PLATE

Finite Element Computer Program for Analysis of Thin Plates

Version 1.0

prepared by

V. N. Kaliakin
Department of Civil & Environmental Engineering
University of Delaware
Newark, DE 19716

I. INTRODUCTION

The PLATE computer program has been written to perform quasi-static analyses of thin plates. The program is based on a similar code, originally written by Professor L. R. Herrmann at the University of California, Davis. PLATE represents a slightly enhanced version of this original code.

This document is divided into five parts. The present introduction represents part one. In the second part the information required to mathematically describe the problem being analyzed (i.e., the Program Input) is outlined. Only the briefest notes of explanation are included in this part of the document. Until the analyst becomes familiar with the program, he or she will need to refer to the third part of the document (titled Explanatory Notes Regarding Program Input) where detailed explanations and short examples are given. An overview of the theory underlying the analysis performed in PLATE is briefly described in Note 1 of this part. In the fourth part, the output from a typical analysis is explained. In the final part some sample analyses performed using the PLATE program are presented.

II. Program Input

The units used for various input quantities must be consistent, and determine the units of the output. The data required by the PLATE program consists of the following quantities.

2.1 Identification Record (A72)

TITLE [alphanumeric] : any title describing the analysis

2.2 Nodal Coordinate Data

A record with a 1 (integer one) in the first field and padded with a sufficient number of zeros (see Part III). For example :

```
1  0  0  0  0  0  0  0  0
```

Next supply as many records as necessary to specify or generate locations of all exterior vertex nodes and of any interior vertex nodes whose positions must be precisely located. Each record must contain the following information.

NSEC [integer] = must be set equal to zero (0)

NUM [integer] = node number

XP [real] = x-coordinate of the node

YP [real] = y-coordinate of the node

INCR [integer] = numbering increment[1]

RATIO [real] = spacing ratio[1]

XC [real] = x-coordinate of a point interior to a circular arc[1]

YC [real] = y-coordinate of a point interior to a circular arc[1] ;
for a straight line, set **XC** = **YC** = 0.0

See also :
Explanatory notes 1, 2, 3 and 4 in Part III.

[1] this quantity is associated with the nodal coordinate generation option; further details regarding this option are given in Part III.

2.3 Element Information

A record with a 2 (integer two) in the first field and padded with a sufficient number of zeros (see Part III). For example :

```
2  0  0  0  0  0  0  0  0
```

Next supply as many records as necessary to define all the elements in the mesh. The order of the element records need bear no relationship to the locations of the elements in the mesh. Each element description consists of the following two records.

–> *Record 1 :*

NSEC [integer] = must be set equal to zero (0)

NODP(1) [integer] = first node number used in describing the element
NODP(2) [integer] = second node number used in describing the element
NODP(3) [integer] = third node number used in describing the element
NODP(4) [integer] = fourth node number used in describing the element
NODP(5) [integer] = fifth node number used in describing the element
NODP(6) [integer] = sixth node number used in describing the element
NODP(7) [integer] = seventh node number used in describing the element
NODP(8) [integer] = eighth node number used in describing the element; enter the above node numbers *in sequence*, reading *counterclockwise* around the element; the first entry must be a vertex node.

NMIS [integer] = number of additional elements in the layer

NMISP [integer] = number of additional layers

INCRP [integer] = numbering increment for the layers

2.3 Element Information (continued)

–> *Record 2 :*

NSEC [integer] = must be set equal to zero (0)

E [real] = Elastic (Young's) modulus

ENU [real] = Poisson's ratio

H [real] = plate thickness

Q [real] = normal load (pressure)

EMT [real] = thermal moment

See also :

Explanatory notes 1, 5 and 6 in Part III.

2.4 Vertex Node Specifications

A record with a 3 (integer three) in the first field and padded with a sufficient number of zeros (see Part III). For example :

```
3 0 0 0 0 0 0 0 0 0 0 0 0
```

Next supply as many records as necessary to specify zero and non-zero values for the transverse displacement (w) and non-zero values for the effective transverse shear (V_n) for all vertex boundary nodes. Any vertex boundary node not included in this section is assigned the boundary condition of $V_n = 0.0$. Each record must contain the following information.

NSEC [integer] = must be set equal to zero (0)

NUM [integer] = vertex node number

IFLAG [integer] = $\begin{Bmatrix} 0 \\ 1 \end{Bmatrix}$ when $\begin{Bmatrix} V_n \\ w \end{Bmatrix}$ is specified

VALUE [real] = value of $\begin{Bmatrix} V_n \\ w \end{Bmatrix}$ specified

NUMEND [integer] = number of the last vertex node in the sequence

INCR [integer] = numbering increment for the sequence of node numbers

See also :

Explanatory notes 1, 7 and 8 in Part III.

2.5 "Mid-Point" Node Specifications

A record with a 4 (integer four) in the first field and padded with a sufficient number of zeros (see Part III). For example :

```
4  0  0  0  0  0  0
```

Next supply as many records as necessary to specify zero and non-zero values for the bending moment intensity (M_n) and non-zero values for the slope (θ_n) for all mid-side boundary nodes. Any mid-side boundary node not included in this section is assigned the boundary condition of $\theta_n = 0.0$. Each record must contain the following information.

NSEC [integer] = must be set equal to zero (0)

NUM [integer] = mid-point node number

IFLAG [integer] = $\left\{\begin{matrix} 0 \\ 1 \end{matrix}\right\}$ when $\left\{\begin{matrix} \theta_n \\ M_n \end{matrix}\right\}$ is specified

VALUE [real] = value of $\left\{\begin{matrix} \theta_n \\ M_n \end{matrix}\right\}$ specified

NUMEND [integer] = number of the last mid-point node in the sequence

INCR [integer] = numbering increment for the sequence of node numbers

See also :

Explanatory notes 1, 7 and 8 in Part III.

2.6 End Record

A record with a 5 (integer five) in the first field and padded with a sufficient number of zeros (see Part III). For example :

```
5  0  0  0  0  0  0
```

The above record signifies the end of the input data.

III. Explanatory Notes Regarding Program Input

• *NOTE 1 : Summary of the Analysis Procedure*

The PLATE program represents the numerical implementation of the thin plate analysis described in the paper “Finite Element Bending Analysis of Plates”, by L. R. Herrmann (*Proceedings of the ASCE*, Vol. 94, No. EM5, pp. 13-25, 1968). For a description of the theory underlying the analysis, the analyst is referred to this paper.

The plate is represented by a series of 8-node quadrilateral elements (Figure 3.1). Except for the restriction that the “center” point (its coordinates are the average of the coordinates of the four vertex points) must lie within the element (if this is not the case, the program prints an error message). The positive coordinate system is defined in Figure 3.1; hence positive loads are directed upward, a positive transverse displacement is upward, etc. Each element is described by eight nodes; i.e., the four vertex nodes n_1, n_3, n_5, and n_7, and by the four mid-side nodes n_2, n_4, n_6, and n_8 (Figure 3.1).

Within a given element, the material properties (E, ν), the thickness of the plate (h), the transverse load per unit area (q), and the thermal moment (m_T) are approximated as constants. These quantities may, however, be varied from element to element.

Displacement and/or effective shear boundary conditions must be specified at each vertex node along the boundary of the plate; moment or slope boundary conditions must be specified for all mid-side nodes along the boundary. In addition, values of displacement or concentrated loads may be specified at interior vertex nodes.

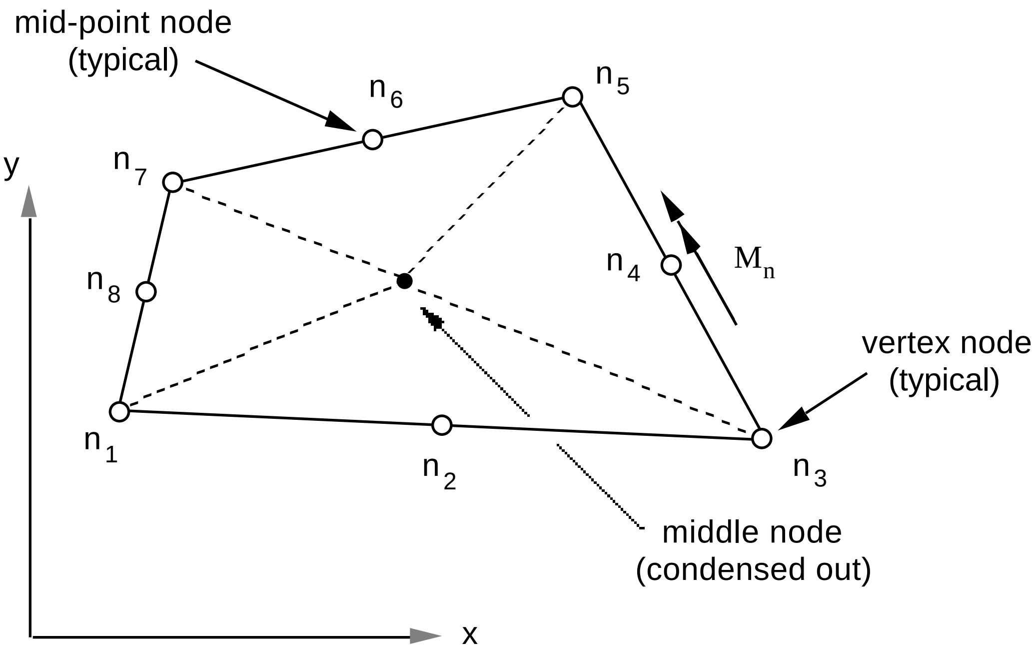

Figure 3.1. Typical Plate Bending Element

Block Input Section

With the exception of the alphanumeric "identification record", the input data for a given analysis occurs in a block mode, i.e., all the nodal data (for example) is placed in one block, all the element data in another, etc. Each block is preceded by an input record with a single integer in the range 1 through 5 in the first field. This integer serves to instruct the program as to the type of data that follows. However, because a list-directed READ statement (used for input to the program) will ignore line boundaries, each record containing the integer flag for a new block of data must be padded with sufficient zeros so that the record contains at least as much data as the previous records (from the previous block). More precisely, consider the following case: The input block for nodal coordinates is finished and input to the element data block is to be read:

Case 1: (error)

`0 65 2.38 5.39 4 1.0 0.0 0.0`	(last nodal coordinate record)
`2`	(integer flag for element data)
`0 1 6 7 2 3 1 2 4 5 2 3`	(1st record for element data)

The program would read the 2 (indicating that element data follows), and in addition, would continue until it exhausts the input list (for node coordinates), somewhere towards the end of the 1st element record. At this point, the program and the analyst would have different opinions as to what problem is being solved.

This difficulty is easily solved in one of three ways. In the first procedure, the input record with the integer flag is padded with sufficient zero fields so that the input list (for node coordinates, in this case) is exhausted and the next read statement finds the 1st record for patch data:

Case 2: (correct - method #1)

```
0   65    2.38  5.39  4     1.0   0.0   0.0       (last nodal coordinate record)
2    0      0      0   0      0      0      0       (integer flag for element data)
0   1   6   7   2   3   1   2   4   5    2    3     (1st record for element data)
```

This method is portable, easy to implement, and is used in all the problems in the Sample Analysis section of this document. A second method can be used in environments where a mechanism for terminating an input list is available. For instance, a (/) character will terminate the read list on many computers:

Case 3: (correct - method #2)

```
0   65    2.38  5.39  4     1.0   0.0   0.0       (last nodal coordinate record)
2    /                                              (integer flag for element data)
0   1   6   7   2   3   1   2   4   5    2    3     (1st record for element data)
```

This second method helps to delineate the block structure of the input file; a check of the appropriate FORTRAN–77 documentation will uncover any details needed for a particular computer.

Finally, the third method is the safest and may be the easiest to use: the flag record consists of the integer flag followed by 11 zero fields:

```
2  0  0  0  0  0  0  0  0  0  0  0
```

This method is motivated by the fact that the longest input record is 12 fields wide (element information). Therefore, a flag record with the flag in field one and zeros in the next 11 fields will always separate any two data blocks. For users accustomed to typical FORTRAN formatted input records, this method is recommended.

Also note that blocks must occur in the order presented in the “Program Input” section of this document.

The popularity of finite element solutions to systems of partial differential equations, especially in engineering fields, has necessitated the direction of considerable energies toward efficient program data input. The finite element method requires the sub-division of the solution domain into a finite number of non-overlapping geometrical sub-domains (usually triangles or quadrilaterals in two-dimensional space). Since the detailed preparation of such meshes may be quite laborious, semi-automatic generation procedures have been developed. The procedures available in this program are described in the following paragraphs.

The region to be analyzed is represented by a series of quadrilateral elements, e.g., See Figure 3.2. The mesh is defined by the location of the nodes and the sets of eight node numbers that define the elements.

Attaining a satisfactory mesh is an evolutionary process, that often requires consideration of a number of alternatives before desired results are obtained. It is recommended that initially a rough sketch be made of the body with nodes placed at approximately their desired locations. The nodes and elements should be numbered on this sketch. A proper numbering scheme for the nodes is extremely important for the minimization of computational cost of a finite element analysis. For a given element denote the greatest difference between the numbers of any two of the eight nodes that define the element as $N_{i.}$ Denote the maximum value of N_i for the whole system as N_{max} (the numbering used in Figure 3.2 gives a value of N_{max} = 13; if the numbering had instead proceeded from left to right a value of N_{max} = 19 would have been obtained). In numbering a mesh, computational efficiency requires a minimum value of N_{max} .

The sketch of the proposed mesh is used to determine how the available generation procedures can be best applied to assist in the preparation of the input data.

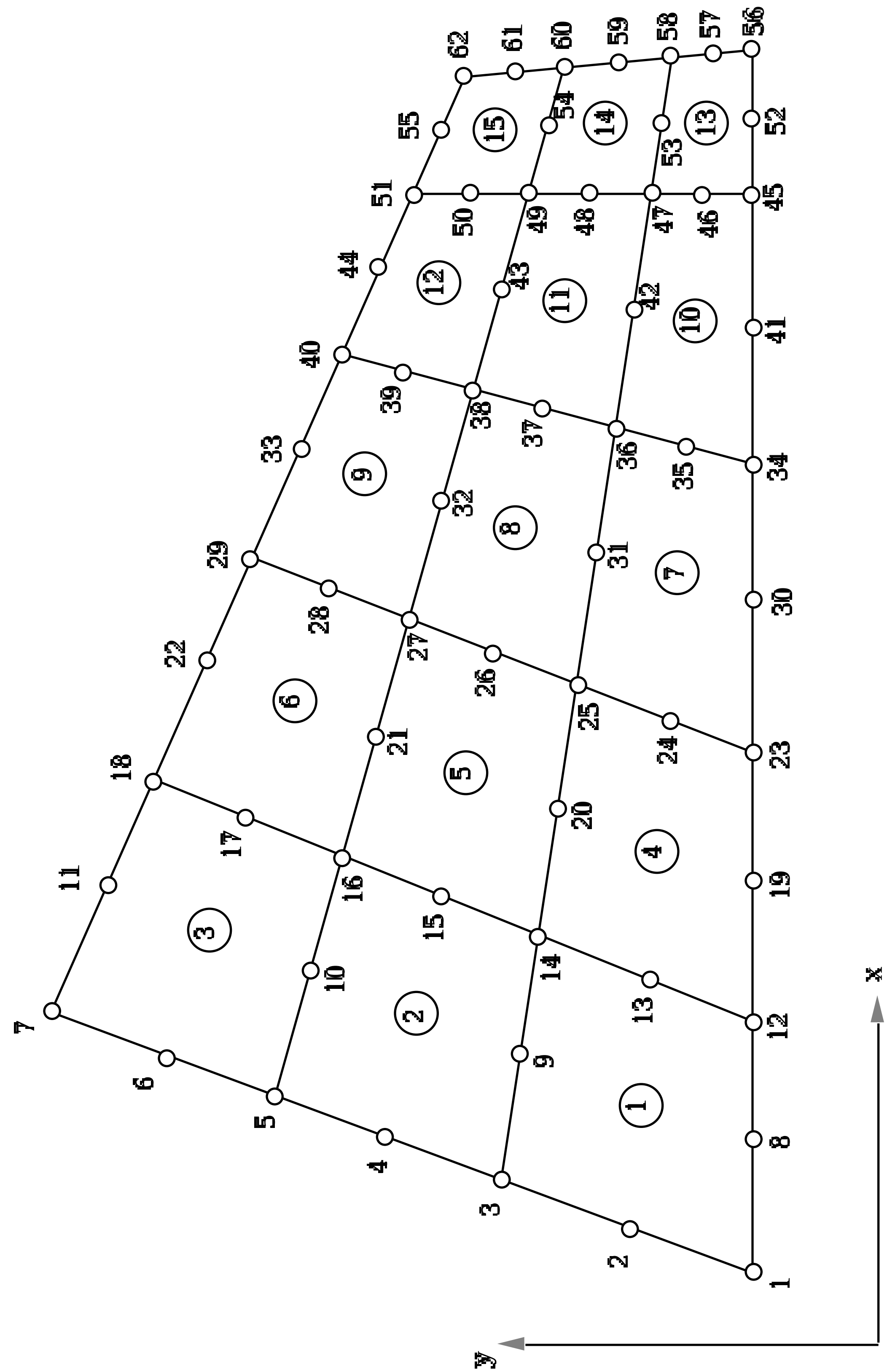

Figure 3.2
Sample Finite Element Mesh

• *NOTE 2 : Data Generation Options*

The program has available two generation procedures to assist the user in describing the location of the node points. The use of these options can, for instance, permit one to describe the location of all the nodes for an arbitrarily large mesh by as few a five records.

• *NOTE 3 : Generation of Boundary Node Points*

The "circular arc (or straight line) coordinate generation option" may be used whenever several sequential vertex points (either interior or exterior points) lie along a circular arc or a straight line. For such point it is only necessary to enter the coordinates for the end points (denoted as N and N') of the sequence and the required generation information **INCR**, **RATIO** and (**XC**, **YC**). The generation information is entered on the second of the two records, i.e., on the data record for N'. (It is to be noted that the record for N' can also serve as a beginning record for the second segment N' → N"; the generation data for this segment would be contained on the record for N", etc.). The quantity **INCR** represents the difference between the numbers of any two successive vertex nodes in the sequence and **RATIO** denotes the ratio of the distances between successive pairs of vertex points. For circular arcs **XC** and **YC** represent the coordinates of some intermediate point (it need not be a node point) on the arc. Thus, for a point N' if **INCR** ≠ 0, then intermediate points are generated along a straight line (if **XC** = **YC** = 0) or along a circular arc (**XC** ≠ 0 and/or **YC** ≠ 0) between N' and the point described on the previous node record (N). That is, vertex points N + **INCR**, N + 2***INCR**, ... , N' – **INCR** are generated. If the segment is a circular arc, it is defined as passing through the end points N and N' and some intermediate point (not necessarily one of the nodes) whose coordinates are (**XC**, **YC**).

The ends of the segment may be entered in any order, i.e., the segment shown in Figure 3.3 may be defined by specifying the end-points in order 7 → 22 or 22 → 7. The spacing of the intermediate vertex node points is controlled by the value of the spacing ratio **RATIO**. **RATIO** is equal to the ratio of the lengths of the successive segments defined by the intermediate vertex node points. A value of **RATIO** = 1.0 gives equally spaced vertex points. The locations of the intermediate vertex points 12 and 17 (see Figure 3.3), could be generated by either specifying points 7 → 22 and **RATIO** = 2.0 (*NOTE*: **RATIO** = 2.0/1.0 = 4.0/2.0), or 22 → 7 and **RATIO** = 0.5 (*NOTE*: **RATIO** = 2.0/4.0 = 1.0/2.0); the value of **INCR** would be 5.

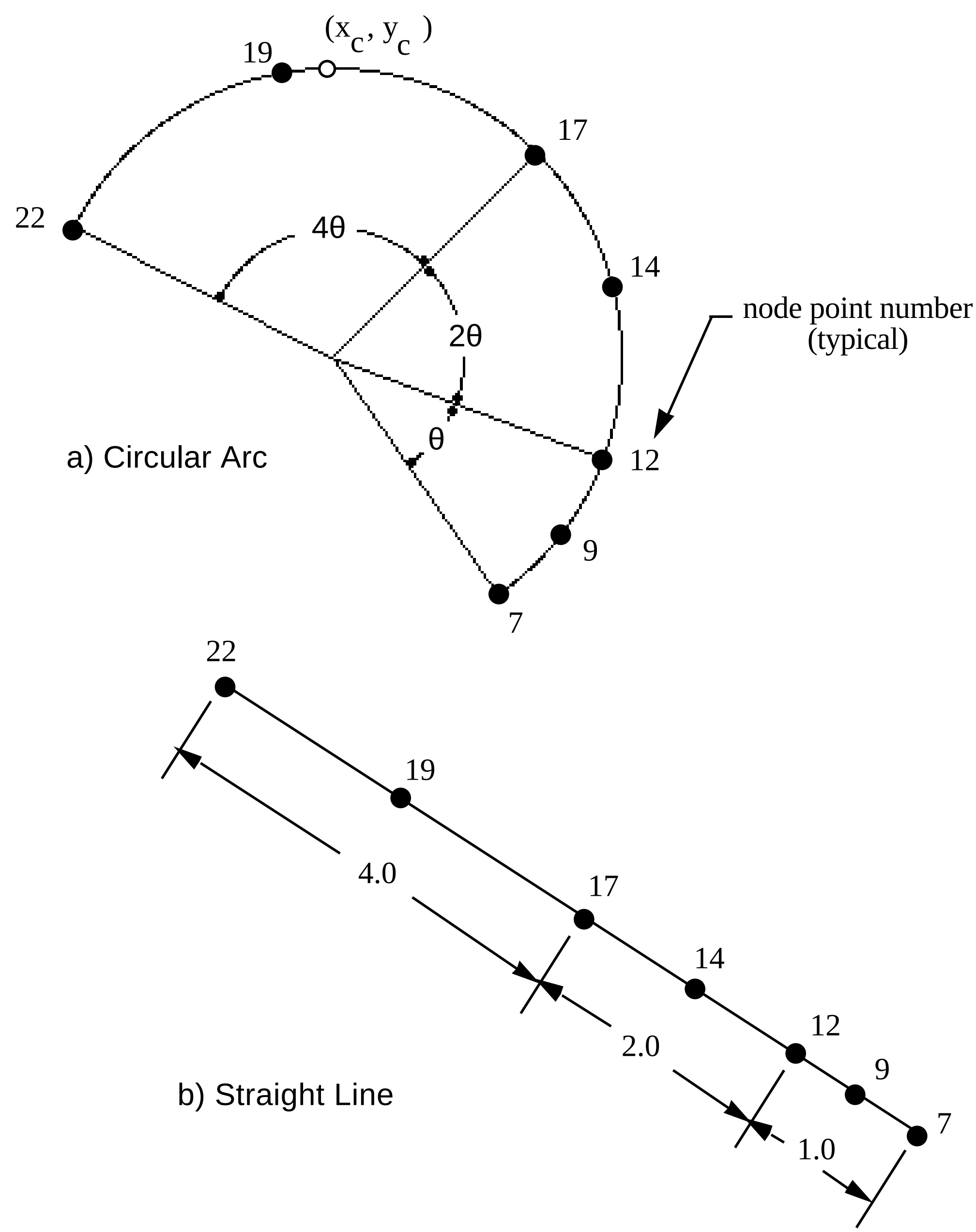

Figure 3.3. Examples of Node Point Generation

For the mesh shown in Figure 3.2, the line segment generation procedure could be used to locate exterior points 1 → 56, 56 → 62, 62 → 7, 7 → 1. In addition if desired it could be used to locate interior nodes 12 → 18, etc.

Sometimes when using this generation procedure it is necessary to enter a record for a node whose coordinates have already been specified or generated, e.g., one might generate 51 → 7 (Figure 3.2) and then wish to generate 18 → 12. In such cases one need not enter the coordinates for the point a second time, instead the number of the point is entered as *negative* and the values specified for **XP**, **YP** are immaterial.

• *NOTE 4 : Generation of Unknown Interior Nodal Coordinates*

The "interior node point generation option" locates all vertex nodes interior to the body whose coordinates have not been specified directly by the user (either by directly inputting the coordinates or by using the line segment generation option). The interior nodes are located by means of a generation scheme described in the paper "Laplacian–Isoparametric Grid Generation Scheme" (L. R. Herrmann 1976, Journal of the Engineering Mechanics Division, ASCE, Vol. 102, NO. EM5).

Recall that all vertex nodes on the boundaries of the body must be either directly specified or generated by means of the line or arc segment generation scheme. In addition in certain situations it may be desirable, or even necessary, to individually or by means of the line generation option to specifically locate certain interior vertex points. This direct location of interior points may be necessary in order to describe an interface between portions of the plate with differing thicknesses or properties, and/or to improve the shape of the generated mesh.

• *NOTE 5 : Specification of Element Information*

Once the coordinates of the vertex nodes have been specified or generated, the last task is to specify the numbers of the eight nodes defining each element. Essentially two alternatives are provided. One of them is to enter the information element by element, the other is to generate it internally within the program. Internal element data generation is again dependent on specific mesh numbering patterns, namely repeating node number increments.

• *NOTE 6 : Generation of Element Information*

To utilize the "element data generation option", it must be possible to divide the mesh into layers. A layer of elements is defined as a series of elements for which six of the eight node numbers of adjacent elements differ by two and the other two by one, e.g., the node numbers for elements 1 and 2 of Figure 3.2 are:

```
1   8   12   13   14   9   3   2
3   9   14   15   16   10   5   4
```

Thus elements 1 → 3, 4 → 6, etc., of Figure 3.2 each constitute a layer. Let the number of elements in each layer be **NMIS** + 1. If the mesh is regular then the corresponding node numbers of elements in adjacent layers will differ by a constant; denote this numbering increment by **INCRP** and the total number of layers by **NMISP** + 1. Thus the element information for the mesh shown in Figure 3.2 can be completely described with one record containing the eight node numbers of element 1 and **NMIS** = 2, **NMISP** = 4, **INCRP** = 11. Two records would be required for the example shown in Figure 3.4. The first record would contain the nodes defining element 1 and **NMIS** = 2, **NMISP** = 2, **INCRP** = 11. The second record would contain the node numbers for element 10 and **NMIS** = 3, **NMISP** = 1, **INCRP** = 14. Note when using this generation procedure either **NMIS** or **NMISP** may be zero.

The element numbers are assigned by the generation procedure beginning with the elements of the first layer then proceeding to the second, etc.

As was noted earlier displacement or transverse shear boundary conditions are specified for vertex points and moment or slope conditions for mid-points.

• *NOTE 7 : Nodal Specifications*

For vertex points on the boundary of the body one either: (i) specifies the transverse displacement by setting **IFLAG** equal to one (1) and **VALUE** equal to the specified displacement; or, (ii) adds in the “effective shear load” by setting **IFLAG** equal to zero (0) and **VALUE** equal to the load (if the load is zero a boundary card is not necessary). The “effective shear load” at a vertex point on the boundary can be approximated as the total shear load applied on the segment of the boundary between the two adjacent mid-points.

At each mid-point on the boundary of the body one either: (i) specifies the normal moment by setting **IFLAG** equal to one (1) and **VALUE** equal to the specified moment; or, (ii) adds in the “normal rotation effect” by setting **IFLAG** equal to zero (0) and **VALUE** equal to the “rotation effect” (if the “rotation effect” is zero a boundary record is *not* necessary). The “rotation effect” is equal to the average value of the specified normal rotation along the element side multiplied by the length of the element side.

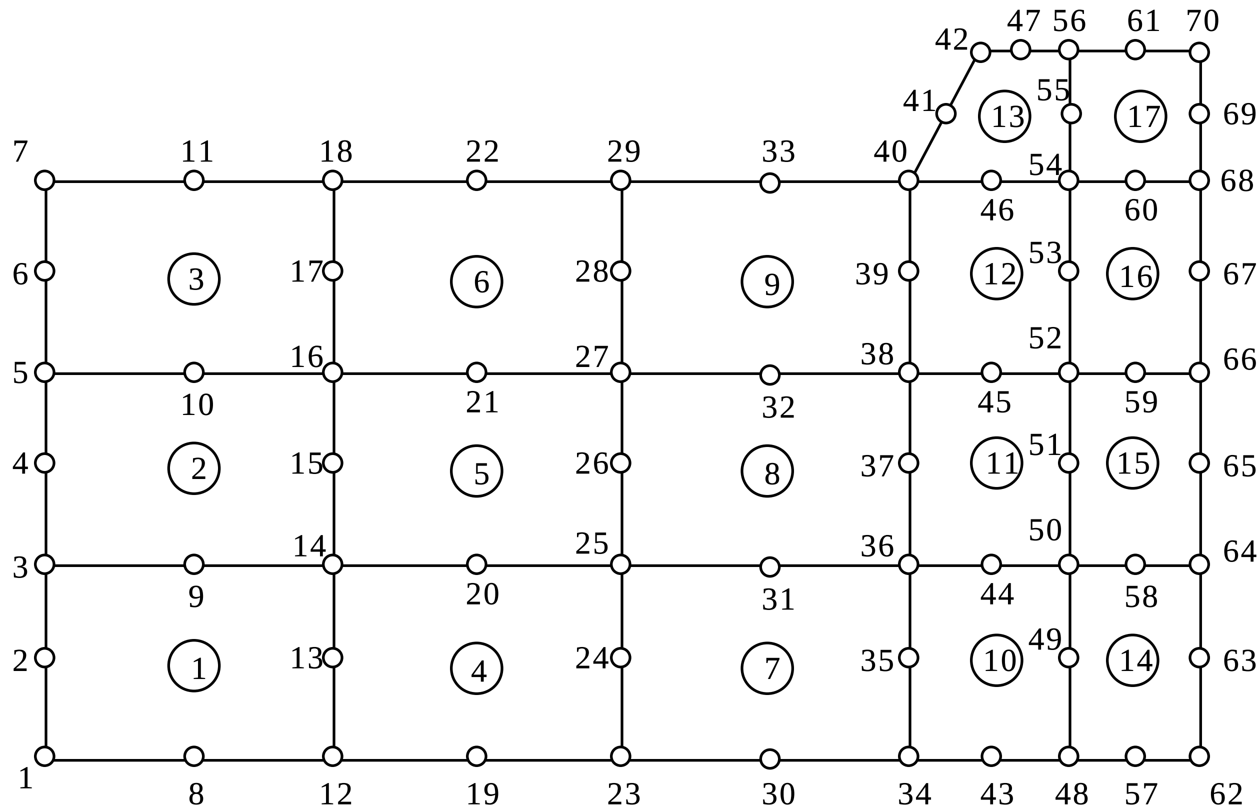

Figure 3.4. Sample Mesh with Variable Number of Elements in the Layers

• *NOTE 8 : Generation of Nodal Specifications*

Whenever the specified boundary condition information is identical for several sequential boundary points (denote the first point in the sequence be N and the last by N') whose successive numbers differ by a constant **INCR**, the only one record is necessary. The record containing the information for N is also supplied with the values for N' and **INCR**.

IV. Program Output

The output of the PLATE program includes the following information: First the coordinates of the vertex nodes are printed. Secondly, a summary of input data for each element is printed. This is followed by a list of boundary conditions for both vertex- and mid-side nodes. Next, for each element the following information is printed: a) the coordinates of the center point; b) the transverse displacement of the center point; and, c) the bending and twisting moment intensities at the center point. Fifth, the transverse displacement is printed for all the vertex nodes. Finally, the values of the normal moments (i.e., normal to the element sides in the manner shown in Figure 3.1) are printed for the mid-side nodes.

V. Sample Analyses

Sample Analysis 1 : simply supported, uniformly loaded plate

A square plate 20 inches on a side and 0.200 inches thick is simply supported along all four edges. A uniform (downward) load of 10 psi is applied to the plate. The plate's material is characterized by the constants $E = 30 \times 10^6$ and $\nu = 0.30$.

The above plate shall be analyzed using the PLATE computer program. In creating the mathematical model, note that due to symmetry, only one-quarter of the plate (in this case, the lower left portion of the plate) needs to be considered in the finite element analysis. The mesh used is shown in Figure 5.1.

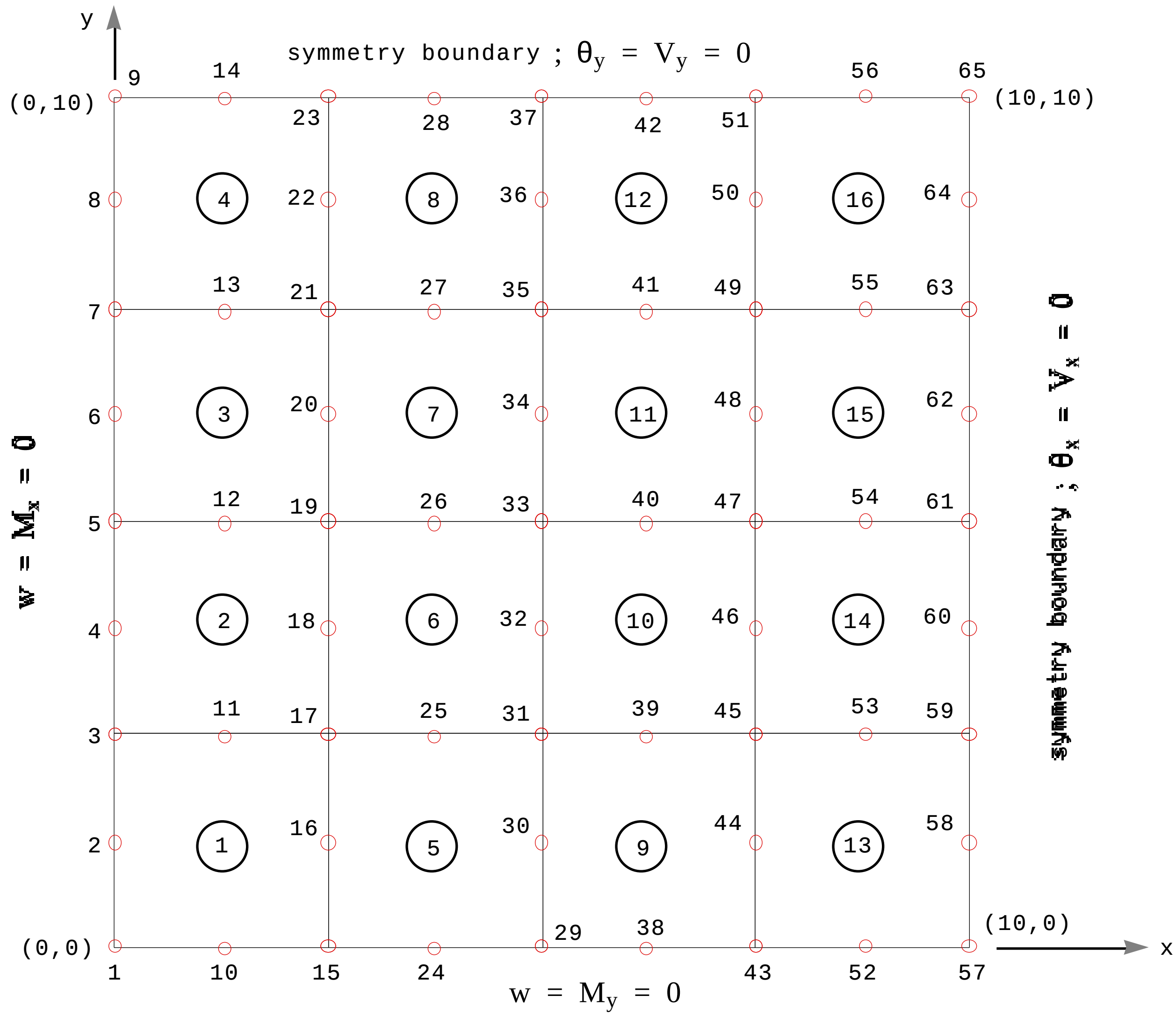

Figure 5.1. Mesh Associated with Sample Analysis #1

The input data file associated with the analysis is shown below.

```
square plate, s.s. on all four edges, uniformly loaded
1
0    1       0.0       0.0    0       1.0       0.0       0.0
0    9       0.0      10.0    2       1.0       0.0       0.0
0   65      10.0      10.0   14       1.0       0.0       0.0
0   57      10.0       0.0   -2       1.0       0.0       0.0
0   -1       0.0       0.0  -14       1.0       0.0       0.0
2   0   0   0   0   0   0   0   0   0   0   0   0
0   1   10   15   16   17   11    3    2    3    3   14
0    30.0e+06       0.3     0.200     -10.0       0.0
3   0   0   0   0   0   0   0   0   0   0   0   0
0   1    1       0.0   57   14
0   1    1       0.0    9    2
4   0   0   0   0   0   0   0   0   0   0   0   0
0     2    1       0.0    8    2
0    10    1       0.0   52   14
5   0   0   0   0   0   0   0   0   0   0   0   0
```

The results obtained by using the PLATE computer program in conjunction with the above data are given on the following page.

square plate, s.s. on all four edges, uniformly loaded

***** GEOMETRY *****

Node Point	X-Y Coordinates	
1	0.00E+00	0.00E+00
3	0.00E+00	2.50E+00
5	0.00E+00	5.00E+00
7	0.00E+00	7.50E+00
9	0.00E+00	1.00E+01
15	2.50E+00	0.00E+00
17	2.50E+00	2.50E+00
19	2.50E+00	5.00E+00
21	2.50E+00	7.50E+00
23	2.50E+00	1.00E+01
29	5.00E+00	0.00E+00
31	5.00E+00	2.50E+00
33	5.00E+00	5.00E+00
35	5.00E+00	7.50E+00
37	5.00E+00	1.00E+01
43	7.50E+00	0.00E+00
45	7.50E+00	2.50E+00
47	7.50E+00	5.00E+00
49	7.50E+00	7.50E+00
51	7.50E+00	1.00E+01
57	1.00E+01	0.00E+00
59	1.00E+01	2.50E+00
61	1.00E+01	5.00E+00
63	1.00E+01	7.50E+00
65	1.00E+01	1.00E+01

ELEMENT INFORMATION

	Node Point number								Youngs modulus	Poissons ratio	thickness	normal load	thermal load
1	1	10	15	16	17	11	3	2	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
2	3	11	17	18	19	12	5	4	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
3	5	12	19	20	21	13	7	6	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
4	7	13	21	22	23	14	9	8	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
5	15	24	29	30	31	25	17	16	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
6	17	25	31	32	33	26	19	18	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
7	19	26	33	34	35	27	21	20	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
8	21	27	35	36	37	28	23	22	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
9	29	38	43	44	45	39	31	30	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
10	31	39	45	46	47	40	33	32	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
11	33	40	47	48	49	41	35	34	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
12	35	41	49	50	51	42	37	36	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
13	43	52	57	58	59	53	45	44	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
14	45	53	59	60	61	54	47	46	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
15	47	54	61	62	63	55	49	48	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00
16	49	55	63	64	65	56	51	50	3.000E+07	3.000E-01	2.000E-01	-1.000E+01	0.000E+00

BOUNDARY CONDITIONS

Boundary Node		
1	W=	0.00000E+00
15	W=	0.00000E+00
29	W=	0.00000E+00
43	W=	0.00000E+00
57	W=	0.00000E+00
1	W=	0.00000E+00
3	W=	0.00000E+00
5	W=	0.00000E+00
7	W=	0.00000E+00
9	W=	0.00000E+00
2	M=	0.00000E+00
4	M=	0.00000E+00
6	M=	0.00000E+00
8	M=	0.00000E+00
10	M=	0.00000E+00
24	M=	0.00000E+00
38	M=	0.00000E+00
52	M=	0.00000E+00

ELEMENT UNKNOWNS

no	X	Y	W	M-x	M-y	M-xy
1	1.250E+00	1.250E+00	-1.368E-02	-1.404E+01	-1.405E+01	1.246E+02
2	1.250E+00	3.750E+00	-3.723E-02	-3.403E+01	-3.076E+01	9.747E+01
3	1.250E+00	6.250E+00	-5.385E-02	-4.582E+01	-3.848E+01	6.093E+01
4	1.250E+00	8.750E+00	-6.238E-02	-5.135E+01	-4.167E+01	2.064E+01
5	3.750E+00	1.250E+00	-3.723E-02	-3.075E+01	-3.403E+01	9.747E+01
6	3.750E+00	3.749E+00	-1.029E-01	-8.076E+01	-8.076E+01	7.932E+01
7	3.750E+00	6.249E+00	-1.498E-01	-1.122E+02	-1.032E+02	5.046E+01
8	3.750E+00	8.750E+00	-1.739E-01	-1.273E+02	-1.126E+02	1.720E+01

9	6.250E+00	1.250E+00	-5.384E-02	-3.848E+01	-4.581E+01	6.093E+01
10	6.250E+00	3.749E+00	-1.497E-01	-1.032E+02	-1.122E+02	5.046E+01
11	6.250E+00	6.249E+00	-2.188E-01	-1.464E+02	-1.464E+02	3.266E+01
12	6.250E+00	8.750E+00	-2.546E-01	-1.678E+02	-1.607E+02	1.123E+01
13	8.750E+00	1.250E+00	-6.237E-02	-4.166E+01	-5.135E+01	2.064E+01
14	8.750E+00	3.750E+00	-1.739E-01	-1.125E+02	-1.273E+02	1.720E+01
15	8.750E+00	6.250E+00	-2.546E-01	-1.607E+02	-1.678E+02	1.123E+01
16	8.750E+00	8.750E+00	-2.965E-01	-1.850E+02	-1.850E+02	3.878E+00

Vertex node	W
1	0.00000E+00
3	0.00000E+00
5	0.00000E+00
7	0.00000E+00
9	0.00000E+00
15	0.00000E+00
17	-5.05924E-02
19	-9.01857E-02
21	-1.14947E-01
23	-1.23338E-01
29	0.00000E+00
31	-9.01704E-02
33	-1.61991E-01
35	-2.07258E-01
37	-2.22640E-01
43	0.00000E+00
45	-1.14927E-01
47	-2.07247E-01
49	-2.65775E-01
51	-2.85713E-01
57	0.00000E+00
59	-1.23338E-01
61	-2.22639E-01
63	-2.85713E-01
65	-3.07222E-01

Mid-point node	M-N
2	0.00000E+00
4	0.00000E+00
6	0.00000E+00
8	0.00000E+00
10	0.00000E+00
11	-2.28146E+01
12	-3.33750E+01
13	-3.83339E+01
14	-3.97902E+01
16	-2.28629E+01
18	-6.28522E+01

```
20          -8.64261E+01
22          -9.74983E+01
24           0.00000E+00
25          -6.28392E+01
26          -9.34470E+01
27          -1.07834E+02
28          -1.12062E+02
30          -3.34033E+01
32          -9.34417E+01
34          -1.32711E+02
36          -1.51910E+02
38           0.00000E+00
39          -8.64240E+01
40          -1.32718E+02
41          -1.54877E+02
42          -1.61410E+02
44          -3.83288E+01
46          -1.07823E+02
48          -1.54872E+02
50          -1.78483E+02
52           0.00000E+00
53          -9.74992E+01
54          -1.51911E+02
55          -1.78484E+02
56          -1.86354E+02
58          -3.97894E+01
60          -1.12063E+02
62          -1.61410E+02
64          -1.86352E+02


     <<<<<  Solution Successfully Completed  >>>>>
```