# Appendice 11.1

# Matrice di rigidezza per elemento finito TIM7

```
DEFINT I-N
DEFDBL A-H, O-Z
SUB STIFFTIM7
REM
REM Subroutine STIFFTIM7
REM
REM ************************************************************************
REM *                                                                      *
REM *                          S T I F F T I M 7                           *
REM *                                                                      *
REM ************************************************************************
REM
REM ************************************************************************
REM *                                                                      *
REM *     Questa subroutine calcola la matrice globale di rigidezza per travi *
REM *          tozze, usando un elemento finito a 6 gradi di liberta'      *
REM *        derivato, per condensazione statica, dall'elemento TIM7       *
REM *                                                                      *
REM ************************************************************************
REM
DIM MEMDIS(NEVAB) AS DOUBLE, ESTIF(NEVAB, NEVAB) AS DOUBLE
FOR IELEM = 1 TO NELEM
IEND = LNODS(IELEM, 1)
JEND = LNODS(IELEM, 2)
XPROJ = COORD(JEND, 1) - COORD(IEND, 1)
YPROJ = COORD(JEND, 2) - COORD(IEND, 2)
SPAN = SQR(XPROJ * XPROJ + YPROJ * YPROJ)
REM
```

```
REM ************************************************************************
REM *                                                                      *
REM *                  Calcolo matrice elementare di rigidezza             *
REM *                                                                      *
REM ************************************************************************
REM
LPROP = MATN(IELEM)
YOUNG = PROPS(LPROP, 1)
RINERZ = PROPS(LPROP, 2)
AREA = PROPS(LPROP, 3)
RLAME = PROPS(LPROP, 4)
CHI = PROPS(LPROP, 5)
CT = RLAME * AREA / CHI
CC = 10 * YOUNG * RINERZ + CT * SPAN ^ 2
CF = YOUNG * RINERZ
CC2 = CT * CF
ESTIF(1, 1) = 6 / 5 * CT / SPAN - 6 * CT ^ 2 * SPAN / (5 * CC)
ESTIF(2, 1) = CC2 / CC
ESTIF(3, 1) = -5 * CC2 / CC
ESTIF(4, 1) = -ESTIF(1, 1)
ESTIF(5, 1) = ESTIF(2, 1)
ESTIF(6, 1) = ESTIF(3, 1)
ESTIF(2, 2) = CT * SPAN / 8 + CC2 / (12 * CC) * SPAN
ESTIF(3, 2) = CT * SPAN / 8 - 5 * CC2 / (12 * CC) * SPAN
ESTIF(4, 2) = -ESTIF(2, 1)
ESTIF(5, 2) = -CT * SPAN / 24 + CC2 * SPAN / (12 * CC)
ESTIF(6, 2) = -CT * SPAN / 24 - 5 * CC2 * SPAN / (12 * CC)
ESTIF(3, 3) = CT * SPAN / 8 + CF / SPAN + 25 * CC2 * SPAN / (12 * CC)
ESTIF(4, 3) = -ESTIF(3, 1)
ESTIF(5, 3) = -CT * SPAN / 24 - 5 * CC2 * SPAN / (12 * CC)
ESTIF(6, 3) = -CT * SPAN / 24 - CF / SPAN + 25 * CC2 * SPAN / (12 * CC)
ESTIF(4, 4) = ESTIF(1, 1)
ESTIF(5, 4) = -ESTIF(2, 1)
ESTIF(6, 4) = -ESTIF(3, 1)
ESTIF(5, 5) = ESTIF(2, 2)
ESTIF(6, 5) = ESTIF(3, 2)
ESTIF(6, 6) = ESTIF(3, 3)
FOR I = 1 TO NEVAB
FOR J = I + 1 TO NEVAB
ESTIF(I, J) = ESTIF(J, I)
NEXT J
NEXT I
```

```
REM
REM ************************************************************************
REM *                                                                      *
REM *          Assembla la matrice elementare nella matrice globale        *
REM *                                                                      *
REM ************************************************************************
REM
FOR IDOFN = 1 TO NDOFN
MEMDIS(IDOFN) = NODFRE(IEND, IDOFN)
MEMDIS(IDOFN + NDOFN) = NODFRE(JEND, IDOFN)
NEXT IDOFN
FOR IEVAB = 1 TO NEVAB
IF MEMDIS(IEVAB) = 0 THEN 111
FOR JEVAB = 1 TO NEVAB
IF MEMDIS(JEVAB) = 0 THEN 121
NEWCOL = MEMDIS(JEVAB) - MEMDIS(IEVAB) + 1
IF NEWCOL > 0 THEN
GSTIF(MEMDIS(IEVAB), NEWCOL) = GSTIF(MEMDIS(IEVAB), NEWCOL) +
ESTIF(IEVAB, JEVAB )
END IF
121 NEXT JEVAB
111 NEXT IEVAB
NEXT IELEM
END SUB
```

# Appendice 11.2

# Matrice di rigidezza per elemento finito CC1

```
DEFINT I-N
DEFDBL A-H, O-Z
SUB STIFFCC1
REM
REM Subroutine STIFFCC1
REM
REM ************************************************************************
REM *                                                                      *
REM *                              S T I F F C C 1                         *
REM *                                                                      *
REM ************************************************************************
REM
REM ************************************************************************
REM *                                                                      *
REM *      Questa subroutine calcola la matrice globale di rigidezza per travi     *
REM *        tozze, usando un elemento finito a 8 gradi di liberta' e 4 nodi       *
REM *                                                                      *
REM ************************************************************************
REM
DIM MEMDIS(NEVAB) AS DOUBLE, ESTIF(NEVAB, NEVAB) AS DOUBLE
FOR IELEM = 1 TO NELEM
IEND = LNODS(IELEM, 1)
QUART = LNODS(IELEM, 2)
TREQUART = LNODS(IELEM, 3)
JEND = LNODS(IELEM, 4)
XPROJ = COORD(JEND, 1) - COORD(IEND, 1)
YPROJ = COORD(JEND, 2) - COORD(IEND, 2)
SPAN = SQR(XPROJ * XPROJ + YPROJ * YPROJ)
```

```
REM
REM ************************************************************************
REM *                                                                      *
REM *                  Calcolo matrice elementare di rigidezza             *
REM *                                                                      *
REM ************************************************************************
REM
LPROP = MATN(IELEM)
YOUNG = PROPS(LPROP, 1)
RINERZ = PROPS(LPROP, 2)
AREA = PROPS(LPROP, 3)
RLAME = PROPS(LPROP, 4)
CHI = PROPS(LPROP, 5)
CT = RLAME * AREA / CHI / SPAN
CF = YOUNG * RINERZ / SPAN / 135
ESTIF(1, 1) = CT * 647 / 135
ESTIF(2, 1) = -CT * SPAN / 2
ESTIF(3, 1) = -CT * 704 / 135
ESTIF(4, 1) = -92 * CT * SPAN / 135
ESTIF(5, 1) = 64 * CT / 135
ESTIF(6, 1) = 4 * CT * SPAN / 15
ESTIF(7, 1) = -CT * 7 / 135
ESTIF(8, 1) = -CT * 23 / 270 * SPAN
ESTIF(2, 2) = CT * 67 * SPAN ^ 2 / 945 + CF * 647
ESTIF(3, 2) = CT * SPAN * 92 / 135
ESTIF(4, 2) = 2 * CT * SPAN ^ 2 / 315 - CF * 704
ESTIF(5, 2) = -4 * CT * SPAN / 15
ESTIF(6, 2) = -34 * CT * SPAN ^ 2 / 945 + CF * 64
ESTIF(7, 2) = 23 * CT * SPAN / 270
ESTIF(8, 2) = CT * SPAN ^ 2 / 70 - 7 * CF
ESTIF(3, 3) = CT * 1088 / 135
ESTIF(4, 3) = 0
ESTIF(5, 3) = -CT * 448 / 135
ESTIF(6, 3) = -CT * 128 * SPAN / 135
ESTIF(7, 3) = CT * 64 / 135
ESTIF(8, 3) = 4 * CT * SPAN / 15
ESTIF(4, 4) = CT * 352 / 945 * SPAN ^ 2 + CF * 1088
ESTIF(5, 4) = CT * 128 / 135 * SPAN
ESTIF(6, 4) = 32 * CT * SPAN ^ 2 / 315 - CF * 448
ESTIF(7, 4) = -4 * CT * SPAN / 15
ESTIF(8, 4) = -34 * CT * SPAN ^ 2 / 945 + 64 * CF
ESTIF(5, 5) = ESTIF(3, 3)
```

```
ESTIF(6, 5) = ESTIF(4, 3)
ESTIF(7, 5) = -704 * CT / 135
ESTIF(8, 5) = -92 * CT * SPAN / 135
ESTIF(6, 6) = ESTIF(4, 4)
ESTIF(7, 6) = 92 * CT * SPAN / 135
ESTIF(8, 6) = 2 * CT * SPAN ^ 2 / 315 - 704 * CF
ESTIF(7, 7) = ESTIF(1, 1)
ESTIF(8, 7) = -ESTIF(2, 1)
ESTIF(8, 8) = ESTIF(2, 2)
FOR I = 1 TO NEVAB
FOR J = I + 1 TO NEVAB
ESTIF(I, J) = ESTIF(J, I)
NEXT J
NEXT I
REM
REM ************************************************************************
REM *                                                                      *
REM *          Assembla la matrice elementare nella matrice globale        *
REM *                                                                      *
REM ************************************************************************
REM
FOR IDOFN = 1 TO NDOFN
MEMDIS(IDOFN) = NODFRE(IEND, IDOFN)
MEMDIS(IDOFN + NDOFN) = NODFRE(QUART, IDOFN)
MEMDIS(IDOFN + 2 * NDOFN) = NODFRE(TREQUART, IDOFN)
MEMDIS(IDOFN + 3 * NDOFN) = NODFRE(JEND, IDOFN)
NEXT IDOFN
FOR IEVAB = 1 TO NEVAB
IF MEMDIS(IEVAB) = 0 THEN 111
FOR JEVAB = 1 TO NEVAB
IF MEMDIS(JEVAB) = 0 THEN 121
NEWCOL = MEMDIS(JEVAB) - MEMDIS(IEVAB) + 1
IF NEWCOL > 0 THEN
GSTIF(MEMDIS(IEVAB), NEWCOL) = GSTIF(MEMDIS(IEVAB), NEWCOL) +
ESTIF(IEVAB, JEVAB )
END IF
121 NEXT JEVAB
111 NEXT IEVAB
NEXT IELEM
END SUB
```

# Appendice 11.3

# Matrice di rigidezza per elemento finito CC2

```
DEFINT I-N
DEFDBL A-H, O-Z
SUB STIFFCC2
REM
REM Subroutine STIFFCC2
REM
REM ************************************************************************
REM *                                                                      *
REM *                              S T I F F C C 2                         *
REM *                                                                      *
REM ************************************************************************
REM
REM ************************************************************************
REM *                                                                      *
REM *      Questa subroutine calcola la matrice globale di rigidezza per travi     *
REM *        tozze, usando un elemento finito a 8 gradi di liberta' e 2 nodi       *
REM *                                                                      *
REM ************************************************************************
REM
DIM MEMDIS(NEVAB) AS DOUBLE, ESTIF(NEVAB, NEVAB) AS DOUBLE
FOR IELEM = 1 TO NELEM
IEND = LNODS(IELEM, 1)
JEND = LNODS(IELEM, 2)
XPROJ = COORD(JEND, 1) - COORD(IEND, 1)
YPROJ = COORD(JEND, 2) - COORD(IEND, 2)
SPAN = SQR(XPROJ * XPROJ + YPROJ * YPROJ)
REM
REM ************************************************************************
```

```
REM *                                                                    *
REM *              Calcolo matrice elementare di rigidezza               *
REM *                                                                    *
REM ************************************************************************
REM
LPROP = MATN(IELEM)
YOUNG = PROPS(LPROP, 1)
RINERZ = PROPS(LPROP, 2)
AREA = PROPS(LPROP, 3)
RLAME = PROPS(LPROP, 4)
CHI = PROPS(LPROP, 5)
CT = RLAME * AREA / CHI / SPAN
CF = YOUNG * RINERZ / SPAN
ESTIF(1, 1) = CT * 6 / 5
ESTIF(2, 1) = CT * SPAN / 10
ESTIF(3, 1) = -CT * SPAN / 2
ESTIF(4, 1) = -CT * SPAN ^ 2 / 10
ESTIF(5, 1) = -ESTIF(1, 1)
ESTIF(6, 1) = ESTIF(2, 1)
ESTIF(7, 1) = ESTIF(3, 1)
ESTIF(8, 1) = -ESTIF(4, 1)
ESTIF(2, 2) = CT * 2 * SPAN ^ 2 / 15
ESTIF(3, 2) = CT * SPAN ^ 2 / 10
ESTIF(4, 2) = 0
ESTIF(5, 2) = -CT * SPAN / 10
ESTIF(6, 2) = -CT * SPAN ^ 2 / 30
ESTIF(7, 2) = -CT * SPAN ^ 2 / 10
ESTIF(8, 2) = CT * SPAN ^ 3 / 60
ESTIF(3, 3) = CT * 13 * SPAN ^ 2 / 35 + CF * 6 / 5
ESTIF(4, 3) = 11 * CT * SPAN ^ 3 / 210 + CF / 10 * SPAN
ESTIF(5, 3) = CT * SPAN / 2
ESTIF(6, 3) = -CT * SPAN ^ 2 / 10
ESTIF(7, 3) = CT * 9 * SPAN ^ 2 / 70 - CF * 6 / 5
ESTIF(8, 3) = -CT * 13 * SPAN ^ 3 / 420 + CF / 10 * SPAN
ESTIF(4, 4) = CT * SPAN ^ 4 / 105 + 2 * CF * SPAN ^ 2 / 15
ESTIF(5, 4) = CT * SPAN ^ 2 / 10
ESTIF(6, 4) = -CT * SPAN ^ 3 / 60
ESTIF(7, 4) = 13 * CT * SPAN ^ 3 / 420 - CF * SPAN / 10
ESTIF(8, 4) = -CT * SPAN ^ 4 / 140 - CF * SPAN ^ 2 / 30
ESTIF(5, 5) = ESTIF(1, 1)
ESTIF(6, 5) = -ESTIF(2, 1)
ESTIF(7, 5) = -ESTIF(3, 1)
```

```
ESTIF(8, 5) = ESTIF(4, 1)
ESTIF(6, 6) = ESTIF(2, 2)
ESTIF(7, 6) = ESTIF(3, 2)
ESTIF(8, 6) = ESTIF(4, 2)
ESTIF(7, 7) = ESTIF(3, 3)
ESTIF(8, 7) = -ESTIF(4, 3)
ESTIF(8, 8) = ESTIF(4, 4)
FOR I = 1 TO NEVAB
FOR J = I + 1 TO NEVAB
ESTIF(I, J) = ESTIF(J, I)
NEXT J
NEXT I
REM
REM ************************************************************************
REM *                                                                      *
REM *          Assembla la matrice elementare nella matrice globale        *
REM *                                                                      *
REM ************************************************************************
REM
FOR IDOFN = 1 TO NDOFN
MEMDIS(IDOFN) = NODFRE(IEND, IDOFN)
MEMDIS(IDOFN + NDOFN) = NODFRE(JEND, IDOFN)
NEXT IDOFN
FOR IEVAB = 1 TO NEVAB
IF MEMDIS(IEVAB) = 0 THEN 111
FOR JEVAB = 1 TO NEVAB
IF MEMDIS(JEVAB) = 0 THEN 121
NEWCOL = MEMDIS(JEVAB) - MEMDIS(IEVAB) + 1
IF NEWCOL > 0 THEN
GSTIF(MEMDIS(IEVAB), NEWCOL) = GSTIF(MEMDIS(IEVAB), NEWCOL) +
ESTIF(IEVAB, JEVAB)
END IF
121 NEXT JEVAB
111 NEXT IEVAB
NEXT IELEM
END SUB
```

# Appendice 11.4

# Matrice di rigidezza per elemento finito QQ1

```
DEFINT I-N
DEFDBL A-H, O-Z
SUB STIFFQQ1
REM
REM Subroutine STIFFQQ1
REM
REM ************************************************************************
REM *                                                                      *
REM *                          S T I F F Q Q 1                             *
REM *                                                                      *
REM ************************************************************************
REM
REM ************************************************************************
REM *                                                                      *
REM *     Questa subroutine calcola la matrice globale di rigidezza per travi *
REM *       tozze, usando un elemento finito a 12 gradi di liberta' e 3 nodi *
REM *                                                                      *
REM ************************************************************************
REM
DIM MEMDIS(NEVAB) AS DOUBLE, ESTIF(NEVAB, NEVAB) AS DOUBLE
FOR IELEM = 1 TO NELEM
IEND = LNODS(IELEM, 1)
MEDIUM = LNODS(IELEM, 2)
JEND = LNODS(IELEM, 3)
XPROJ = COORD(JEND, 1) - COORD(IEND, 1)
YPROJ = COORD(JEND, 2) - COORD(IEND, 2)
SPAN = SQR(XPROJ * XPROJ + YPROJ * YPROJ)
REM
```

```
REM ************************************************************************
REM *                                                                      *
REM *                  Calcolo matrice elementare di rigidezza             *
REM *                                                                      *
REM ************************************************************************
REM
LPROP = MATN(IELEM)
YOUNG = PROPS(LPROP, 1)
RINERZ = PROPS(LPROP, 2)
AREA = PROPS(LPROP, 3)
RLAME = PROPS(LPROP, 4)
CHI = PROPS(LPROP, 5)
CT = RLAME * AREA / CHI / SPAN
CF = YOUNG * RINERZ / SPAN
ESTIF(1, 1) = CT * 278 / 105
ESTIF(2, 1) = 13 * CT * SPAN / 210
ESTIF(3, 1) = -CT * SPAN / 2
ESTIF(4, 1) = -23 * CT * SPAN ^ 2 / 630
ESTIF(5, 1) = -CT * 256 / 105
ESTIF(6, 1) = 8 * CT * SPAN / 21
ESTIF(7, 1) = -8 * CT * SPAN / 21
ESTIF(8, 1) = 32 * CT * SPAN ^ 2 / 315
ESTIF(9, 1) = -22 * CT / 105
ESTIF(10, 1) = -CT * SPAN / 70
ESTIF(11, 1) = -5 * CT * SPAN / 42
ESTIF(12, 1) = CT * SPAN ^ 2 / 90
ESTIF(2, 2) = CT * 2 * SPAN ^ 2 / 45
ESTIF(3, 2) = 23 * CT * SPAN ^ 2 / 630
ESTIF(4, 2) = 0
ESTIF(5, 2) = -CT * SPAN / 105 * 8
ESTIF(6, 2) = -CT * SPAN ^ 2 / 315 * 4
ESTIF(7, 2) = -CT * 8 * SPAN ^ 2 / 315
ESTIF(8, 2) = 2 * CT * SPAN ^ 3 / 315
ESTIF(9, 2) = SPAN * CT / 70
ESTIF(10, 2) = -CT * SPAN ^ 2 / 126
ESTIF(11, 2) = -CT * SPAN ^ 2 / 90
ESTIF(12, 2) = CT * SPAN ^ 3 / 1260
ESTIF(3, 3) = 523 * CT * SPAN ^ 2 / 3465 + CF * 278 / 105
ESTIF(4, 3) = 19 * CT * SPAN ^ 3 / 2310 + CF * SPAN * 13 / 210
ESTIF(5, 3) = CT * 8 / 21 * SPAN
ESTIF(6, 3) = -32 * CT * SPAN ^ 2 / 315
ESTIF(7, 3) = 4 * CT * SPAN ^ 2 / 63 - 256 * CF / 105
```

```
ESTIF(8, 3) = -8 * CT * SPAN ^ 3 / 693 + 8 * CF * SPAN / 21
ESTIF(9, 3) = 5 * CT * SPAN / 42
ESTIF(10, 3) = -CT * SPAN ^ 2 / 90
ESTIF(11, 3) = 131 * CT * SPAN ^ 2 / 6930 - 22 * CF / 105
ESTIF(12, 3) = -29 * CT * SPAN ^ 3 / 13860 - CF * SPAN / 70
ESTIF(4, 4) = 2 * CT * SPAN ^ 4 / 3465 + 2 * CF * SPAN ^ 2 / 45
ESTIF(5, 4) = 8 * CT * SPAN ^ 2 / 315
ESTIF(6, 4) = -2 * CT * SPAN ^ 3 / 315
ESTIF(7, 4) = 2 * CT * SPAN ^ 3 / 315 - 8 * SPAN * CF / 105
ESTIF(8, 4) = -CT * SPAN ^ 4 / 1155 - 4 * CF * SPAN ^ 2 / 315
ESTIF(9, 4) = CT * SPAN ^ 2 / 90
ESTIF(10, 4) = -CT * SPAN ^ 3 / 1260
ESTIF(11, 4) = 29 * CT * SPAN ^ 3 / 13860 + CF * SPAN / 70
ESTIF(12, 4) = -CT * SPAN ^ 4 / 4620 - CF * SPAN ^ 2 / 126
ESTIF(5, 5) = CT * 512 / 105
ESTIF(6, 5) = 0
ESTIF(7, 5) = 0
ESTIF(8, 5) = -64 * CT * SPAN ^ 2 / 315
ESTIF(9, 5) = -CT * 256 / 105
ESTIF(10, 5) = 8 * CT * SPAN / 105
ESTIF(11, 5) = -8 * CT * SPAN / 21
ESTIF(12, 5) = 8 * CT * SPAN ^ 2 / 315
ESTIF(6, 6) = 128 * CT * SPAN ^ 2 / 315
ESTIF(7, 6) = 64 * CT * SPAN ^ 2 / 315
ESTIF(8, 6) = 0
ESTIF(9, 6) = -8 * CT * SPAN / 21
ESTIF(10, 6) = -4 * CT * SPAN ^ 2 / 315
ESTIF(11, 6) = -32 * CT * SPAN ^ 2 / 315
ESTIF(12, 6) = 2 * CT * SPAN ^ 3 / 315
ESTIF(7, 7) = 128 * CT * SPAN ^ 2 / 315 + CF * 512 / 105
ESTIF(8, 7) = 0
ESTIF(9, 7) = 8 * CT * SPAN / 21
ESTIF(10, 7) = -8 * CT * SPAN ^ 2 / 315
ESTIF(11, 7) = 4 * CT * SPAN ^ 2 / 63 - CF * 256 / 105
ESTIF(12, 7) = -2 * CT * SPAN ^ 3 / 315 + CF * 8 * SPAN / 105
ESTIF(8, 8) = 32 * CT * SPAN ^ 4 / 3465 + CF * 128 * SPAN ^ 2 / 315
ESTIF(9, 8) = 32 * CT * SPAN ^ 2 / 315
ESTIF(10, 8) = -2 * CT * SPAN ^ 3 / 315
ESTIF(11, 8) = 8 * CT * SPAN ^ 3 / 693 - 8 * CF * SPAN / 21
ESTIF(12, 8) = -CT * SPAN ^ 4 / 1155 - 4 * CF * SPAN ^ 2 / 315
ESTIF(9, 9) = ESTIF(1, 1)
ESTIF(10, 9) = -ESTIF(2, 1)
```

```
ESTIF(11, 9) = -ESTIF(3, 1)
ESTIF(12, 9) = ESTIF(4, 1)
ESTIF(10, 10) = ESTIF(2, 2)
ESTIF(11, 10) = ESTIF(3, 2)
ESTIF(12, 10) = ESTIF(4, 2)
ESTIF(11, 11) = ESTIF(3, 3)
ESTIF(12, 11) = -ESTIF(4, 3)
ESTIF(12, 12) = ESTIF(4, 4)
FOR I = 1 TO NEVAB
FOR J = I + 1 TO NEVAB
ESTIF(I, J) = ESTIF(J, I)
NEXT J
NEXT I
REM
REM ************************************************************************
REM *                                                                      *
REM *          Assembla la matrice elementare nella matrice globale        *
REM *                                                                      *
REM ************************************************************************
REM
FOR IDOFN = 1 TO NDOFN
MEMDIS(IDOFN) = NODFRE(IEND, IDOFN)
MEMDIS(IDOFN + NDOFN) = NODFRE(MEDIUM, IDOFN)
MEMDIS(IDOFN + 2 * NDOFN) = NODFRE(JEND, IDOFN)
NEXT IDOFN
FOR IEVAB = 1 TO NEVAB
IF MEMDIS(IEVAB) = 0 THEN 111
FOR JEVAB = 1 TO NEVAB
IF MEMDIS(JEVAB) = 0 THEN 121
NEWCOL = MEMDIS(JEVAB) - MEMDIS(IEVAB) + 1
IF NEWCOL > 0 THEN
GSTIF(MEMDIS(IEVAB), NEWCOL) = GSTIF(MEMDIS(IEVAB), NEWCOL) +
ESTIF(IEVAB, JEVAB)
END IF
121 NEXT JEVAB
111 NEXT IEVAB
NEXT IELEM
END SUB
```

# Appendice 11.5

# Matrice di rigidezza per elemento finito QQ2

```
DEFINT I-N
DEFDBL A-H, O-Z
SUB STIFFQQ2
REM
REM Subroutine STIFFQQ2
REM
REM ************************************************************************
REM *                                                                      *
REM *                              S T I F F Q Q 2                         *
REM *                                                                      *
REM ************************************************************************
REM
REM ************************************************************************
REM *                                                                      *
REM *      Questa subroutine calcola la matrice globale di rigidezza per travi     *
REM *        tozze, usando un elemento finito a 12 gradi di liberta' e 2 nodi      *
REM *                                                                      *
REM ************************************************************************
REM
DIM MEMDIS(NEVAB) AS DOUBLE, ESTIF(NEVAB, NEVAB) AS DOUBLE
FOR IELEM = 1 TO NELEM
REM
REM ************************************************************************
REM *                                                                      *
REM *                    Calcolo matrice elementare di rigidezza           *
REM *                                                                      *
REM ************************************************************************
REM
```

```
IEND = LNODS(IELEM, 1)
JEND = LNODS(IELEM, 2)
XPROJ = COORD(JEND, 1) - COORD(IEND, 1)
YPROJ = COORD(JEND, 2) - COORD(IEND, 2)
SPAN = SQR(XPROJ * XPROJ + YPROJ * YPROJ)
LPROP = MATN(IELEM)
YOUNG = PROPS(LPROP, 1)
RINERZ = PROPS(LPROP, 2)
AREA = PROPS(LPROP, 3)
RLAME = PROPS(LPROP, 4)
CHI = PROPS(LPROP, 5)
CT = RLAME * AREA / CHI / SPAN
CF = YOUNG * RINERZ / SPAN
ESTIF(1, 1) = CT * 10 / 7
ESTIF(2, 1) = CT * 3 * SPAN / 14
ESTIF(3, 1) = CT * SPAN ^ 2 / 84
ESTIF(4, 1) = -CT * SPAN / 2
ESTIF(5, 1) = -CT * 11 * SPAN ^ 2 / 84
ESTIF(6, 1) = -CT * SPAN ^ 3 / 84
ESTIF(7, 1) = -ESTIF(1, 1)
ESTIF(8, 1) = ESTIF(2, 1)
ESTIF(9, 1) = -ESTIF(3, 1)
ESTIF(10, 1) = ESTIF(4, 1)
ESTIF(11, 1) = -ESTIF(5, 1)
ESTIF(12, 1) = ESTIF(6, 1)
ESTIF(2, 2) = 8 * CT * SPAN ^ 2 / 35
ESTIF(3, 2) = CT * SPAN ^ 3 / 60
ESTIF(4, 2) = CT * 11 * SPAN ^ 2 / 84
ESTIF(5, 2) = 0
ESTIF(6, 2) = -CT * SPAN ^ 4 / 1008
ESTIF(7, 2) = -CT * SPAN / 14 * 3
ESTIF(8, 2) = -CT * SPAN ^ 2 / 70
ESTIF(9, 2) = CT * SPAN ^ 3 / 210
ESTIF(10, 2) = -11 * CT * SPAN ^ 2 / 84
ESTIF(11, 2) = 13 * CT * SPAN ^ 3 / 420
ESTIF(12, 2) = -13 * CT * SPAN ^ 4 / 5040
ESTIF(3, 3) = CT * SPAN ^ 4 / 630
ESTIF(4, 3) = CT * SPAN ^ 3 / 84
ESTIF(5, 3) = CT * SPAN ^ 4 / 1008
ESTIF(6, 3) = 0
ESTIF(7, 3) = -CT * SPAN ^ 2 / 84
ESTIF(8, 3) = -CT * SPAN ^ 3 / 210
```

ESTIF(9, 3) = CT * SPAN ^ 4 / 1260
ESTIF(10, 3) = -CT * SPAN ^ 3 / 84
ESTIF(11, 3) = CT * 13 * SPAN ^ 4 / 5040
ESTIF(12, 3) = -CT * SPAN ^ 5 / 5040
ESTIF(4, 4) = 181 * CT * SPAN ^ 2 / 462 + 10 * CF / 7
ESTIF(5, 4) = 311 * CT * SPAN ^ 3 / 4620 + 3 * CF * SPAN / 14
ESTIF(6, 4) = 281 * CT * SPAN ^ 4 / 55440 + CF * SPAN ^ 2 / 84
ESTIF(7, 4) = CT * SPAN / 2
ESTIF(8, 4) = -CT * SPAN ^ 2 * 11 / 84
ESTIF(9, 4) = CT * SPAN ^ 3 / 84
ESTIF(10, 4) = 25 * CT * SPAN ^ 2 / 231 - 10 * CF / 7
ESTIF(11, 4) = -151 * CT * SPAN ^ 3 / 4620 + 3 * SPAN * CF / 14
ESTIF(12, 4) = 181 * CT * SPAN ^ 4 / 55440 - CF * SPAN ^ 2 / 84
ESTIF(5, 5) = 52 * CT * SPAN ^ 4 / 3465 + CF * 8 * SPAN ^ 2 / 35
ESTIF(6, 5) = 23 * CT * SPAN ^ 5 / 18480 + CF * SPAN ^ 3 / 60
ESTIF(7, 5) = 11 * CT * SPAN ^ 2 / 84
ESTIF(8, 5) = -13 * CT * SPAN ^ 3 / 420
ESTIF(9, 5) = 13 * CT * SPAN ^ 4 / 5040
ESTIF(10, 5) = 151 * CT * SPAN ^ 3 / 4620 - 3 * CF * SPAN / 14
ESTIF(11, 5) = -19 * CT * SPAN ^ 4 / 1980 - CF * SPAN ^ 2 / 70
ESTIF(12, 5) = 13 * CT * SPAN ^ 5 / 13860 + CF * SPAN ^ 3 / 210
ESTIF(6, 6) = CT * SPAN ^ 6 / 9240 + CF * SPAN ^ 4 / 630
ESTIF(7, 6) = CT * SPAN ^ 3 / 84
ESTIF(8, 6) = -13 * CT * SPAN ^ 4 / 5040
ESTIF(9, 6) = CT * SPAN ^ 5 / 5040
ESTIF(10, 6) = 181 * CT * SPAN ^ 4 / 55440 - CF * SPAN ^ 2 / 84
ESTIF(11, 6) = -13 * CT * SPAN ^ 5 / 13860 - CF * SPAN ^ 3 / 210
ESTIF(12, 6) = CT * SPAN ^ 6 / 11088 + CF * SPAN ^ 4 / 1260
ESTIF(7, 7) = ESTIF(1, 1)
ESTIF(8, 7) = -ESTIF(2, 1)
ESTIF(9, 7) = ESTIF(3, 1)
ESTIF(10, 7) = -ESTIF(4, 1)
ESTIF(11, 7) = ESTIF(5, 1)
ESTIF(12, 7) = -ESTIF(6, 1)
ESTIF(8, 8) = ESTIF(2, 2)
ESTIF(9, 8) = -ESTIF(3, 2)
ESTIF(10, 8) = ESTIF(4, 2)
ESTIF(11, 8) = -ESTIF(5, 2)
ESTIF(12, 8) = ESTIF(6, 2)
ESTIF(9, 9) = ESTIF(3, 3)
ESTIF(10, 9) = -ESTIF(4, 3)
ESTIF(11, 9) = ESTIF(5, 3)

```
ESTIF(12, 9) = -ESTIF(6, 3)
ESTIF(10, 10) = ESTIF(4, 4)
ESTIF(11, 10) = -ESTIF(5, 4)
ESTIF(12, 10) = ESTIF(6, 4)
ESTIF(11, 11) = ESTIF(5, 5)
ESTIF(12, 11) = -ESTIF(6, 5)
ESTIF(12, 12) = ESTIF(6, 6)
FOR I = 1 TO NEVAB
FOR J = I + 1 TO NEVAB
ESTIF(I, J) = ESTIF(J, I)
NEXT J
NEXT I
REM
REM ************************************************************************
REM *                                                                      *
REM *          Assembla la matrice elementare nella matrice globale        *
REM *                                                                      *
REM ************************************************************************
REM
FOR IDOFN = 1 TO NDOFN
MEMDIS(IDOFN) = NODFRE(IEND, IDOFN)
MEMDIS(IDOFN + NDOFN) = NODFRE(JEND, IDOFN)
NEXT IDOFN
FOR IEVAB = 1 TO NEVAB
IF MEMDIS(IEVAB) = 0 THEN 111
FOR JEVAB = 1 TO NEVAB
IF MEMDIS(JEVAB) = 0 THEN 121
NEWCOL = MEMDIS(JEVAB) - MEMDIS(IEVAB) + 1
IF NEWCOL > 0 THEN
GSTIF(MEMDIS(IEVAB), NEWCOL) = GSTIF(MEMDIS(IEVAB), NEWCOL) +
ESTIF(IEVAB, JEVAB)
END IF
121 NEXT JEVAB
111 NEXT IEVAB
NEXT IELEM
END SUB
```

# Appendice 11.6

# Matrice di rigidezza per elemento finito di Dawe

```
DEFDBL A-Z
SUB STIFFV5
REM
REM Subroutine STIFFV5
REM
REM ************************************************************************
REM *                                                                      *
REM *                              S T I F F V 5                           *
REM *                                                                      *
REM ************************************************************************
REM
REM ************************************************************************
REM *                                                                      *
REM *      Questa subroutine calcola la matrice globale di rigidezza per travi     *
REM *        tozze, usando un elemento finito a 6 gradi di liberta' vincolato      *
REM *                                                                      *
REM ************************************************************************
REM
DIM MEMDIS(6) AS DOUBLE, ESTIF(6, 6) AS DOUBLE
FOR IELEM = 1 TO NELEM
IEND = LNODS(IELEM, 1)
MEDIUM = LNODS(IELEM, 2)
JEND = LNODS(IELEM, 3)
XPROJ = COORD(JEND, 1) - COORD(IEND, 1)
YPROJ = COORD(JEND, 2) - COORD(IEND, 2)
SPAN = SQR(XPROJ * XPROJ + YPROJ * YPROJ)
REM
REM ************************************************************************
```

```
REM *                                                                    *
REM *                Calcolo matrice elementare di rigidezza             *
REM *                                                                    *
REM ************************************************************************
REM
LPROP = MATN(IELEM)
YOUNG = PROPS(LPROP, 1)
RINERZ = PROPS(LPROP, 2)
AREA = PROPS(LPROP, 3)
RLAME = PROPS(LPROP, 4)
CHI = PROPS(LPROP, 5)
E = CHI * YOUNG * RINERZ / AREA / RLAME / SPAN ^ 2
D1 = (48 * E + 1) ^ 2
D2 = (60 * E + 1) ^ 2
EI = YOUNG * RINERZ
GA = AREA * RLAME / CHI * SPAN ^ 2
CC = 1273 * EI + 132960 * E * EI + 3513600 * E ^ 2 * EI
CC = CC + 159180 * E ^ 2 * GA + 15926400 * E ^ 3 * GA
CC = CC + 401587200 * E ^ 4 * GA
ESTIF(1, 1) = 4 * CC / (35 * D1 * D2 * SPAN ^ 3)
CC = -569 * EI - 57600 * E * EI - 1370880 * E ^ 2 * EI + 5529600 * E ^ 3 * EI
CC = CC - 61740 * E ^ 2 * GA - 5443200 * E ^ 3 * GA - 82252800 * E ^ 4 * GA
CC = CC + 1857945600 * E ^ 5 * GA
ESTIF(2, 1) = 2 * CC / (35 * D1 * D2 * SPAN ^ 2)
CC = EI + 60 * E ^ 2 * GA
ESTIF(3, 1) = -512 * CC / (5 * D1 * SPAN ^ 3)
CC = EI + 5 * E * EI + 175 * E ^ 2 * GA + 1680 * E ^ 3 * GA
ESTIF(4, 1) = -384 * CC / (7 * D2 * SPAN ^ 2)
CC = 377 * EI + 25440 * E * EI + 288000 * E ^ 2 * EI + 105420 * E ^ 2 * GA
CC = CC + 9475200 * E ^ 3 * GA + 208051200 * E ^ 4 * GA
ESTIF(5, 1) = -4 * CC / (35 * D1 * D2 * SPAN ^ 3)
CC = -121 * EI - 3840 * E * EI + 241920 * E ^ 2 * EI + 5529600 * E ^ 3 * EI
CC = CC - 34860 * E ^ 2 * GA - 2217600 * E ^ 3 * GA + 14515200 * E ^ 4 * GA
CC = CC + 1857945600 * E ^ 5 * GA
ESTIF(6, 1) = 2 * CC / (35 * D1 * D2 * SPAN ^ 2)
CC = 83 * EI + 10410 * E * EI + 467340 * E ^ 2 * EI + 10788480 * E ^ 3 * EI
CC = CC + 161049600 * E ^ 4 * EI + 6195 * E ^ 2 * GA + 500640 * E ^ 3 * GA
CC = CC + 5160960 * E ^ 4 * GA - 154828800 * E ^ 5 * GA
CC = CC + 3715891200 * E ^ 6 * GA
ESTIF(2, 2) = 4 * CC / (35 * SPAN * D1 * D2)
CC = EI + 60 * E ^ 2 * GA
ESTIF(3, 2) = 128 * CC / (5 * D1 * SPAN ^ 2)
```

```
CC = EI - 33 * E * EI - 960 * E ^ 2 * EI + 189 * E ^ 2 * GA
CC = CC - 1680 * E ^ 3 * GA - 40320 * E ^ 4 * GA
ESTIF(4, 2) = 64 * CC / (7 * D2 * SPAN)
CC = 121 * EI + 3840 * E * EI - 241920 * E ^ 2 * EI - 5529600 * E ^ 3 * EI
CC = CC + 34860 * E ^ 2 * GA + 2217600 * E ^ 3 * GA
CC = CC - 14515200 * E ^ 4 * GA - 1857945600 * E ^ 5 * GA
ESTIF(5, 2) = 2 * CC / (35 * D1 * D2 * SPAN ^ 2)
CC = 19 * EI - 180 * E * EI - 78360 * E ^ 2 * EI - 195840 * E ^ 3 * EI
CC = CC + 31795200 * E ^ 4 * EI + 5670 * E ^ 2 * GA + 194880 * E ^ 3 * GA
CC = CC - 13870080 * E ^ 4 * GA - 309657600 * E ^ 5 * GA
CC = CC + 7431782400 * E ^ 6 * GA
ESTIF(6, 2) = 2 * CC / (35 * SPAN * D1 * D2)
CC = EI + 60 * E ^ 2 * GA
ESTIF(3, 3) = 1024 * CC / (5 * D1 * SPAN ^ 3)
ESTIF(4, 3) = 0
ESTIF(5, 3) = -512 * CC / (5 * D1 * SPAN ^ 3)
ESTIF(6, 3) = -128 * CC / (5 * D1 * SPAN ^ 2)
CC = EI + 24 * E * EI + 480 * E ^ 2 * EI + 168 * E ^ 2 * GA
CC = CC + 3360 * E ^ 3 * GA + 20160 * E ^ 4 * GA
ESTIF(4, 4) = 256 * CC / (7 * D2 * SPAN)
CC = EI + 5 * E * EI + 175 * E ^ 2 * GA + 1680 * E ^ 3 * GA
ESTIF(5, 4) = 384 * CC / (7 * D2 * SPAN ^ 2)
ESTIF(6, 4) = ESTIF(4, 2)
ESTIF(5, 5) = ESTIF(1, 1)
ESTIF(6, 5) = -ESTIF(2, 1)
ESTIF(6, 6) = ESTIF(2, 2)
FOR I = 1 TO NEVAB
FOR J = I + 1 TO NEVAB
ESTIF(I, J) = ESTIF(J, I)
NEXT J
NEXT I
REM
REM ************************************************************************
REM *                                                                      *
REM *          Assembla la matrice elementare nella matrice globale        *
REM *                                                                      *
REM ************************************************************************
REM
REM
FOR IDOFN = 1 TO NDOFN
MEMDIS(IDOFN) = NODFRE(IEND, IDOFN)
MEMDIS(IDOFN + NDOFN) = NODFRE(MEDIUM, IDOFN)
```

```
MEMDIS(IDOFN + 2 * NDOFN) = NODFRE(JEND, IDOFN)
NEXT IDOFN
FOR IEVAB = 1 TO NEVAB
IF MEMDIS(IEVAB) = 0 THEN 111
FOR JEVAB = 1 TO NEVAB
IF MEMDIS(JEVAB) = 0 THEN 121
NEWCOL = MEMDIS(JEVAB) - MEMDIS(IEVAB) + 1
IF NEWCOL > 0 THEN
GSTIF(MEMDIS(IEVAB), NEWCOL) = GSTIF(MEMDIS(IEVAB), NEWCOL) +
ESTIF(IEVAB, JEVAB)
END IF
121 NEXT JEVAB
111 NEXT IEVAB
NEXT IELEM
END SUB
```