排架设计计算通用程序*

中山大学数学力学系力学教研室
广东省建筑設計院計算組

一、前　言

在工业建筑中，厂房排架的结构形式和荷载组合比较复杂，结构计算的工作量很大，尽管目前在实际工作中已采用各种简化计算图表，以现成的公式替代费时的建立方程式的推演，但是多跨多阶厂房排架柱计算仍相当繁琐，我们根据目前生产上的需要，遵照伟大领袖毛主席关于**"我们必须打破常规，尽量采用先进技术，在一个不太长的历史时期内，把我国建设成一个社会主义的现代化的强国。"** 的教导，编制了排架设计计算通用程序，它是用国产 $DJS-21$ 型电子计算机算法语言编写的（以 ALGOL－60 为基础），用以解决平面排架的内力分析、内力组合，并对钢筋混凝土结构进行配筋计算。从目前使用情况来看，它具有适用范围较广、计算速度较快、数据填写较简单等优点。但程序中未考虑柱脚基础部分的计算。对地震水平力的影响及排架空间工作等问题也有待进一步考虑。

二、计 算 方 法

排架计算采用位移法。排架的变阶柱可以人为地分成若干个等截面杆件——单元。

单元的节点位移为

$$\mathbf{V}=[\theta_L,\ u_L,\ v_L,\ \theta_R,\ u_R,\ v_R]^T,$$

相应的杆端对节点的反力为

$$\mathbf{F}=[M_L,\ F_{XL},\ F_{YL},\ M_R,\ F_{XR},\ F_{YR}]^T。$$

其中 $\theta_L,\ u_L,\ v_L,\ \theta_R,\ u_R,\ v_R$ 分别表示杆的左端和右端 的转角、x向位移、y 向位移。

---

* 本稿由陈树坚、梁达鸿整理。在編制本程序时，大連工学院数理力学系提供了帮助。

转角以顺时针方向为正，位移以 x、y 轴之正向为正，如图 2.1所示。M_L, F_{XL}, F_{YL} M_R, F_{XR}, F_{YR} 分别表示杆左端和右端对节点的反作用力矩、x 向反作用力、y向反作用力。力矩以逆时针方向为正，力以 x、y 轴之负向为正。

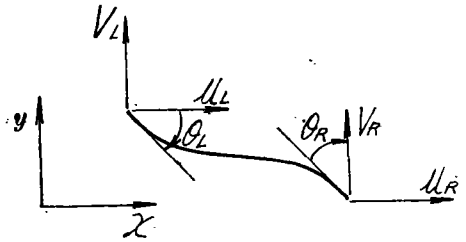


图 2.1

（排架柱一律以近基础端为左端，以上端为右端）

V 与 F 的关系是：

$$F = K_e V,$$

K_e 表示单元刚度矩阵，

$$K_e = \frac{EI}{l}\begin{bmatrix} 4 & & & & & \\ 0 & A/I & & \text{对} & & \\ -6/l & 0 & 12/l^2 & & \text{称} & \\ 2 & 0 & -6/l & 4 & & \\ 0 & -A/I & 0 & 0 & A/I & \\ 6/l & 0 & -12/l^2 & 6/l & 0 & 12/l^2 \end{bmatrix},$$

式中 E, A, I, l 分别是杆件的弹性模数、截面面积、截面惯性矩和长度。

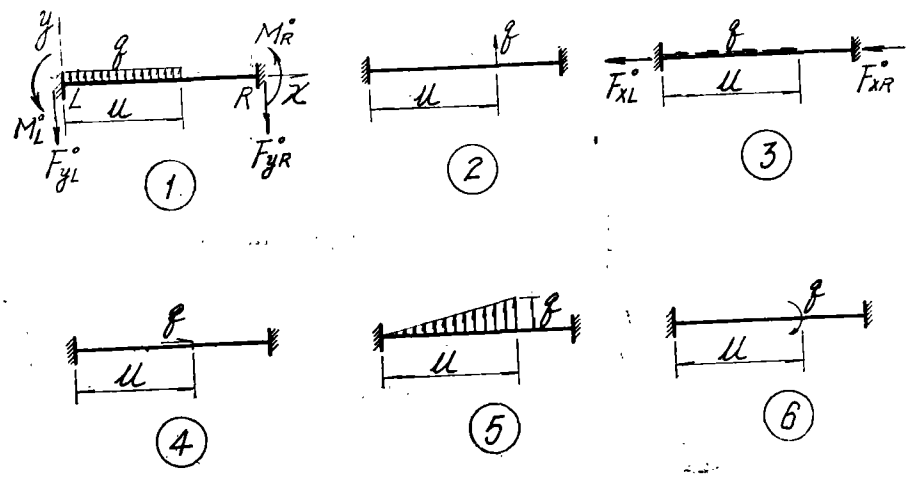


图2.2　兩端固定梁六种荷載簡图。

计算固端反力时分别考虑图2.2中所示的六种荷载情况。图中 M_L°，M_R°分别表示左、右端的固端弯矩；F_X°，F_{XR}°分别表示左、右端的固端轴力；F_{YL}°，F_{YR}°分别表示左、右端的固端剪力。为了程序中迭加时的方便，我们规定弯矩以逆时针向为正；反力以坐标轴之负向为正。

对于左端固定、右端铰支的情况，只要用 $M_L^\circ-\frac{1}{2}M_R^\circ$代替$M_L^\circ$；0代替$M_R^\circ$；$F_{YL}^\circ+\frac{3}{2l}M_R^\circ$代替 F_{YL}°；$F_{YL}^\circ-\frac{3}{2l}M_R^\circ$ 代替 F_{YR}° 就可以了。

内力正向如图 2.3 所示。即轴力以拉为正；剪力以使杆顺时针转动为正；弯矩以使杆中垂为正。

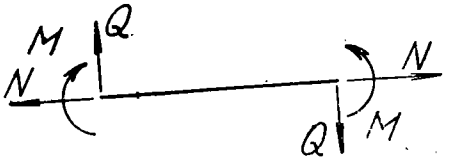


图 2.3

三、线代数方程组的求解

由于总刚度矩阵具有高度烯疏性（带状性）和对称正定性的特点，所以总刚度矩阵采用标记对角元位置的方法，以每行第一个非零元开始到该行对角元为止的所有元素按行一维压缩存放。每行从第一个非零元开始到该行对角元为止的元素数目称为该行的行带宽。

本程序采用修改的 *Cholesky* 法求解。先对总刚矩阵作对称分解：

$$\mathbf{A}=\mathbf{L}\mathbf{D}\mathbf{L}^T,$$

其中 **L** 是实的下三角矩阵，主对角元为1，**D** 是正对角矩阵，$\mathbf{L}^T$ 是 **L** 的转置矩阵。

设 **A** 为 n 阶方阵，其第 i 行第一个非零元为 $a_{i,}$ m_i，令

$$j_b=max\{m_i, m_j\},$$

则计算 **L**，**D** 元素的公式为

$$t_{ij}=a_{ij}-\sum_{k=j_b}^{j-1}t_{ik}\,l_{jk}\quad(i\geqslant j,\ j=m_i, m_i+1, \cdots, i-1)$$

$$l_{ij}=t_{ij}/d_{jj},$$

$$d_{ii}=a_{ii}-\sum_{k=m_i}^{i-1}t_{ik}\,l_{ik}\quad(i=1, 2, \cdots, n),$$

引进 t_{ij} 可以避免重复的计算，对于变带宽型的矩阵，上述分解不改变行带宽。

分解后按下列次序求解；。

$$LZ = B,$$
$$DY = Z,$$
$$L^TX = Y.$$

以 m 记最大行带宽，并对第 i 行记

$$i_b = min\{n, m+i-1\},$$

则有计算公式：

$$z_i = b_i - \sum_{k=m_i}^{i-1} l_{ik} z_k \quad (i = 1, 2, \cdots, n),$$

$$x_i = z_i/d_{ii} - \sum_{k=i+1}^{i_b} l_{ki} x_k \quad (i = n, \cdots, 2, 1).$$

为了进一步节约存贮单元，程序先进行特殊节点处理，即把柱固定端的 θ, u, v 及横梁右端的水平位移 u 从未知量中除去后再对未知量重新自动编号。这样，形成总刚度矩阵后，就可以直接对代数方程组求解了。

四、内力组合及配筋计算

每组荷载作用下每根杆都计算三个截面的内力 N, M, Q, 并进行不利荷载组合，内容有六种：

(1) $Mmax$ 及相应的 N, Q;
(2) $Mmin$ 及相应的 N, Q;
(3) $|N|min$ 及相应的 $Mmax$, Q;
(4) $|N|min$ 及相应的 $Mmin$, Q;
(5) $|N|max$ 及相应的 $Mmax$, Q;
(6) $|N|max$ 及相应的 $Mmin$, Q。

从目前单层工业厂房设计计算情况来看，排架上作用的荷载主要是：恒载、屋面活载、风载、吊车荷载。其中组合主要有三种：

(1) 恒+(活)+吊;
(2) 恒+(活)+风;
(3) 恒+(活)+Qv×(风+吊);

其中

$$Qv = 0.8或0.9,$$

视具体情况由数据输入。

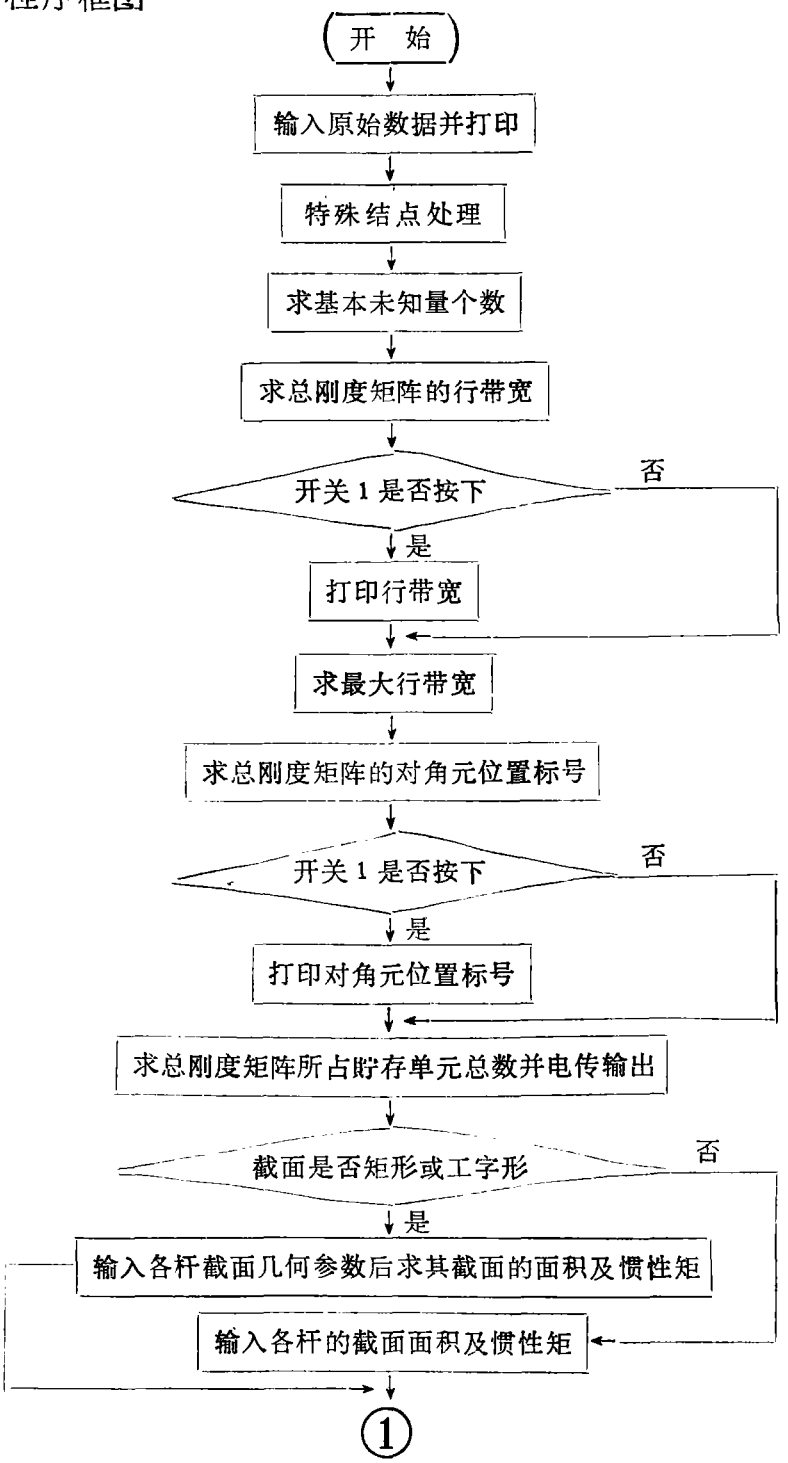
(活)表示屋面活载，它只当组合情况是(5)，(6)时才加进去。

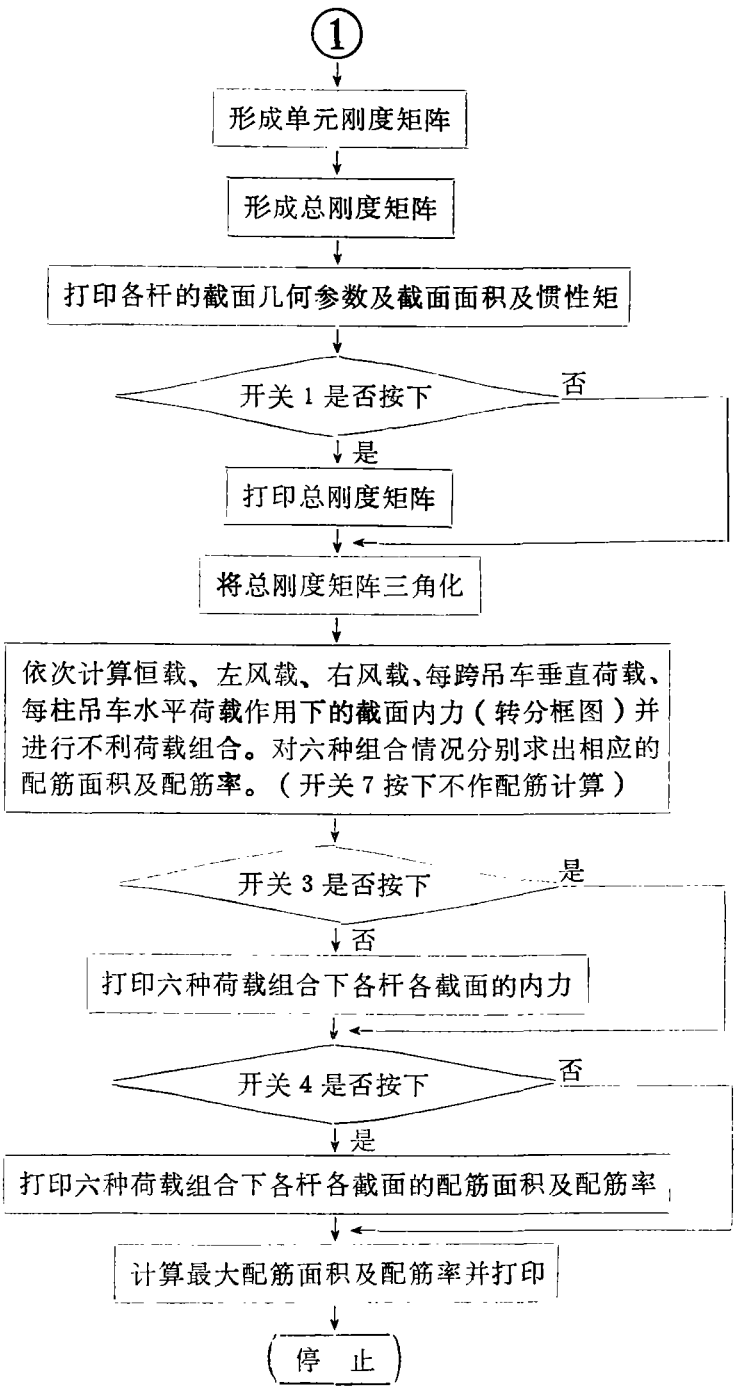
对上述三种情况进行不利荷载组合时，都是由 $|M|max$ 来控制的。

配筋计算主要参照[2]。

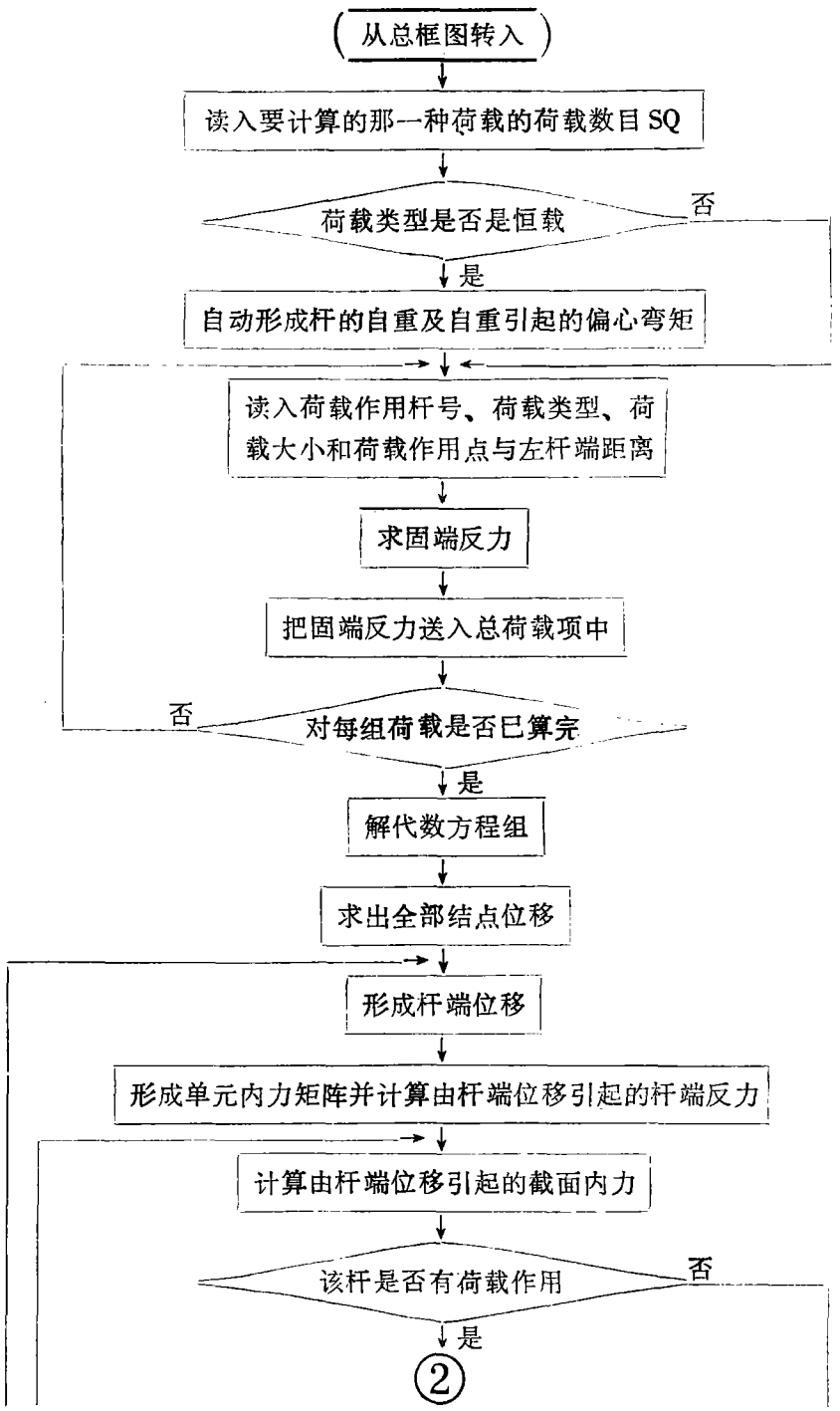
程序中所用到的一些 DJS—21 机扩充标准过程请参阅[3]。

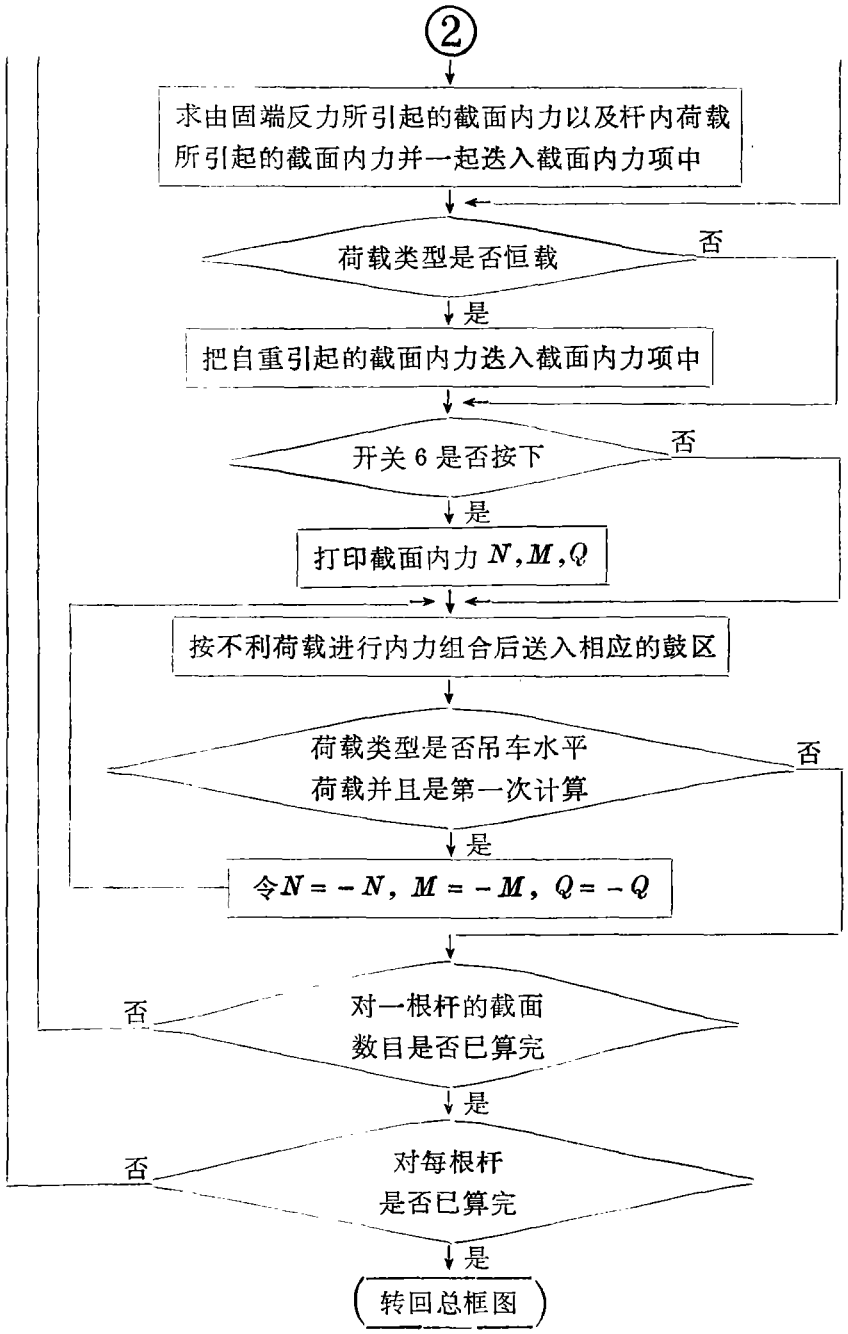
五、程序框图





总 框 图





计算截面内力的分框图

六、应 用 实 例

使用本程序，曾对某些生产实际问题进行了计算。现按图6.1所示的双跨单层排架作为例子，其荷载情况见〔1〕，表6—1整理出它的部分计算结果。重量单位用吨，长度单位用米。

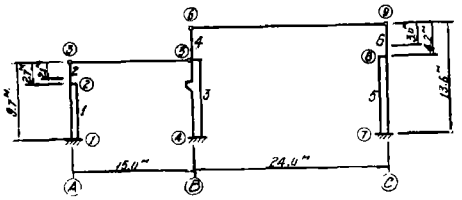


图6,1 計算草图与杆件节点編号

七、数 据 填 写 次 序

现以上节图6.1为例，说明数据填写次序如下：

表1 基 本 数 据

节点数N	杆 数 M	固端数NC	跨 数NA	一端固定 一端铰支 杆数GG	横梁数NV
9	6	3	2	3	2

表2 杆 端 节 点 编 号

杆 号 (1~M)	1	2	3	4	5	6
左节点編号 HL	1	2	4	5	7	8
右节点編号 HR	2	3	5	6	8	9

表3 横梁左右节点编号

跨数号 (1~NA)	1	2
左节点編号 KHL	3	6
右节点編号 KHR	5	9

表4 固定端节点编号

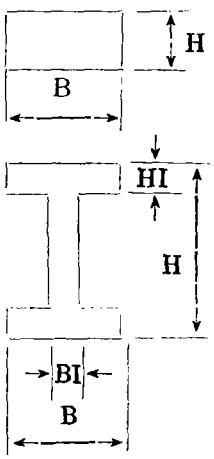
固定端数号(1~NC)	1	2	3
固定端节点号 KNC	1	4	7

表5 一端固定一端铰支杆杆号

杆数（1～GG）	1	2	3
杆号 KGG	2	4	6

表6 杆的物理与几何性能参数

杆号 (1～M)	截面形狀參數 SW	彈性模数 E	杆長 L	截面寬 B	截面高 H	腹板寬 BI	翼板厚 HI
1	2	1	7	0.4	0.6	0.12	0.1
2	1	1	2.7	0.4	0.4		
3	2	1	9.4	0.4	0.8	0.12	0.2
4	1	1	4.2	0.4	0.4		
5	2	1	9.4	0.4	0.8	0.12	0.2
6	1	1	4.2	0.4	0.4		



截面为矩形SW = 1，工字型SW = 2，其它SW = 0。

当SW = 0：则在B,H栏内填F,J值。

BI,HI栏不填。

SW = 1：则只填B,H。

BI,HI栏不填。

SW = 2：各栏均填，

表7 配筋计算参数

保护层厚度 (a)	混凝土抗弯計算强度 (Rw)	鋼筋抗拉計算强度 (Rg)	荷載組合系数 (Qv)
0.04	1000	21000	0.8

表8。 杆截面偏心距（使自重产生顺时针偏心弯矩之偏心距为正）。

杆 号 (1~M)	1	2	3	4	5	6
偏 心 距 e	0	−0.1	0	0.1	0	0.2

表9。 荷 载 情 况

荷载类型	每类荷载数目 SQ	作用杆号 QI[1]	类型 QI[2]	大小 QR[1]	作用点离左端距离 QR[2]
恒 载	14				
		1	4	−3.48	7
		2	4	−7.56	2.7
		3	4	−3.48	7
		3	4	−26.15	9.4
		4	4	−13.08	4.2
		5	4	−7.48	9.4
		6	4	−13.08	4.2
		1	6	0.81	7
		2	6	−0.38	2.7
		3	6	−2.96	7
		3	6	1.50	9.4
		4	6	−0.65	4.2
		5	6	0	9.4
		6	6	0.65	4.2
左来风	7				
		1	1	−0.25	7
		2	1	−0.25	2.7
		2	2	0.078	2.7
		4	1	−0.125	4.2
		5	1	−0.156	9.4
		6	1	−0.156	4.2
		6	2	0.125	4.2
右来风	7				
		1	1	0.156	7
		2	1	0.156	2.7
		2	2	−0.078	2.7
		4	1	0.125	4.2
		5	1	0.25	9.4
		6	1	0.25	4.2
		6	2	−0.25	4.2

吊車垂直荷載	4				
		1	6	9.27	7
		1	4	−20.6	7
		3	6	−8.75	7
		3	4	−10.3	7
	4				
		1	6	4.62	7
		1	4	−10.3	7
		3	6	−17.5	7
		3	4	−20.6	7
	4				
		3	6	42.8	9.4
		3	4	−65.8	9.4
		5	6	−7.42	9.4
		5	4	−21.2	9.4
	4				
		3	6	13.8	9.4
		3	4	−21.2	9.4
		5	6	−23.1	9.4
		5	4	−65.8	9.4
吊車水平荷載	1				
		2	2	0.58	0.6
	1				
		3	2	0.58	7.6
	2				
		3	2	0.29	7.6
		4	2	1.225	10.6
	1				
		4	2	2.45	1.2
	1				
		6	2	2.45	1.2
屋面活載	10				
		1	6	−0.18	7
		2	6	−0.09	7
		3	6	−0.16	9.4
		4	6	−0.14	4.23
		5	6	0.56	9.4
		6	6	0.14	4.2
		2	4	−1.76	2.7
		4	4	−1.76	0.3
		4	4	−2.81	4.2
		6	4	−2.81	4.2

八、源 程 序

Y

```
BEGIN
 INTEG N, M, NC, NA, GG, I, J, K, R, S, SW, NV;
 REAL T, FU, RK, LK, BK, HK, BI, HI, H;
 DRUM ARRAY DMS[1:13000];
 READI 1 (6,N,M,NC,NA,GG,NV);
 PUSH(0,-1); PUSH (1,1);
 PRINTI(6, N, M, NC, NA, GG, NV);
 BEGIN
  INTEG NEQ, NN, MBW;
  INTEG ARRAY HL, HR[1:M], KHL, KHR[1:NV],
  KNC[1:NC], KGG[1:GG], DE[1:6], DN[1:N, 1:3];
  READI(6, HL, HR, KHL, KHR, KNC, KGG);
  PUSH(0,-1); PUSH(2,1);
  APRINT1(6, HL, HR, KHL, KHR, KNC, KGG);
  FOR I:=1 STEP 1 UNTIL N DO
  FOR K:=1,2,3 DO
   DN[I, K]:=1;
  FOR I:=1 STEP 1 UNTIL NC DO
  FOR K:=1,2,3 DO
   DN[KNC[I], K]:=0;
  FOR I:=1 STEP 1 UNTIL NV DO
   DN[KHR[I], 3]:=0;
   MBW:=NEQ:=0;
  FOR I:=1 STEP 1 UNTIL N DO
  FOR K:=1,2,3 DO
  IF DN[I, K]=1 THEN
   NEQ:=DN[I, K]:=NEQ+1;
  FOR I:=1 STEP 1 UNTIL NV DO
   DN[KHR[I], 3]:=DN[KHL[I], 3];
  BEGIN
   INTEG ARRAY BW[0:NEQ];
   PROC CDE(I);
```

```
    VALUE I; INTEG I;
    BEGIN
     INTEG K;
     FOR K:=1,2,3 DO
     BEGIN
      DE[K]:=DN[HL[I],K];
      DE[3+K]:=DN[HR[I],K]
     END
    END;
    FOR I:=0 STEP 1 UNTIL NEQ DO
     BW[I]:=0;
    FOR I:=1 STEP 1 UNTIL M DO
    BEGIN
     CDE(I);
     FOR K:=1 STEP 1 UNTIL 6 DO
     BEGIN
      IF DE[K]=0 THEN GOTO L2;
      FOR S:=1 STEP 1 UNTIL 6 DO
      BEGIN
       IF DE[K] LS DE[S] OR DE[S]=0 THEN GOTO L1;
       R:=DE[K]-DE[S]+1;
       IF R GR BW[DE[K]] THEN
        BW[DE[K]]:=R;
L1:    END;
L2:   END
    END;
    JUMP(1,LP);  GOTO LQ;
LP: PUSH(0,-1); PUSH(3,1); APRINT(BW);
LQ: FOR I:=1 STEP 1 UNTIL NEQ DO
    IF BW[I] GR MBW THEN
     MBW:=BW[I];
    FOR I:=2 STEP 1 UNTIL NEQ DO
     BW[I]:=BW[I]+BW[I-1];
     NN:=BW[NEQ];
    TPRINT(0=0, NN, "NN=");
    JUMP(1,LS); GOTO LT;
```

```
LS:PUSH(0,-1); PUSH(4,1); APRINT(BW);
LT:BEGIN
    ARRAY FJ[1:M, 0:8], KE[1:6, 1:6],
          KM[1:6, 1:4], KA[1:NN];
    PROC  CFJ1(B,H,F,R);
    VALUE B,H;  REAL B,H,F,R;
    BEGIN
     F:=B×H;
     R:=F×H×H/12
    END;
    PROC  CFJ2(B,H,BI,HI,F,R);
    VALUE B,H,BI,HI;
    REAL  B,H,BI,HI,F,R;
    BEGIN
     F:=B×H-(B-BI)×(H-2×HI);
     R:=(B×H××3-(B-BI)×H-2×HI)××3)/12
    END;
    PROC  CKE(E,F,R,L);
    VALUE E,F,R,L;  REAL E,F,R,L;
    BEGIN
     INTEG I,J;  REAL T;
     KE[1,1]:=KE[4,4]:=4;
     KE[2,2]:=KE[5,5]:=F/R;
     KE[3,3]:=KE[6,6]:=12/L/L;
     KE[4,1]:=2;
     KE[5,2]:=-KE[2,2];
     KE[6,1]:=KE[6,4]:=6/L;
     KE[3,1]:=KE[4,3]:=-KE[6,1];
     KE[6,3]:=-KE[3,3];
     T:=E×R/L;
     FOR I:=1 STEP 1 UNTIL 6 DO
     FOR J:=1 STEP 1 UNTIL I DO
     KE[J,I]:=KE[I,J]:=T×KE[I,J]
    END;
    PROC CKM(E,F,R,L);
    VALUE E,F,R,L;  REAL E,F,R,L;
```

```
    BEGIN
     INTEG I,J;  REAL T;
     KM[5,1]:=F/R;
     KM[2,1]:=-KM[5,1];
     KM[1,2]:=KM[4,3]:=4;
     KM[4,2]:=2;
     KM[6,2]:=KM[6,3]:=6/L;
     KM[3,2]:=KM[3,3]:=KM[1,4]:=KM[4,4]:=-KM[6,2];
     KM[3,4]:=12/L/L;
     KM[6,4]:=-KM[3,4];
     T:=E×R/L;
     FOR I:=1 STEP 1 UNTIL 6 DO
     FOR J:=1 STEP 1 UNTIL 4 DO
     KM[I,J]:=T×KM[I,J]
    END;
    PROC FK(E);
    VALUE E;  INTEG E;
    BEGIN
     INTEG U,V,K;  CDE(E);
     FOR K:=1 STEP 1 UNTIL 6 DO
     BEGIN
      IF DE[K]=0 THEN GOTO X1
      FOR V:=1 STEP 1 UNTIL 6 DO
      BEGIN
       IF DE[K]LS DE[V] OR DE[V]=0 THEN GOTO X2;
       U:=BW[DE[K]]-DE[K]+DE[V];
       KA[U]:=KA[U]+KE[K,V];
X2:   END;
X1:  END
    END;
    PROC LDLTVB(N,BW,A);
    VALUE N;  INTEG N;
    INTEG ARRAY BW;  ARRAY A;
    BEGIN
     INTEG I,J,K,IB,JB,IW,JW;
     REAL  SS,T;
```

```
  FOR I:=1STEP 1 UNTIL N DO
  BEGIN
   IW:=BW[I]-I;
   IB:=BW[I-1]-IW+1;
   FOR J:=IB+1 STEP 1 UNTIL I-1 DO
   BEGIN
    JW:=BW[J]-J;
    JB:=BW[J-1]-JW+1;
    IF JB LS IB THEN JB:=IB;
    SS:=A[IW+J];
    FOR K:=JB STEP 1 UNTIL J-1 DO
    SS:=SS-A[IW+K]×A[JW+K];
    A[IW+J]:=SS
   END;
   SS:=A[BW[I]];
   FOR J:=IB STEP 1 UNTIL I-1 DO
   BEGIN
    T:=A[IW+J];
    A[IW+J]:=T×A[BW[J]];
    SS:=SS-T×A[IW+J]
   END;
   A[BW[I]]:=1/SS
  END
 END;
STOA(0,FJ); STOA(0,KE); STOA(0,KA);
FOR I:=1 STEP 1 UNTIL M DO
BEGIN
 SW:=READI; FJ[I,0]:=SW;
 READR1 (2, FJ[I,1], FJ[I,2]);
 IF SW=1 THEN
 BEGIN
  READR1 (2, FJ[I,3], FJ[I,4]);
  CFJ1 (FJ[I,3], FJ[I,4], FJ[I,7], FJ[I,8]);
  GOTO L3
 END;
 IF SW=2 THEN
```

```
      BEGIN
       READR1 (4, FJ[I,3], FJ[I,4], FJ[I,5], FJ[I,6]);
       CFJ2 (FJ[I,3], FJ[I,4], FJ[I,5], FJ[I,6], FJ[I,7],FJ[I,8]);
       GOTO L3
      END;
      READR1 (2, FJ[I,7], FJ[I,8]);
L3:   CKE(FJ[I,1], FJ[I,7], FJ[I, 8], FJ[I,2]);
      FK(I)
     END;
     PUSH(0,-1); PUSH(5,3); APRINT(FJ);
     JUMP(1, L4); GOTO L5;
L4:  PUSH(0,-1); PUSH(6,3); APRINT(KA);
L5:  LDLTVB (NEQ, BW, KA);
     BEGIN
      INTEG PA, PB, PC, I1, J1;
      REAL RBX, RBY, MB, T1, N1, BM, U, G, PX;
      ARRAY QD[0:NEQ], W[1:N,1:3], MEO[1:M],
            V [1:6], MOM[1:4], RKV[1:4],
            R2[1:6], QR[1:150,1:2],
            QCON[1:M, 1:3, 1:3],
            MSD[1:6, 1:M, 1:3, 1:3];
      INTEG ARRAY QI[1:150, 1:2];
      PROC SOLVB(N, M, BW, A, B);
      VALUE N,M;  INTEG N,M;
      INTEG ARRAY BW;  ARRAY A,B;
      BEGIN
       INTEG I, J, IB, IW;
       REAL SS;
       FOR I:=2 STEP 1 UNTIL N DO
       BEGIN
        SS:=B[I];
        IW:=BW[I]-I;
        IB:=BW[I-1]-IW+1;
        FOR J:=I-1 STEP -1 UNTIL IB DO
        SS:=SS-A[IW+J]×B[J];
        B[I]:=SS
```

```
      END;
      FOR I:=N STEP -1 UNTIL 1 DO
      BEGIN
       SS:=B[I]×A[BW[I]];
       IB:=IF M+I-1 LS N THEN M+I-1 ELSE N;
       FOR J:=I+1 STEP 1 UNTIL IB DO
       IF I GQ J-BW[J]+BW[J-1]+1 THEN
       SS:=SS-A[BW[J]-J+I]×B[J];
       B[I]:=SS
      END
     END;
     PROC FORCE(KK);
     VALUE KK; INTEG KK;
     BEGIN
      INTEG I, K;
      REAL RAX, RAY, MA, E, T;
      SWITCH SS:=S1, S2, S3, S4, S5, S6;
      RAX:=RAY:=RBX:=RBY:=MA:=MB:=0;
      STOA(0,R2); K:=QI[KK,1];
      G:=QR[KK,1]; U:=QR[KK,2];
      LK:=FJ[K,2]; E:=U/LK;
      T:=1-E; GOTO SS[QI[KK,2]];
S1:   RAY:=G×U×(2-2×E×E+E××3)/2;
      RBY:=-RAY+G×U;
      MA:=-G×U×U×(6-8×E+3×E×E)/12;
      MB:=G×U×U×E×(4-3×E)/12;
      GOTO S9;
S2:   RAY:=G×T×T×(1+2×E);
      RBY:=-RAY+G;
      MA:=-G×U×T×T;
      MB:=G×E×E×T×LK;
      GOTO S9;
S3:   RAX:=G×U×(1-E/2);
      RBX:=-RAX+G×U;
      GOTO S10;
S4:   RAX:=G×T;
```

```
      RBX:=-RAX+G;
      GOTO S10;
S5:   RAY:=G×U×(2-3×E×E+1.6×E×E×E)/4;
      RBY:=-RAY+G×U/2;
      MA:=-G×U×U×(2-3×E+1.2×E×E)/6;
      MB:=G×U×U×E×(1-0.8×E)/4;
      GOTO S9;
S6:   RAY:=6×G×E×T/LK;
      RBY:=-RAY;
      MA:=-G×T×(2-3×T);
      MB:=-G×E×(2-3×E);
S9:   FOR I:=1 STEP 1 UNTIL GG DO
      IF KGG[I]=K THEN
      BEGIN
       RAY:=RAY+3×MB/2/LK;
       RBY:=RBY-3×MB/2/LK;
       MA:=MA-MB/2;  MB:=0
      END;
S10:  R2[1]:=MA;      R2[4]:=MB;
      R2[2]:=RAX;     R2[3]:=RAY;
      R2[5]:=RBX;     R2[6]:=RBY;
     END;
     PROC IFORCE(KK);
     VALUE KK;      INTEG KK;
     BEGIN
      REAL T2, N2, BM2;
      SWITCH SS:=S1, S2, S3, S4, S5, S6;
      FORCE(KK);  T2:=N2:=BM2:=0;
      T1:=RBY;  N1:=-RBX;
      BM:=MB-RBY×(LK-PX);
      IF U GQ PX THEN GOTO SS[QI.KK,2]] ELSE GOTO S10;
S1:   T2:=-G×(U-PX);
      BM2:=G×(U-PX)××2/2;
      GOTO S9;
S2:   T2:=-G;
      BM2:=G×(U-PX);
```

```
      GOTO S9;
S3:   N2:=G×(U-PX);
      GOTO S9;
S4:   N2:=G; GOTO S9;
S5:   T2:=-G×(1+PX/U)×(U-PX)/2;
      BM2:=G×(U-PX)×2×(1/3+PX/U/6);
      GOTO S9;
S6:   BM2:=-G;
S9:   BM:=BM+BM2
      T1:=TI+T2;
      N1:=N1+N2;
S10: END;
     PROC CQD(K);
     VALUE K; INTEG K;
     BEGIN
      INTEG J;
      BEGIN
       CDE(K);
       FOR J:=1 STEP 1 UNTIL 6 DO
       QD[DE[J]]:=QD[DE[J]]+R2[J]
      END
     END;
     PROC CW;
     BEGIN
      INTEG I,J,K;
      FOR I:=1 STEP 1 UNTIL N DO
      FOR J:=1,2,3 DO
      BEGIN
       K:=DN[I,J];
       W[I,J]:=W[I,J]+QD[K]
      END
     END;
     PROC QCOM(SQ,P);
     VALUE SQ, P; INTEG SQ,P;
     BEGIN
      INTEG S,I,J,K,I1,J1;  REAL T;
```

```
STOA(0,QD);
IF P=1 THEN
FOR I:=1 STEP 1 UNTIL M DO
BEGIN
 K:=QI[I,1]:=I; QI[I,2]:=4;
 QR[I, 1]:=-2.75×FJ[I,7]×FJ[I,2];
 QR[I,2]:=0;
 FORCE(I); CQD(K);
 T:=-MEO[I]; QI[I,2]:=6;
 QR[I,1]:=QR[I,1]×T; QR[I,2]:=0;
 FORCE(I); CQD(K)
END;
FOR I:=1 STEP I UNTIL SQ DO
BEGIN
 READI1(2,QI[I,1], QI[I,2]);
 PRINTI(2,QI[I,1], QI[I, 2]);
 READR1(2,QR[I,1],QR[I,2]);
 PRINTR(2,QR[I,1], QR[I,2]);
 FORCE(I); K:=QI[I,1]; CQD(K)
END;
QD[0]:=0;
SOLVB(NEQ, MBW, BW, KA, QD);
STOA(0, W); CW; STOA(0, KM);
FOR I:=1 STEP 1 UNTIL M DO
BEGIN
 FOR K:=1, 2, 3 DO
 BEGIN
  V[K]:=W[HL[I], K];
  V[3+K]:=W[HR[I],K]
 END;
 CKM(FJ[2, 1], FJ[I,7], FJ[I,8], FJ,[I,2]);
 FOR J1:=1 STEP 1 UNTIL 4 DO
 BEGIN
  T:=0;
  FOR I1:=1 STEP 1 UNTIL 6 DO
  T:=T+V[I1]×KM[I1, J1];
```

```
 MOM[J1]:=T
END;
FOR J:=1, 2, 3 DO
BEGIN
 ARRAY IF[1:3];
 PROC MAD(SK);
 VALUE SK; INTEG SK;
 BEGIN
  FOR K:=1,2,3 DO
  MSD[SK,I,J, K]:=IF[K]
 END;
 PROC GLQ(S);
 VALUE S; INTEG S;
 BEGIN
  IF IF[2] GR 0 AND IF[2] GR MSD[S,I,J,2] THEN
  MAD(S);
  IF IF[2] LS 0 AND IF[2] LS MSD[S+1,I,J,2]THEN
  MAD(S+1)
 END;
 PX:=IF J=1 THEN 0 ELSE
 IF J=2 THEN FJ[I,2]/2 ELSE FJ[I,2];
 IF[1]:=MOM[1];
 IF[2]:=MOM[2]+MOM[4]×PX;
 IF[3]:=MOM[4];
 IF P=1 THEN
 BEGIN
  T:=-2.75×FJ[I,7]×FJ[I,2];
  IF J=3 THEN H:=0;
  IF J=2 THEN H:=T/2;
  IF J=1 THEN H:=T;
  IF[1]:=IF[1]+H
 END;
 FOR S:=1 STEP 1 UNTIL SQ DO
 BEGIN
   IF I=QI[S,1] THEN
   BEGIN
```

```
        IFORCE(S);
        IF[1]:=IF[1]+N1;
        IF[2]:=IF[2]+BM;
        IF[3]:=IF[3]+T1
       END
      END;
      JUMP(6,LIF); GOTO LGT;
LIF:  PUSH(0,-1); PUSH(10×I+J,1);
      APRINT(IF);
LGT:  IF P=1 THEN GOTO DX;
      IF P=4 AND IF[1] LS 0 THEN GOTO DX;
      IF P=2 OR P=3 THEN GOTO DZ;
DX:   FOR K:=1,2,3 DO
      QCON[I,J,K]:=QCON[I,J,K]+IF[K];
      GOTO DY;
DZ:   J1:=0;
DY:   J1:=J1+1;
      IF IF[2] GR MSD[1,I,J,2] THEN MAD(1);
      IF IF[2] LS MSD[2,I,J,2] THEN MAD(2);
      IF IF[1] GR MSD[3,I,J,1] THEN
      BEGIN
       MAD(3); MAD(4)
      END;
      IF IF[1] LS MSD[5,I,J,1] THEN
      BEGIN
       MAD(5); MAD(6)
      END;
      IF IF[1]=0 THEN
      BEGIN
       GLQ(3);     GLQ(5)
      END;
      IF P=3 AND J1=1 THEN
      BEGIN
       FOR K:=1,2,3 DO IF[K]:=-IF[K];
       GOTO DY
      END;
```

```
DV:   END
      END
     END;
     PROC ADDP;
     BEGIN
      INTEG S,I,J,K;
      FOR S:=1 STEP 1 UNTIL 6 DO
      BEGIN
       GUDXIN(DMS,QCON,(6+S)×9×M+1);
       FOR I:=1 STEP 1 UNTIL M DO
       FOR J:=1,2,3 DO
       FOR K:=1,2,3 DO
       MSD[S,I,J,K]:=MSD[S,I,J,K]+QCON[I,J,K]
      END;
      XINDGU(MSD,DMS,63×M+1)
     END;
     PROC DGU(K1,K2);
     VALUE K1,K2;   INTEG K1,K2;
     BEGIN
      GUDXIN(DMS,QCON,K1);
      ATOB(QCON,MSD,K2)
     END;
     PROC MAX(P);
     VALUE P;   INTEG P;
     BEGIN
      REAL A2,A3,A4;
      FOR I:=1 STEP 1 UNTIL M DO
      FOR J:=1,2,3 DO
      FOR K:=1,2,3 DO
      BEGIN
       A2:=ABS(MSD[2,I,J,P]);
       A3:=ABS(MSD[3,I,J,P]);
       A4:=ABS(MSD[4,I,J,P]);
       QCON[I,J,K]:=IF A2 GR
       (IF A3 GR A4 THEN A3 ELSE A4)
       THEN MSD[2,I,J,K] ELSE
```

```
  (IF A3 GR A4 THEN MSD[3,I,J,K] ELSE MSD[4,I,J,K])
 END
END;
PROC MIN(P);
VALUE P;  INTEG P;
BEGIN
 REAL A2,A3,A4;
 IF S NQ 5 THEN
 FOR I:=1 STEP 1 UNTIL M DO
 FOR J:=1,2,3 DO
 FOR K:=1,2,3 DO
 QCON[I,J,K]:=IF S=2 THEN 0 ELSE MSD[6,I,J,K];
 FOR I:=1 STEP 1 UNTIL M DO
 FOR J:=1,2,3 DO
 FOR K:=1,2,3 DO
 BEGIN
  A2:=ABS(MSD[2,I,J,P]+QCON[I,J,P]);
  A3:=ABS(MSD[3,I,J,P]+QCON[I,J,P]);
  A4:=ABS(MSD[4,I,J,P]+QCON[I,J,P]);
  QCON[I,J,K]:=IF A2 GR
  (IF A3 GR A4 THEN A3 ELSE A4)
  THEN MSD[2,I,J,K]+QCON[I,J,K]
  ELSE (IF A3 GR A4 THEN MSD[3,I,J,K]+QCON[I,J,K]
  ELSE MSD[4,I,J,K]+QCON[I,J,K])
 END
END;
READ1(2,RKV,MEO);
PUSH(0,-1); APRINT(MEO); PUSH(0,-1);
STOA(0,QCON);  STOA(0,MSD);
XINDGU(QCON,DMS,1);
XINDGU(MSD,DMS,9×M+1);
XINDGU(MSD,DMS,63×M+1);
PA:=READI;  PUSH(7,3);
QCOM(ABS(PA),1);
IF PA LS 0 THEN GOTO L14;
XINDGU(QCON,DMS,1);
```

```
    PB:=READI;   PUSH(8,3);
    QCOM(PB,2);
    PC:=READI;   PUSH(9,3);
    QCOM(PC,2);
    XINDGU(MSD,DMS,9×M+1);   STOA(0,MSD);
    FOR I1:=1 STEP 1 UNTIL NA DO
    BEGIN
     PB:=READI;   PUSH(2×I1+8,3);
     QCOM(PB,2);
     PC:=READI;   PUSH(2×I1+9,3);
     QCOM(PC,2);
     ADDP;   STOA(0,MSD)
    END;
    FOR J1:=1 STEP 1 UNTIL 3×NA-1 DO
    BEGIN
     PB:=READI;   PUSH(2×NA+9+I1,3);
     QCOM(PB,3)
    END;
    ADDP;   STOA(0,QCON);
L6: BEGIN
     REAL DTA,RW,RG,AGMAX,QV,H0;
     ARRAY AG[1:6,1:M,1:3];
     DTA:=RKV[1];RW:=RKV[2];RG:=RKV[3];QV:=RKV[4];
     PUSH(5×NA+9,1);
     PRINTR(4,DTA,RW,RG,QV);
     PUSH(5×NA+10,3);
     STOA(0,AG);
     FOR S:=1,2,3,4,5,6 DO
     BEGIN
      SWITCH PP:=P1,P2,P3,P4,P5,P6;
      DGU(1,1);
      DGU(9×S×M+1,9×L+1);
      DGU(9×(S+6)×M+1,18×M+1);
      STOA(0,QCON);
P0:   FOR I:=1 STEP 1 UNTIL M DO
      FOR J:=1,2,3 DO
```

```
        FOR K:=1,2,3 DO
        BEGIN
         MSD[4,I,J,K]:=MSD[1,I,J,K]+QV×(MSD[2,I,J,K]+MSD[3,I,J,K]);
         MSD[2,I,J,K]:=MSD[1,I,J,K]+MSD[2,I,J,K];
         MSD[3,I,J,K]:=MSD[1,I,J,K]+MSD[3,I,J,K]
        END;
        GOTO PP[S];
P1:     MAX(2);   GOTO P7;
P2:     MAX(2);   GOTO P7;
P3:     MAX(2);   GOTO P7;
P4:     MAX(2);   GOTO P7;
P5:     PB:=READI;   QCOM(PB,4);
        ATOB(QCON,MSD,45×M+1);
        MIN(2);   GOTO P7;
P6:     MIN(2);
P7:     JUMP(3,L7);
        FOR I:=1 STEP 1 UNTIL M DO
        FOR J:=1,2,3 DO
        BEGIN
         PUSH(0,-1);  PUSH(10000×S+100×I+J,1);
         FOR K:=1,2,3 DO
         PRINT(QCON[I,J,K]);
L7:     END;
        JUMP(7,L13);
        FOR I:=1 STEP 1 UNTIL M DO
        BEGIN
         FU:=FJ[I,7];
         RK:=GN2(FJ[I,8]/FJ[I,7]);
         LK:=FJ[I,2];  BK:=FJ[I,3];
         HK:=FJ[I,4];  BI:=FJ[I,5];
         HI:=FJ[I,6];  H:=LK/RK;  H0:=HK-DTA;
         FOR J:=1,2,3 DO
         BEGIN
          REAL E0,E1,E2,E3,QN,X,RAMT;
          QN:=-QCON[I,J,1];
          IF QN=0 THEN GOTO L10;
```

```
        RAMT:=IF H LQ 3.5 THEN 1 ELSE
        1/(1-QN×H××2/4800/RW/FU);
        E0:=RAMT×ABS(QCON[I,J,2]/QN);
        E1:=E0+HK/2-DTA;
        E2:=ABS(E0-HK/2+DTA);
        E3:=IF E1 GR E2 THEN E1 ELSE E2;
        IF FJ[I,0]=0 THEN GOTO L10;
        IF FJ[I,0]=1 THEN GOTO L9;
        IF FJ[I,0]=2 AND QN/RW/BK LQ HI THEN GOTO L9;
        X:=(QN-0.8×RW×(BK-BI)×HI)/RW/BI;
        IF X LQ HK-HI THEN
        AG[S,I,J]:=(QN×E1-0.8×RW×(BK-BI)×HI×(H0-HI/2))/RG
        /(H0-DTA)-(RW×BI×X×(H0-X/2))/RG/(H0-DTA);
        IF X GR HK-HI OR AG[S,I,J] LS 0
        THEN
        AG[S,I,J]:=(QN×E3-0.8×RW×(BK-BI)×HI×(H0-HI/2))
                   RG/(H0-DTA)
        -0.4×RW×(BI×H0××2+(BK-BI)×(HI-DTA)××2)/RG
        /(H0-DTA);
        GOTO L10;
L9:     T:=QN/RW/BK/H0;
        IF T GR 0.55 THEN
        BEGIN
         AG[S,I,J]:=(QN×E1-0.4×BK×H0××2×RW)/RG/(H0-DTA);
         GOTO L10
        END;
        IF T LS 2×DTA/H0 THEN
        AG[S,I,J]:=QN×E2/RG/(H0-DTA)
        ELSE
        AG[S,I,J]:=QN×(E1-H0+QN/2/RW/BK)/RG/(H0-DTA);
L10:    JUMP(4,L11);   GOTO L12;
L11:    T:=AG[S,I,J];   PUSH(0,-1);
        PUSH(1000000×S+100×I+J,1);
        PRINT(T);
        T:=T/FU;   PRINT(T);
L12:  END
```

```
      END
     END;
     PUSH(5×NA+11,3);
     FOR I:=1 STEP 1 UNTIL M DO
     FOR J:=1,2,3 DO
     BEGIN
      AGMAX:=0;
      FOR S:=1,2,3,4,5,6 DO
      IF AG[S,I,J] GR AGMAX THEN
      AGMAX:=AG[S,I,J];
      PUSH(0,-1);
      PUSH(100×I+J,1);
      PRINT(AGMAX);
      T:=AGMAX/FJ[I,7];
      PRINT(T);
L13:  END
    END
   END
  END
 END
END;
L14:  END
```

五个以上字母键

J

参 考 资 料

〔1〕建筑結构設計手册排架計算，中国工业出版社。

〔2〕鋼筋混凝土結构設計規范(BJG21—66)。

〔3〕怎样使用121机算法語言，中山大学数力系計算数学教研室（1974,10)。