELASTICHE A SINISTRA"
PRINT #ITER,
FOR I = 1 TO 3

```



```

    PRINT #ITER, TAB(2); A(I, 1); TAB(20); A(I, 2); TAB(45); A(I, 3)
NEXT I
PRINT #ITER,
PRINT #ITER, "MATRICE DELLE CEDIBILITA' ELASTICHE A DESTRA"
PRINT #ITER,
FOR I = 1 TO 3
    PRINT #ITER, TAB(2); B(I, 1); TAB(20); B(I, 2); TAB(45); B(I, 3)
NEXT I
PRINT #ITER, "VETTORE DELLE FORZE DEL TRENO E LORO DISTANZA
    DA DESTRA"
PRINT #ITER,
FOR I = 1 TO NFOR
    PRINT #ITER, TAB(2); I; TAB(20); TREN(I); TAB(45); DIST(I)
NEXT I
PRINT #ITER,
PRINT #ITER, "ASCISSA DELLA LINEA DI INFLUENZA = "; ZINF
REM -----
REM STAMPA LE QUOTE DELL'ARCO PER CIASCUNA DIVIDENTE
REM -----
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
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 *
REM *      Questa subroutine si occupa dell'uscita (su video, su stampa
REM *      e su file) dei risultati
REM *
REM *****
REM
FOR ITER = ITER1 + 1 TO ITER2
  PRINT #ITER, "LINEA DI INFLUENZA DEL "; LINF$
  PRINT #ITER, "ASCISSA DELLA LINEA DI INFLUENZA = "; ZINF
  PRINT #ITER,
  PRINT #ITER, "AREA AD ORDINATA NEGATIVA ="; ANEG
  PRINT #ITER, "AREA AD ORDINATA POSITIVA ="; APOS
  PRINT #ITER, "VALORE MINIMO ="; LMIN
  PRINT #ITER, "ASCISSA DELLA TESTA DEL TRENO ="; ZMIN
  PRINT #ITER, "VALORE MASSIMO ="; LMAX
  PRINT #ITER, "ASCISSA DELLA TESTA DEL TRENO ="; ZMAX
  PRINT #ITER,
  PRINT #ITER, " ===== "
  PRINT #ITER, " = "
  PRINT #ITER, " =          LINEA DI INFLUENZA          = "
  PRINT #ITER, " = "
  PRINT #ITER, " ===== "
  PRINT #ITER,
  PRINT #ITER, " _____ "
  PRINT #ITER, " ASCISSA          ORDINATA          "
  PRINT #ITER, " _____ "
  PRINT #ITER,
  FOR I = 1 TO TRAT + 1
    PRINT #ITER, TAB(2); I; TAB(10); LDI(I)
  NEXT I
  PRINT #ITER,
  PRINT #ITER, "TEMPO DI ESECUZIONE = "; TIME2; " SEC."
  PRINT #ITER, "RUN DEL "; DATE$; " ORE "; TIME$
NEXT ITER
END SUB

```

Appendice 5.2. Il programma AMAX

```
DECLARE SUB Aree ()
DECLARE SUB Estremi ()
DECLARE SUB LineaDiInfluenza ()
DEFDBL A-Z
DECLARE SUB Caratteristiche ()
DECLARE SUB Cedibilita ()
DECLARE SUB TerminiNoti ()
DECLARE SUB Geometria ()
DECLARE SUB Spostamenti ()
DECLARE SUB Iperstatiche ()
DECLARE SUB slsim (A#(), B#(), X#(), N#)
DECLARE SUB Imposte (A#(), B#(), RA#(), RB#(), PA#(), PB#(), C#)
DECLARE SUB product (A#(), B#(), C#(), N#)
DECLARE SUB dot (A#(), B#(), C#, N#)
DECLARE SUB Plv (MA#(), MB#(), NA#(), NB#(), TA#(), TB#(), C#)
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 M A X          *
REM *
REM *                      di                      *
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. Gennaio 1988 (in BASIC HP) *
REM *          Release 3.0.0. Agosto 1994 (in Microsoft QBasic) *
REM *
REM *****
```

```

REM
REM
REM *****
REM *
REM *      Questo programma studia l'arco in fase elastica tracciando il
REM *      diagramma delle forze interne massime e minime, per effetto di
REM *      un treno di forze verticali viaggianti o di un carico uniforme
REM *
REM *****
OPTION BASE 1
REM
REM *****
REM *
REM *      C O M M O N
REM *
REM *****
REM
COMMON SHARED LUCE, T, E, NU, G, FREC, RIAL, TRAT, CONC, NFOR,
KRAP
COMMON SHARED LARS, LARC, LARD, ALTS, ALTC, ALTD
COMMON SHARED SPES, SPEC, SPED, SPTS, SPTC, SPTD
COMMON SHARED APOS, ANEG, LMIN, LMAX, ZMIN, ZMAX, ZINF
COMMON SHARED DELTA, SMALLDELTA, BIGDELTA
COMMON SHARED U7, W7, K2RAP
COMMON SHARED QUNIF, TR1, SMALL2DELTA, IP
COMMON SHARED ITER1, ITER2, TIME2
COMMON SHARED M() AS DOUBLE, M1() AS DOUBLE
COMMON SHARED M2() AS DOUBLE, M3() AS DOUBLE
COMMON SHARED TG() AS DOUBLE, T1() AS DOUBLE
COMMON SHARED T2() AS DOUBLE, T3() AS DOUBLE
COMMON SHARED N() AS DOUBLE, N1() AS DOUBLE
COMMON SHARED N2() AS DOUBLE, N3() AS DOUBLE
COMMON SHARED X() AS DOUBLE, INCL() AS DOUBLE
COMMON SHARED R0() AS DOUBLE, P0() AS DOUBLE
COMMON SHARED R() AS DOUBLE, P() AS DOUBLE, INERZ() AS DOUBLE
COMMON SHARED R1() AS DOUBLE, R2() AS DOUBLE, R3() AS DOUBLE
COMMON SHARED P1() AS DOUBLE, P2() AS DOUBLE, P3() AS DOUBLE
COMMON SHARED C() AS DOUBLE, AREA() AS DOUBLE
COMMON SHARED CHI() AS DOUBLE, Q() AS DOUBLE
COMMON SHARED TREN() AS DOUBLE, DIST() AS DOUBLE
COMMON SHARED A() AS DOUBLE, B() AS DOUBLE, BB() AS DOUBLE
COMMON SHARED V() AS DOUBLE, PHI() AS DOUBLE, LDI() AS DOUBLE

```

```

COMMON SHARED ZASC() AS DOUBLE
COMMON SHARED EMAX() AS DOUBLE
COMMON SHARED EMIN() AS DOUBLE
COMMON SHARED TITLE$, STAMPA$, STAMPARIS$, FILEINGR$, FILEUSC$
COMMON SHARED ASSE$, FILEINPUT$, LINF$, CARICO$
REM
REM *****
REM *
REM *          DIMENSIONA GLI ARRAY STATICI          *
REM *
REM *
REM *****
REM
DIM TITLE$(100), TITLE1$(100), STAMPA$(10), STAMPARIS$(10)
DIM FILEINGR$(10), FILEUSC$(10), LINF$(30)
DIM R0(3) AS DOUBLE, R1(3) AS DOUBLE, R2(3) AS DOUBLE
DIM P0(3) AS DOUBLE, P1(3) AS DOUBLE, P2(3) AS DOUBLE
DIM P(3) AS DOUBLE, R(3) AS DOUBLE, R3(3) AS DOUBLE
DIM C(3, 3) AS DOUBLE, BB(3) AS DOUBLE, X(3) AS DOUBLE
DIM A(3, 3) AS DOUBLE, B(3, 3) AS DOUBLE, P3(3) AS DOUBLE
REM
REM *****
REM *
REM *          DEFINISCE ALCUNE LEGGI DI VARIAZIONE          *
REM *
REM *
REM *****
REM
DEF FNA (Z) = LARS + (LARD - LARS) / LUCE * Z - 4 * (LARS + (LARD -
    LARS) / 2 - LARC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNB (Z) = ALTS + (ALTD - ALTS) / LUCE * Z - 4 * (ALTS + (ALTD -
    ALTS) / 2 - ALTC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNC (Z) = SPTS + (SPTD - SPTS) / LUCE * Z - 4 * (SPTS + (SPTD -
    SPTS) / 2 - SPTC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNS (Z) = SPES + (SPED - SPES) / LUCE * Z - 4 * (SPES + (SPED -
    SPES) / 2 - SPEC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNQ (Z) = RIAL * Z / LUCE + 4 * FREC / LUCE ^ 2 * Z * (LUCE - Z)
CLS
REM
REM *****
REM *
REM *          LEGGE IL NOME DEL FILE DI INPUT DATI          *
REM *
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
T = TRAT * CONC
DELTA = LUCE / T
BIGDELTA = LUCE / TRAT
SMALLDELTA = BIGDELTA / KRAP
SMALL2DELTA = BIGDELTA / K2RAP
TR1 = TRAT * K2RAP
G = E / (2 * (1 + NU))
REM
REM *****
REM *
REM *          REDIMENSIONA GLI ARRAY DINAMICI
REM *
REM *****
REM
REDIM M(T + 1) AS DOUBLE, M1(T + 1) AS DOUBLE
REDIM M2(T + 1) AS DOUBLE, M3(T + 1) AS DOUBLE
REDIM TG(T + 1) AS DOUBLE, T1(T + 1) AS DOUBLE
REDIM T2(T + 1) AS DOUBLE, T3(T + 1) AS DOUBLE
REDIM N(T + 1) AS DOUBLE, N1(T + 1) AS DOUBLE
REDIM N2(T + 1) AS DOUBLE, N3(T + 1) AS DOUBLE
REDIM TREN(NFOR) AS DOUBLE, DIST(NFOR) AS DOUBLE
REDIM AREA(T + 1) AS DOUBLE, INERZ(T + 1) AS DOUBLE
REDIM CHI(T + 1) AS DOUBLE, Q(T + 1) AS DOUBLE
REDIM INCL(T + 1) AS DOUBLE, LDI(TRAT + 1) AS DOUBLE
REDIM V(T + 1) AS DOUBLE, PHI(T + 1) AS DOUBLE
REDIM EMAX(TRAT + TR1) AS DOUBLE
REDIM EMIN(TRAT + TR1) AS DOUBLE
REDIM ZASC(TRAT + TR1) AS DOUBLE
REM -----
REM CHIAMA LA SUBROUTINE PER L'INGRESSO DEI DATI IN ARRAY
REM -----
CALL IngressoDatiArray
REM -----

```

'cfr. 5.63

```

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
REM LA MATRICE DELLE FLESSIBILITA'
REM -----
CALL Cedibilita
W7 = 0: U7 = 0
ZINF = - SMALL2DELTA
FOR IP = 1 TO TR1 + TRAT
  REM -----
  REM CHIAMA LA SUBROUTINE PER CALCOLARE I TERMINI NOTI
  REM DELLE EQUAZIONI DI CONGRUENZA
  REM -----
  CALL TerminiNoti
  REM -----
  REM CHIAMA LA SUBROUTINE PER CALCOLARE LE IPERSTATICHE
  REM -----
  CALL Iperstatiche
  REM -----
  REM CHIAMA LA SUBROUTINE PER CALCOLARE LE CARATTERISTICHE
  REM -----
  CALL Caratteristiche
  REM -----
  REM CHIAMA LA SUBROUTINE PER CALCOLARE GLI SPOSTAMENTI
  REM -----
  CALL Spostamenti
  REM -----
  REM CHIAMA LA SUBROUTINA PER RIEMPIRE LE LINEE DI INFLUENZA
  REM -----
  CALL LineaDiInfluenza
  IF CARICO$ = "UNIFORME" THEN
    CALL Aree
    EMAX(IP) = QUNIF * APOS
    EMIN(IP) = QUNIF * ANEG

```

```

ELSE
  CALL Estremi
  EMAX(IP) = LMAX
  EMIN(IP) = LMIN
END IF
ZASC(IP) = ZINF
NEXT IP
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 non
REM *           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-
  TAFILES" OR STAMPA$ = "TUTTO" THEN
  INPUT #7, FILEINGR$
END IF
IF STAMPARIS$ = "FILES" OR STAMPARIS$ = "VIDEOFILES" OR STAM-
  PARIS$ = "CARTAFILES" OR STAMPARIS$ = "TUTTO" THEN
  INPUT #7, FILEUSC$
END IF
CALL OpenFiles
REM
REM *****
REM *
REM * LEGGE: *
REM * LUCE Luce dell'arco *
REM * TRAT Numero di tratti tra le stilate *
REM * CONC Numero dei conci per tratto *
REM * NFOR Numero delle forze del treno *
REM * KRAP Rapporto tra la luce del tratto tra le *
REM *          stilate ed il passo di transito del treno *
REM * QUNIF Valore del carico accidentale uniforme *
REM * K2RAP Rapporto tra la luce del tratto tra le *
REM *          stilate ed il passo delle ordinate *
REM * E Modulo di Young *
REM * NU Modulo di Poisson *
REM * LARS Larghezza della sezione retta dell'arco a sinistra *
REM * LARC Larghezza della sezione retta dell'arco al centro *
REM * LARD Larghezza della sezione retta dell'arco a destra *
REM * ALTS Altezza della sezione retta dell'arco a sinistra *
REM * ALTC Altezza della sezione retta dell'arco al centro *
REM * ALTD Altezza della sezione retta dell'arco a destra *
REM * SPES Spessore della soletta superiore o *

```

```

REM *           inferiore della sezione retta a sinistra *
REM *   SPEC   Spessore della soletta superiore o *
REM *           inferiore della sezione retta al centro. *
REM *   SPED   Spessore della soletta superiore o *
REM *           inferiore della sezione retta a destra *
REM *   SPTS   Spessore complessivo dei timpani dell'arco a sinistra *
REM *   SPTC   Spessore complessivo dei timpani dell'arco al centro *
REM *   SPTD   Spessore complessivo dei timpani dell'arco a destra *
REM *   A(3,3) Matrice (3,3) dei cedimenti elastici della *
REM *           imposta di sinistra *
REM *   B(3,3) Matrice (3,3) dei cedimenti elastici della *
REM *           imposta di destra *
REM *   CARICO$ Può assumere i seguenti valori: *
REM *           UNIFORME *
REM *           TRENO *
REM *           per carico uniforme, o per treno *
REM *   INFL$   Può assumere i seguenti valori: *
REM *           MOMENTO BARICENTRICO *
REM *           MOMENTO INTRADOSSALE *
REM *           MOMENTO ESTRADOSSALE *
REM *           SFORZO NORMALE *
REM *           TAGLIO *
REM *   ASSE$   Se l'asse dell'arco e' descritto da una *
REM *           funzione, ASSE$ = FUNZ, se invece e' dato *
REM *           per punti, allora ASSE$ = DATI *
REM *   Se l'asse e' definito da funzione: *
REM *   FREC   Freccia dell'arco *
REM *   RIAL   Rialzo a destra *
REM * *
REM *****
REM
INPUT #7, LUCE, TRAT, CONC, NFOR, KRAP, QUNIF, K2RAP, E, NU
INPUT #7, LARS, LARC, LARD, ALTS, ALTC, ALTD
INPUT #7, SPES, SPEC, SPED, SPTS, SPTC, SPTD
FOR I = 1 TO 3
  FOR J = 1 TO 3
    INPUT #7, A(I, J)
  NEXT J
NEXT I
FOR I = 1 TO 3
  FOR J = 1 TO 3
    INPUT #7, B(I, J)
  
```

```

NEXT J
NEXT I
INPUT #7, CARICO$, LINF$
INPUT #7, ASSE$
IF ASSE$ = "FUNC" THEN
  INPUT #7, FREC, RIAL
END IF
END SUB

SUB TerminiNoti
REM
REM *****
REM *
REM *          T e r m i n i N o t i
REM *
REM *****
REM
REM *****
REM *
REM *          Q U E S T A   S U B R O U T I N E   C A L C O L A   I   T E R M I N I   N O T I
REM *          D E L L E   E Q U A Z I O N I   D I   C O N G R U E N Z A
REM *
REM *****
REM
REM -----
REM Estrazione delle caratteristiche geometriche in ZINF
REM -----
ZINF = ZINF + SMALL2DELTA
U7 = U7 + 1
SELECT CASE W7
CASE IS = 2
  ZINF = ZINF - SMALL2DELTA / 10000
  W7 = 0
  U = INT(ZINF / DELTA) + 1
  ZAUX = (ZINF - (U - 1) * DELTA) / DELTA
  Q = Q(U) + (Q(U + 1) - Q(U)) * ZAUX
  INCL = INCL(U) + (INCL(U + 1) - INCL(U)) * ZAUX
CASE IS = 1
  ZINF = ZINF - SMALL2DELTA + 2 * SMALL2DELTA / 10000
  W7 = 2
  U = INT(ZINF / DELTA) + 1
  ZAUX = (ZINF - (U - 1) * DELTA) / DELTA
  Q = Q(U) + (Q(U + 1) - Q(U)) * ZAUX

```

```

INCL = INCL(U) + (INCL(U + 1) - INCL(U)) * ZAUX
CASE IS = 0
  IF IP = TR1 + TRAT THEN
    U = T + 1
    Q = Q(T + 1)
    INCL = INCL(T + 1)
  ELSE
    IF IP > 1 AND INT((U7 - 1) / K2RAP) = (U7 - 1) / K2RAP THEN
      ZINF = ZINF - SMALL2DELTA / 10000
      W7 = 1
      U7 = U7 - 1
    END IF
    U = INT(ZINF / DELTA) + 1
    ZAUX = (ZINF - (U - 1) * DELTA) / DELTA
    Q = Q(U) + (Q(U + 1) - Q(U)) * ZAUX
    INCL = INCL(U) + (INCL(U + 1) - INCL(U)) * ZAUX
  END IF
END SELECT
B = FNB(ZINF)
REM -----
REM Riempe l'array dei termini noti con i casi opportuni
REM -----
SELECT CASE LINF$
CASE "MOMENTO BARICENTRICO"
  REM Formule 5.22, 5.23, 5.24
  BB(1) = (RIAL / 2 / FREC / LUCE - 1 / LUCE) * ZINF - 1 / 2 / FREC * Q +
  1
  BB(2) = -RIAL / FREC / LUCE * ZINF + Q / FREC
  BB(3) = (RIAL / 2 / FREC + 1 / LUCE) * ZINF - Q / 2 / FREC
CASE "MOMENTO INTRADOSSALE"
  REM Formula 5.53
  BB(1) = (RIAL / 2 / FREC / LUCE - 1 / LUCE) * ZINF - 1 / 2 / FREC * Q +
  1
  BB(1) = BB(1) - B / 2 * (1 / LUCE - RIAL / 2 / FREC / LUCE) * SIN(INCL)
  BB(1) = BB(1) + B / 2 * COS(INCL) / 2 / FREC
  BB(2) = -RIAL / FREC / LUCE * ZINF + Q / FREC
  BB(2) = BB(2) - B / 2 * RIAL / FREC / LUCE * SIN(INCL)
  BB(2) = BB(2) - B / 2 * COS(INCL) / FREC
  BB(3) = (RIAL / 2 / FREC + 1 / LUCE) * ZINF - Q / 2 / FREC
  BB(3) = BB(3) - B / 2 * (-RIAL / 2 / FREC / LUCE + 1 / LUCE) * SIN(INCL)
  BB(3) = BB(3) + B / 2 * COS(INCL) / 2 / FREC
CASE "MOMENTO ESTRADOSSALE"

```

```

REM Formula 5.52
BB(1) = (RIAL / 2 / FREC / LUCE - 1 / LUCE) * ZINF - 1 / 2 / FREC * Q +
1
BB(1) = BB(1) + B / 2 * (1 / LUCE - RIAL / 2 / FREC / LUCE) * SIN(INCL)
BB(1) = BB(1) - B / 2 * COS(INCL) / 2 / FREC
BB(2) = -RIAL / FREC / LUCE * ZINF + Q / FREC
BB(2) = BB(2) + B / 2 * RIAL / FREC / LUCE * SIN(INCL)
BB(2) = BB(2) + B / 2 * COS(INCL) / FREC
BB(3) = (RIAL / 2 / FREC + 1 / LUCE) * ZINF - Q / 2 / FREC
BB(3) = BB(3) + B / 2 * (-RIAL / 2 / FREC / LUCE + 1 / LUCE) * SIN(INCL))
BB(3) = BB(3) - B / 2 * COS(INCL) / 2 / FREC
CASE "SFORZO NORMALE"
REM Formula 5.39
BB(1) = (1 / LUCE - RIAL / 2 / FREC / LUCE) * SIN(INCL)
BB(1) = BB(1) - COS(INCL) / 2 / FREC
BB(2) = RIAL / FREC / LUCE * SIN(INCL)
BB(2) = BB(2) + COS(INCL) / FREC
BB(3) = (-RIAL / 2 / FREC / LUCE + 1 / LUCE) * SIN(INCL))
BB(3) = BB(3) - COS(INCL) / 2 / FREC
CASE "TAGLIO"
REM Formula 5.44
BB(1) = (RIAL / 2 / FREC / LUCE - 1 / LUCE) * COS(INCL)
BB(1) = BB(1) - SIN(INCL) / 2 / FREC
BB(2) = -RIAL / FREC / LUCE * COS(INCL)
BB(2) = BB(2) + SIN(INCL) / FREC
BB(3) = ((RIAL / 2 / FREC / LUCE + 1 / LUCE) * COS(INCL))
BB(3) = BB(3) - SIN(INCL) / 2 / FREC
CASE ELSE
CLS
PRINT "SCELTA NON VALIDA PER LA LINEA DI INFLUENZA"
STOP
END SELECT
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 *****
FOR ITER = 1 TO ITER1
  PRINT #ITER, " ===== "
  PRINT #ITER, "PROGRAMMA AMAX — MASSIMI E MINIMI"
  PRINT #ITER, TITLE$
  PRINT #ITER, "LINEA DI INFLUENZA DEL "; LINF$
  PRINT #ITER, " ===== "
  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, "MODULO DI POISSON ="; NU
  PRINT #ITER, "NUMERO DI FORZE DEL TRENO ="; NFOR
  PRINT #ITER, "RAPPORTO TRA PASSO DELLE STILATE E PASSO DEL
    TRENO ="; KRAP
  PRINT #ITER, "RAPPORTO TRA PASSO DELLE STILATE E PASSO DELLE
    ORDINATE ="; K2RAP
  PRINT #ITER, "VALORE DEL CARICO UNIFORME = "; QUNIF
  PRINT #ITER, "NUMERO DI TRATTI TRA LE STILATE = "; TRAT
  PRINT #ITER, "NUMERO DI CONCI PER OGNI TRATTO = "; CONC
  PRINT #ITER, "NUMERO DI TRATTI COMPLESSIVO = "; T
  PRINT #ITER, "LARGHEZZA DELLA SEZIONE RETTA DELL'ARCO A SIN-
    ISTRA = "; LARS
  PRINT #ITER, "LARGHEZZA DELLA SEZIONE RETTA DELL'ARCO IN CHI-
    AVE = "; LARC
  PRINT #ITER, "LARGHEZZA DELLA SEZIONE RETTA DELL'ARCO A DE-
    STRA = "; LARD
  PRINT #ITER, "ALTEZZA DELLA SEZIONE RETTA DELL'ARCO A SINIS-
    TRA = "; ALTS
  PRINT #ITER, "ALTEZZA DELLA SEZIONE RETTA DELL'ARCO IN CHIAVE
    = "; ALTC
  PRINT #ITER, "ALTEZZA DELLA SEZIONE RETTA DELL'ARCO A DESTRA
    = "; ALTD
  PRINT #ITER, "SPESSORE DELLA SOLETTA A SINISTRA = "; SPES
  PRINT #ITER, "SPESSORE DELLA SOLETTA AL CENTRO = "; SPEC

```

```

PRINT #ITER, "SPESSORE DELLA SOLETTA A DESTRA = "; SPED
PRINT #ITER, "SPESSORE COMPLESSIVO DELLE NERVATURE A SINIS-
  TRA = "; SPTS
PRINT #ITER, "SPESSORE COMPLESSIVO DELLE NERVATURE AL CEN-
  TRO = "; SPTC
PRINT #ITER, "SPESSORE COMPLESSIVO DELLE NERVATURE A DESTRA
  = "; SPTD
PRINT #ITER,
PRINT #ITER, "MATRICE DELLE CEDIBILITA' ELASTICHE A SINISTRA"
PRINT #ITER,
FOR I = 1 TO 3
  PRINT #ITER, TAB(2); A(I, 1); TAB(20); A(I, 2); TAB(45); A(I, 3)
NEXT I
PRINT #ITER,
PRINT #ITER, "MATRICE DELLE CEDIBILITA' ELASTICHE A DESTRA"
PRINT #ITER,
FOR I = 1 TO 3
  PRINT #ITER, TAB(2); B(I, 1); TAB(20); B(I, 2); TAB(45); B(I, 3)
NEXT I
PRINT #ITER, "VETTORE DELLE FORZE DEL TRENO E LORO DISTANZA
  DA DESTRA"
PRINT #ITER,
FOR I = 1 TO NFOR
  PRINT #ITER, TAB(2); I; TAB(20); TREN(I); TAB(45); DIST(I)
NEXT I
PRINT #ITER,
REM
REM STAMPA LE QUOTE DELL'ARCO PER CIASCUNA DIVIDENTE
REM
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

```

```
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 *
REM *           Questa subroutine si occupa dell'uscita (su video, su stampa
REM *           e su file) dei risultati
REM *
REM *****
REM
FOR ITER = ITER1 + 1 TO ITER2
  IF CARICO$ = "UNIFORME" THEN
    PRINT #ITER, "STRUTTURA PERCORSO DA CARICO UNIFORME"
  ELSE
    PRINT #ITER, "STRUTTURA PERCORSO DA TRENO DI FORZE"
  END IF
  PRINT #ITER, " _____"
  PRINT #ITER, " ASCISSA           MASSIMI "
  PRINT #ITER, " _____"
  PRINT #ITER, MINIMI
  FOR I = 1 TO TRAT + TR1
    PRINT #ITER, TAB(2); ZASC(I); TAB(20); EMAX(I); TAB(40); EMIN(I)
    IF INT(I / (K2RAP + 1)) = I / (K2RAP + 1) THEN
      PRINT #ITER,
    END IF
  NEXT I
  PRINT #ITER,
  PRINT #ITER, "TEMPO DI ESECUZIONE = "; TIME2; " SEC."
  PRINT #ITER, "RUN DEL "; DATE$; " ORE "; TIME$
NEXT ITER
END SUB
```