BAB VII

KESIMPULAN DAN SARAN

VII.1 Kesimpulan

Berdasarkan studi kasus yang telah dilakukan pada bab-bab sebelumnya, maka dapat ditarik beberapa hal-hal penting sebagai berikut:

a. formula dalam bentuk integrasi dari gaya jepit ujung akibat beban terbagi merata, beban segi tiga, beban trapesium, dan beban terpusat pada batang yang lebar dan tingginya non-prismatis telah berhasil diturunkan menggunakan sistem persamaan dalam metode kerja minimum, dan solusi integrasinya diselesaikan menggunakan metode Gauss Kuadratur 15 titik;
b. solusi numerik pada formula gaya jepit ujung yang diturunkan tersebut sangat memuaskan dengan persentase beda minimum 0.0143% dan persentase beda maksimum 1.8902% terhadap program pembanding SAP2000 v 7.42;
c. *software* "n-Prime 2004 Project" ini berhasil dikompilasi menggunakan *Microsoft Visual Basic 6.0 Enterprise Edition* yang *source codes*-nya dapat dilihat pada lampiran 7;
d. tampilan grafis untuk *joint displacement*, NFD, SFD, dan BMD juga telah berhasil dibuat dalam salah satu sub-program *software* "n-Prime 2004 Project" dengan cara membagi panjang batang menjadi pias-pias yang sangat kecil untuk kemudian ditinjau besar gaya aksial, gaya geser, dan momen pada pias tesebut;
e. *software* "n-Prime 2004 Project" ini cukup akurat untuk menghitung *joint displacement* dan besar gaya-gaya dalam pada batang prismatis dan batang non-

prismatis *linear*. Persentase beda *joint displacement* yang mencapai 50.3833%, dan persentase beda gaya-gaya batang yang mencapai 68.9612% dapat terjadi karena besar perpindahannya yang sudah mendekati nol, dan terutama diakibatkan oleh perbedaan persamaan lengkung parabola yang digunakan dalam "n-Prime 2004 Project" dan SAP2000 v 7.42. Persamaan lengkung parabola dalam SAP2000 v 7.42 tidak diketahui karena *input*-nya hanya berupa lebar dan tinggi penampang melintang pada ujung-ujung batang, sehingga lebar dan tinggi penampang melintang pada tengah bentang tidak diketahui. Pendekatan yang dilakukan dalam rangka mendekati persamaan lengkung parabola dan memperoleh persentase beda yang kecil dilakukan dengan membagi-bagi batang non-prismatis menjadi beberapa batang yang lebih kecil (diskretisasi);

f. persamaan kelengkungan parabola untuk batang yang non-prismatis pada lebar dan tingginya mempunyai pengaruh yang sangat besar terhadap hasil analisis struktur yang dilakukan.

VII.2 Saran

Beberapa saran yang kiranya dapat disampaikan adalah *software* "n-Prime 2004 Project" dapat dikembangkan lagi menuju ke arah desain penulangan untuk tiap-tiap batang yang dipakai. Selain itu, kiranya *software* "n-Prime 2004 Project" ini dapat dikembangkan lebih lanjut ke arah analisis portal ruang atau rangka batang /*plane-truss*. Aspek permasalahan perbedaan persamaan kelengkungan parabolis yang dipakai dalam kedua *software* ini dapat dianalisis dan diteliti lebih jauh lagi.

DAFTAR PUSTAKA

Carroll, W. F., *A Primer for Finite Elements in Elastic Structures*, John Wiley & Sons, Inc., 1999.

Cook, R. D., *Concepts and Applications of Finite Element Analysis*, John Wiley &Sons, Inc., 1981.

Ghali, A., dan Neville, A. M., *Structural Analysis: A Unified Classical and Matrix Approach*, Chapman and Hall, London, 1978.

Hsieh, Y. Y., *Teori Dasar Struktur*, P.T. Erlangga, Jakarta, 1985.

Kardestuncer, H., *Elementary Matrix Analysis of Structures*, McGraw-Hill, New York, 1974.

Pribadi, Arif Susanto, *Analisis Portal Bidang dengan Batang Tak Prismatis pada Lebar dan Tingginya*, Yogyakarta, 2003.

Weaver, W., Jr., dan Gere, J. M., *Matrix Analisys of Framed Structures*, D. Van Nostrand Company, 1980.

Weaver, W., Jr., dan Johnston, P. R., *Elemen Hingga untuk Analisis Struktur*, P.T. Eresco, Bandung, 1989.

***Input* Data Struktural Kasus 1**
Nama *file*: kasus1a.txt

```
3     4     6     2     1e6
1     0     0
2     0     5
3     5     5
4     5     0
1     1     2     0.2   0.2   0.2   0.2   0.2   0.2
2     2     3     0.2   0.2   0.2   0.2   0.2   0.2
3     3     4     0.2   0.2   0.2   0.2   0.2   0.2
1     1     1     1
4     1     1     1
```

***Input* Data Beban Kasus 1**
Nama *file*: kasus1b.txt

```
1     1
2     5     0     0
2     20    0     20    5     0     0     0     0     0     0     0     0
```

***Output* Kasus 1**
Nama *file*: kasus1c.txt

n-Prime 2004 Project
[D:\Skripsi Stevie\New Folder\inputTA\kasus1c.txt]
Unit: kN - m

Frame Element Fixed-End Forces

Frame ID	R-i	M-i	R-j	M-j
2	50.00002	41.66672	50.00002	-41.66672

Joint Displacements

Joint ID	Ux	Uy	Rz
1	0	0	0
2	0.2799322	-5.98226E-03	-0.2941272
3	0.2785785	-6.517745E-03	0.2269774
4	0	0	0

Frame Element Forces

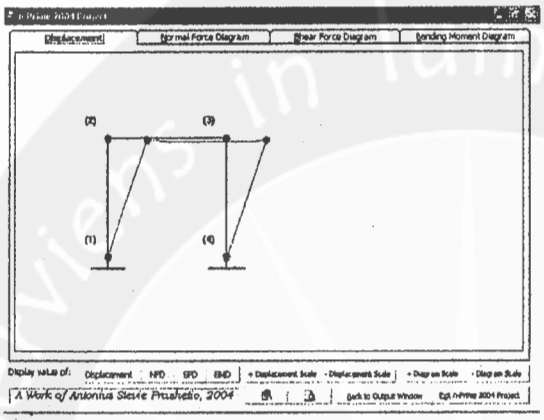
Frame ID	Axial-i	Shear-i	Moment-i	Axial-j	Shear-j	Moment-j
1	-47.85808	-5.82894	6.728957	-47.85808	-5.82894	-22.41574
2	-10.82897	47.85808	-22.41574	-10.82897	-52.14196	-33.12544
3	-52.14196	10.82908	-33.12544	-52.14196	10.82908	21.01998

Support Reactions

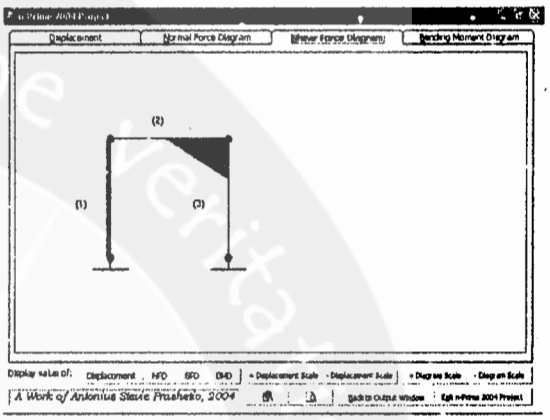
Joint ID	Fx	Fy	Mz
1	5.82894	47.85808	-6.728957
4	-10.82908	52.14196	21.01998

Tampilan Grafis *Joint Displacement*, NFD, SFD, dan BMD kasus 1

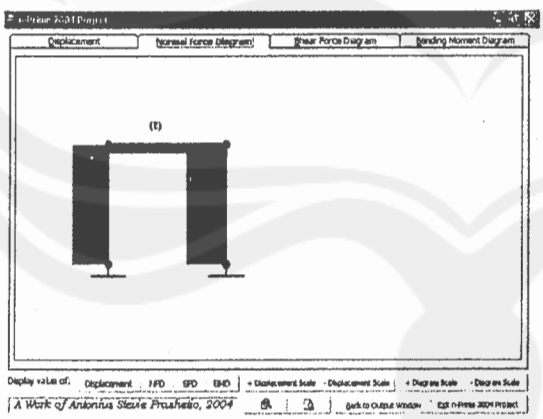
Joint Displacement



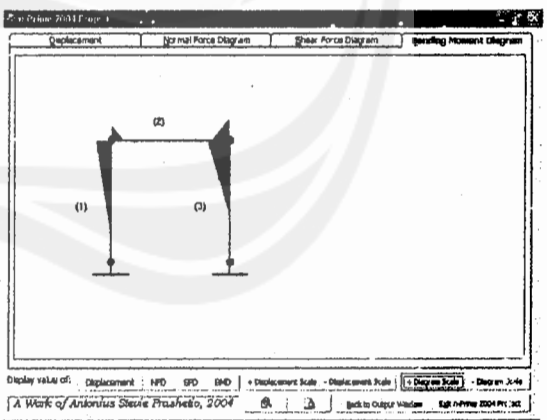
Shear Force Diagram



Normal Force Diagram



Bending Moment Diagram



Input Tables SAP2000 Kasus 1
Nama *file*: val1a.txt

SAP2000 v7.42 File: KASUS1 KN-m Units PAGE 1
5/20/04 17:26:11

Antonius Stevie Prashetio

S T A T I C L O A D C A S E S

STATIC CASE	CASE TYPE	SELF WT FACTOR
LOAD1	DEAD	0.0000

J O I N T D A T A

JOINT	GLOBAL-X	GLOBAL-Y	GLOBAL-Z	RESTRAINTS	ANGLE-A	ANGLE-B	ANGLE-C
1	0.00000	0.00000	0.00000	1 1 1 1 1 1	0.000	0.000	0.000
2	0.00000	0.00000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
3	5.00000	0.00000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
4	5.00000	0.00000	0.00000	1 1 1 1 1 1	0.000	0.000	0.000

F R A M E E L E M E N T D A T A

FRAME	JNT-1	JNT-2	SECTION	ANGLE	RELEASES	SEGMENTS	R1	R2	FACTOR	LENGT
1	1	2	200X200	0.000	000000	2	0.000	0.000	1.000	5.00
2	2	3	200X200	0.000	000000	4	0.000	0.000	1.000	5.00
3	3	4	200X200	0.000	000000	2	0.000	0.000	1.000	5.00

J O I N T F O R C E S Load Case LOAD1

JOINT	GLOBAL-X	GLOBAL-Y	GLOBAL-Z	GLOBAL-XX	GLOBAL-YY	GLOBAL-ZZ
2	5.000	0.000	0.000	0.000	0.000	0.000

F R A M E S P A N D I S T R I B U T E D L O A D S Load Case LOAD1

FRAME	TYPE	DIRECTION	DISTANCE-A	VALUE-A	DISTANCE-B	VALUE-B
2	FORCE	GLOBAL-Z	0.0000	-20.0000	1.0000	-20.0000

Output Tables SAP2000 Kasus 1
Nama file: val1b.txt

SAP2000 v7.42 File: KASUS1 KN-m Units PAGE 1
5/20/04 17:25:46

Antonius Stevie Prashetio

J O I N T D I S P L A C E M E N T S

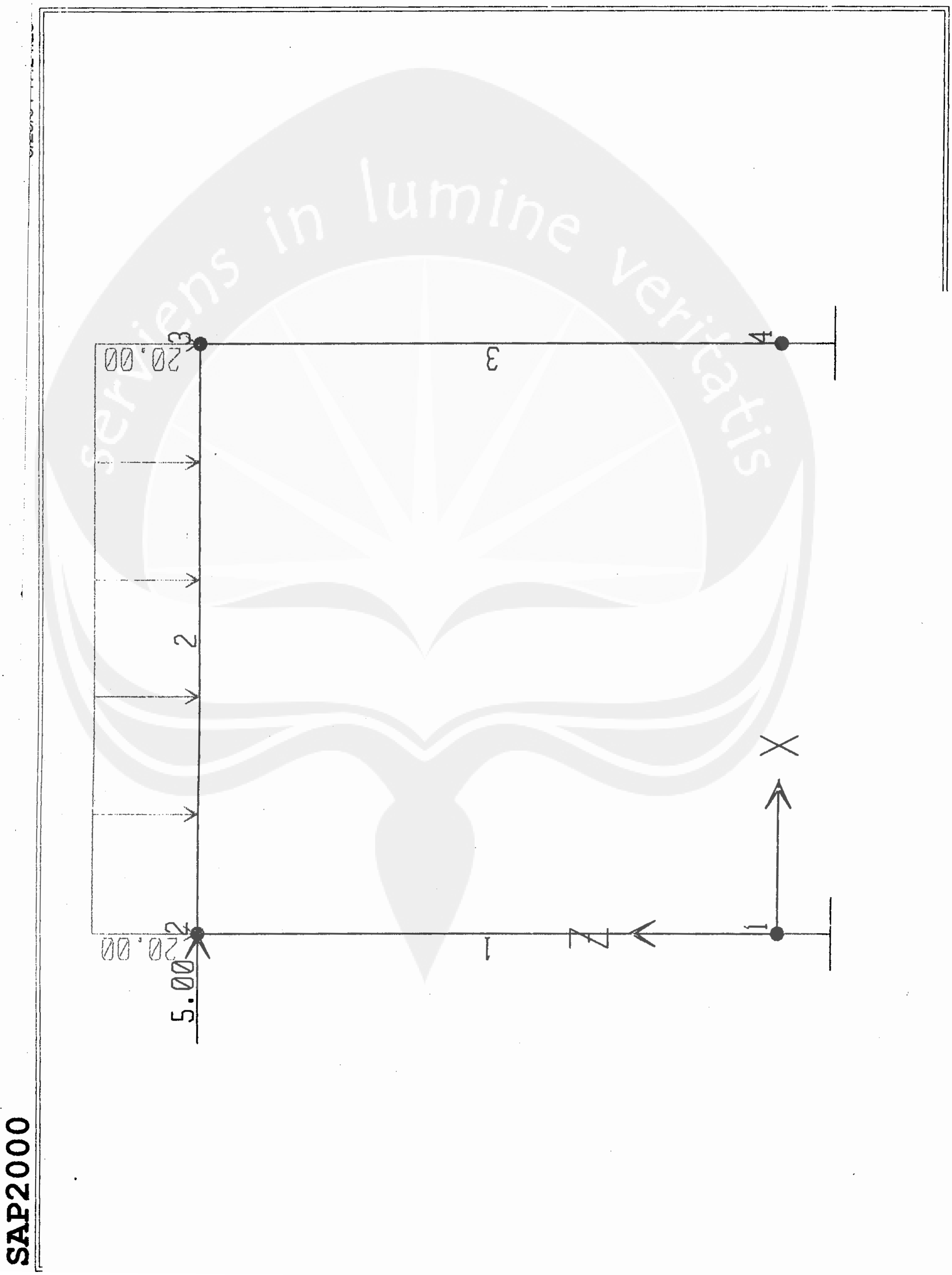
JOINT	LOAD	U1	U2	U3	R1	R2	R3
1	LOAD1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	LOAD1	0.2812	0.0000	-5.982E-03	0.0000	0.2949	0.0000
3	LOAD1	0.2798	0.0000	-6.518E-03	0.0000	-0.2274	0.0000
4	LOAD1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

J O I N T R E A C T I O N S

JOINT	LOAD	F1	F2	F3	M1	M2	M3
1	LOAD1	5.8099	0.0000	47.8595	0.0000	6.6618	0.0000
4	LOAD1	-10.8099	0.0000	52.1405	0.0000	-20.9595	0.0000

F R A M E E L E M E N T F O R C E S

FRAME	LOAD	LOC	P	V2	V3	T	M2	M3
1	LOAD1							
		0.00	-47.86	-5.81	0.00	0.00	0.00	-6.66
		2.50	-47.86	-5.81	0.00	0.00	0.00	7.86
		5.00	-47.86	-5.81	0.00	0.00	0.00	22.39
2	LOAD1							
		0.00	-10.81	-47.86	0.00	0.00	0.00	-22.39
		1.25	-10.81	-22.86	0.00	0.00	0.00	21.81
		2.50	-10.81	2.14	0.00	0.00	0.00	34.76
		3.75	-10.81	27.14	0.00	0.00	0.00	16.46
		5.00	-10.81	52.14	0.00	0.00	0.00	-33.09
3	LOAD1							
		0.00	-52.14	-10.81	0.00	0.00	0.00	-33.09
		2.50	-52.14	-10.81	0.00	0.00	0.00	-6.07
		5.00	-52.14	-10.81	0.00	0.00	0.00	20.96



SAP2000
20.00
20.00
5.00
1
2
3
4
X
Z

Input Data Struktural Kasus 2
Nama _file_: kasus2a.txt

```
5     6     5     2     2e7
1     0     0
2     0     5
3     1.5   5
4     8.5   5
5     10    5
6     10    0
1     1     2     0.5   0.5   0.5   0.6   0.6   0.6
2     2     3     0.5   0.3   0.4   0.8   0.5   0.65
3     3     4     0.3   0.3   0.3   0.5   0.5   0.5
4     4     5     0.3   0.5   0.4   0.5   0.8   0.65
5     5     6     0.5   0.5   0.5   0.6   0.6   0.6
1     1     1     1
6     1     1     0
```

Input Data Beban Kasus 2
Nama _file_: kasus2b.txt

```
1     3
2     20    0     0
2     50    0     50    1.5   0     0     0     0     0     0     0     0
3     50    0     50    7     0     0     0     0     0     0     0     0
4     50    0     50    1.5   0     0     0     0     0     0     0     0
```

***Output* Kasus 2**
Nama *file*: kasus2c.txt

n-Prime 2004 Project
[D:\Skripsi Stevie\New Folder\inputTA\kasus2c.txt]
Unit: kN - m

Frame Element Fixed-End Forces

Frame ID	R-i	M-i	R-j	M-j
2	42.28868	13.39282	32.71135	-6.209813
3	175	204.1668	175	-204.1668
4	32.71136	6.209821	42.2887	-13.39285

Joint Displacements

Joint ID	Ux	Uy	Rz
1	0	0	0
2	7.564174E-03	-1.986175E-04	-4.769005E-03
3	7.537199E-03	-8.447456E-03	-6.312075E-03
4	7.314805E-03	-6.274526E-03	5.409427E-03
5	7.287831E-03	-2.180497E-04	2.955006E-03
6	0	0	-3.663852E-03

Frame Element Forces

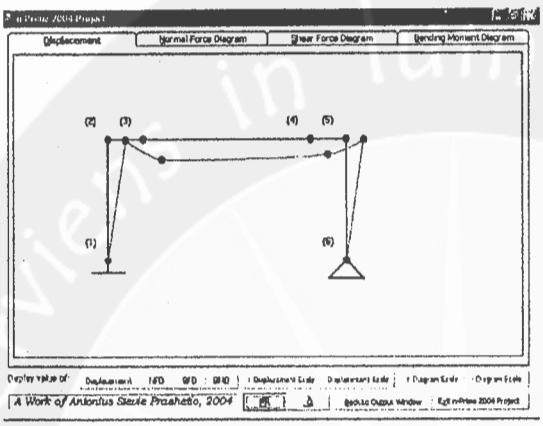
Frame ID	Axial-i	Shear-i	Moment-i	Axial-j	Shear-j	Moment-j
1	-238.341	-75.3121	16.59606	-238.341	-75.3121	-359.9644
2	-95.31105	238.341	-359.9642	-95.31105	163.341	-58.70307
3	-95.31142	163.3407	-58.70294	-95.31142	-186.6594	-140.3182
4	-95.31105	-186.6595	-140.3182	-95.31105	-261.6596	-476.5579
5	-261.6596	95.31159	-476.5579	-261.6596	95.31159	3.453952E-05

Support Reactions

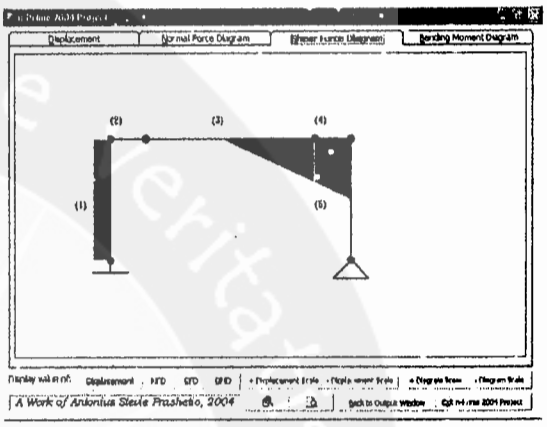
Joint ID	Fx	Fy	Mz
1	75.3121	238.341	-16.59606
6	-95.31159	261.6596	0

Tampilan Grafis *Joint Displacement*, NFD, SFD, dan BMD kasus 2

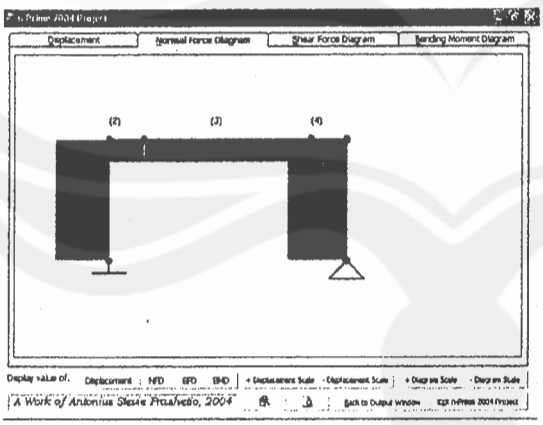
Joint Displacement



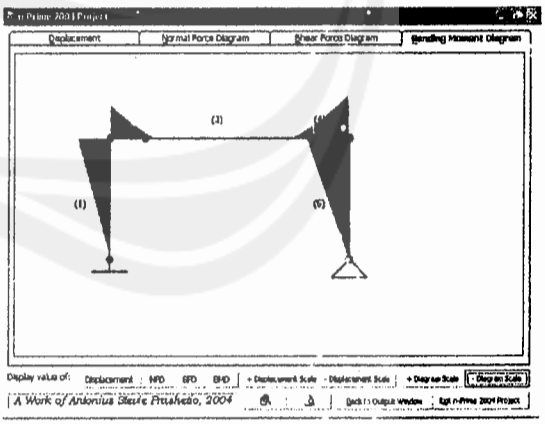
Shear Force Diagram



Normal Force Diagram



Bending Moment Diagram



Input Tables SAP2000 Kasus 2
Nama *file*: val2a.txt

SAP2000 v7.42 File: KASUS2 KN-m Units PAGE 1
5/20/04 17:41:47

Antonius Stevie Prashetio

S T A T I C L O A D C A S E S

STATIC CASE	CASE TYPE	SELF WT FACTOR
LOAD1	DEAD	0.0000

J O I N T D A T A

JOINT	GLOBAL-X	GLOBAL-Y	GLOBAL-Z	RESTRAINTS	ANGLE-A	ANGLE-B	ANGLE-C
1	0.00000	0.00000	0.00000	1 1 1 1 1 1	0.000	0.000	0.000
2	0.00000	0.00000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
3	1.50000	0.00000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
4	8.50000	0.00000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
5	10.00000	0.00000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
6	10.00000	0.00000	0.00000	1 1 1 0 0 0	0.000	0.000	0.000

F R A M E E L E M E N T D A T A

FRAME	JNT-1	JNT-2	SECTION	ANGLE	RELEASES	SEGMENTS	R1	R2	FACTOR	LENGT
1	1	2	500X600	0.000	000000	2	0.000	0.000	1.000	5.00
2	2	3	BS	0.000	000000	4	0.000	0.000	1.000	1.50
3	3	4	300X500	0.000	000000	4	0.000	0.000	1.000	7.00
4	4	5	SB	0.000	000000	4	0.000	0.000	1.000	1.50
5	5	6	500X600	0.000	000000	2	0.000	0.000	1.000	5.00

J O I N T F O R C E S Load Case LOAD1

JOINT	GLOBAL-X	GLOBAL-Y	GLOBAL-Z	GLOBAL-XX	GLOBAL-YY	GLOBAL-ZZ
2	20.000	0.000	0.000	0.000	0.000	0.000

F R A M E S P A N D I S T R I B U T E D L O A D S Load Case LOAD1

FRAME	TYPE	DIRECTION	DISTANCE-A	VALUE-A	DISTANCE-B	VALUE-B
2	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	-50.0000
3	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	-50.0000
4	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	-50.0000

Output Tables SAP2000 Kasus 2
Nama *file*: val2b.txt

SAP2000 v7.42 File: KASUS2 KN-m Units PAGE 1
5/20/04 17:42:04

Antonius Stevie Prashetio

J O I N T D I S P L A C E M E N T S

JOINT	LOAD	U1	U2	U3	R1	R2	R3
1	LOAD1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	LOAD1	9.372E-03	0.0000	-1.978E-04	0.0000	6.248E-03	0.0000
3	LOAD1	9.344E-03	0.0000	-0.0106	0.0000	7.316E-03	0.0000
4	LOAD1	9.127E-03	0.0000	-8.318E-03	0.0000	-6.287E-03	0.0000
5	LOAD1	9.100E-03	0.0000	-2.188E-04	0.0000	-4.418E-03	0.0000
6	LOAD1	0.0000	0.0000	0.0000	0.0000	4.872E-03	0.0000

J O I N T R E A C T I O N S

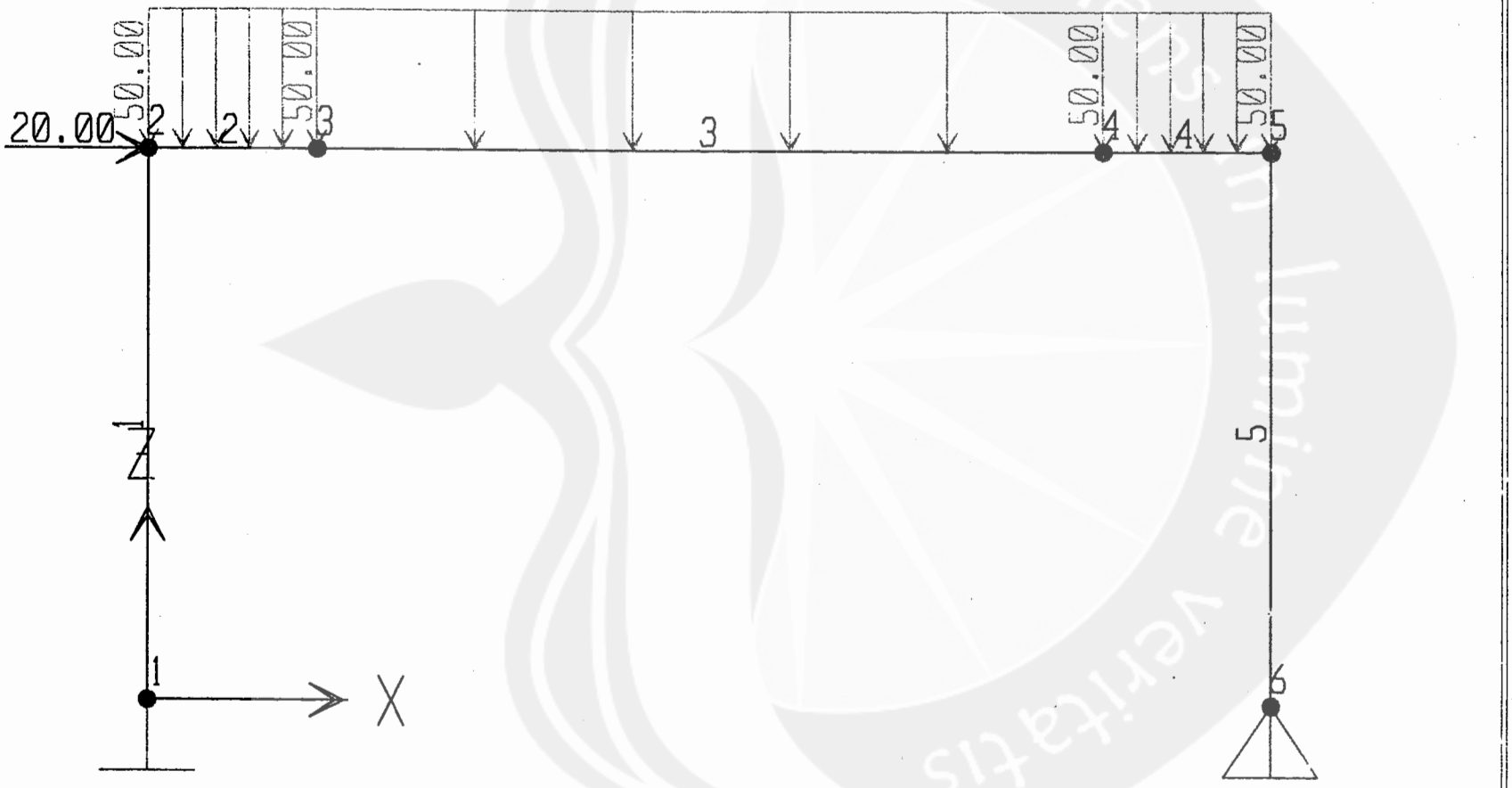
JOINT	LOAD	F1	F2	F3	M1	M2	M3
1	LOAD1	72.8953	0.0000	237.3973	0.0000	26.0271	0.0000
6	LOAD1	-92.8953	0.0000	262.6027	0.0000	0.0000	0.0000

F R A M E E L E M E N T F O R C E S

FRAME	LOAD	LOC	P	V2	V3	T	M2	M3
1	LOAD1							
		0.00	-237.40	-72.90	0.00	0.00	0.00	-26.03
		2.50	-237.40	-72.90	0.00	0.00	0.00	156.21
		5.00	-237.40	-72.90	0.00	0.00	0.00	338.45
2	LOAD1							
		0.00	-92.90	-237.40	0.00	0.00	0.00	-338.45
		3.8E-01	-92.90	-218.65	0.00	0.00	0.00	-252.94
		7.5E-01	-92.90	-199.90	0.00	0.00	0.00	-174.46
		1.13	-92.90	-181.15	0.00	0.00	0.00	-103.02
		1.50	-92.90	-162.40	0.00	0.00	0.00	-38.60
3	LOAD1							
		0.00	-92.90	-162.40	0.00	0.00	0.00	-38.60
		1.75	-92.90	-74.90	0.00	0.00	0.00	169.03
		3.50	-92.90	12.60	0.00	0.00	0.00	220.54
		5.25	-92.90	100.10	0.00	0.00	0.00	124.92
		7.00	-92.90	187.60	0.00	0.00	0.00	-126.82
4	LOAD1							
		0.00	-92.90	187.60	0.00	0.00	0.00	-126.82
		3.8E-01	-92.90	206.35	0.00	0.00	0.00	-200.69
		7.5E-01	-92.90	225.10	0.00	0.00	0.00	-281.59
		1.13	-92.90	243.85	0.00	0.00	0.00	-369.52
		1.50	-92.90	262.60	0.00	0.00	0.00	-464.48
5	LOAD1							
		0.00	-262.60	-92.90	0.00	0.00	0.00	-464.48
		2.50	-262.60	-92.90	0.00	0.00	0.00	-232.24
		5.00	-262.60	-92.90	0.00	0.00	0.00	0.00

SAP2000

5/20/04 17:47:15



SAP2000 v7.42 - File:kasus2 - Frame Span Loads (LOAD1) - KN-m Units

serviceforlumire veritatis

***Input* Data Struktural Kasus 3**
Nama *file*: kasus3a.txt

```
14    14    9     3     2e7
1     0     0
2     0     5
3     1.5   5
4     4.5   5
5     6     5
6     6     0
7     7.5   5
8     10.5  5
9     12    5
10    12    0
11    6     10
12    7.5   10
13    10.5  10
14    12    10
1     1     2     0.4   0.4   0.4   0.4   0.4   0.4
2     2     3     0.4   0.3   0.35  0.6   0.3   0.5
3     3     4     0.3   0.3   0.3   0.3   0.3   0.3
4     4     5     0.3   0.4   0.35  0.3   0.6   0.5
5     6     5     0.4   0.4   0.4   0.4   0.4   0.4
6     5     7     0.4   0.3   0.35  0.6   0.3   0.5
7     7     8     0.3   0.3   0.3   0.3   0.3   0.3
8     8     9     0.3   0.4   0.35  0.3   0.6   0.5
9     10    9     0.4   0.4   0.4   0.4   0.4   0.4
10    5     11    0.4   0.4   0.4   0.4   0.4   0.4
11    11    12    0.4   0.3   0.35  0.6   0.3   0.5
12    12    13    0.3   0.3   0.3   0.3   0.3   0.3
13    13    14    0.3   0.4   0.35  0.3   0.6   0.5
14    9     14    0.4   0.4   0.4   0.4   0.4   0.4
1     1     1     1
6     1     1     1
10    1     1     1
```

***Input* Data Beban Kasus 3**
Nama *file*: kasus3b.txt

```
0     9
2     25    0     25    1.5   0     0     0     0     0     0     0     0
3     25    0     25    3     0     0     0     0     0     0     0     0
4     25    0     25    1.5   0     0     0     0     0     0     0     0
6     25    0     25    1.5   0     0     0     0     0     0     0     0
7     25    0     25    3     0     0     0     0     0     0     0     0
8     25    0     25    1.5   0     0     0     0     0     0     0     0
11    25    0     25    1.5   0     0     0     0     0     0     0     0
12    25    0     25    3     0     0     0     0     0     0     0     0
13    25    0     25    1.5   0     0     0     0     0     0     0     0
```

***Output* Kasus 3**
Nama *file*: kasus3c.txt

n-Prime 2004 Project
[D:\Skripsi Stevie\New Folder\inputTA\kasus3c.txt]
Unit: kN - m

Frame Element Fixed-End Forces

Frame ID	R-i	M-i	R-j	M-j
2	21.57681	6.907298	15.92321	-2.667089
3	37.5	18.75	37.5	-18.75
4	15.92321	2.667089	21.57681	-6.907298
6	21.57681	6.907298	15.92321	-2.667089
7	37.5	18.75	37.5	-18.75
8	15.92321	2.667089	21.57681	-6.907298
11	21.57681	6.907298	15.92321	-2.667089
12	37.5	18.75	37.5	-18.75
13	15.92321	2.667089	21.57681	-6.907298

Joint Displacements

Joint ID	Ux	Uy	Rz
1	0	0	0
2	5.961754E-04	-1.032385E-04	-1.893764E-03
3	5.8877E-04	-3.225918E-03	-2.120633E-03
4	5.605195E-04	-2.164182E-03	1.836532E-03
5	5.53114E-04	-3.810073E-04	6.07269E-04
6	0	0	0
7	5.595692E-04	-4.616509E-04	-8.52479E-04
8	5.841944E-04	-1.530764E-03	1.089988E-03
9	5.906495E-04	-2.188793E-04	5.90518E-04
10	0	0	0
11	-1.193375E-03	-5.003276E-04	-2.178336E-03
12	-1.203529E-03	-4.260616E-03	-2.77387E-03
13	-1.242265E-03	-4.381008E-03	2.858619E-03
14	-1.252418E-03	-3.339339E-04	2.416362E-03

Frame Element Forces

Frame ID	Axial-i	Shear-i	Moment-i	Axial-j	Shear-j	Moment-j
1	-66.07266	-16.95021	26.21541	-66.07266	-16.95021	-58.53565
2	-16.9503	66.07268	-58.53568	-16.9503	28.57265	12.44821
3	-16.95027	28.57267	12.44823	-16.95027	-46.42733	-14.33375
4	-16.9503	-46.42732	-14.33379	-16.9503	-83.92734	-112.0998
5	-243.8447	8.48399	-16.02795	-243.8447	8.48399	26.392
6	14.77517	83.55231	-120.0407	14.77517	46.05228	-22.83728
7	14.77513	46.05226	-22.83729	14.77513	-28.94774	2.819496
8	14.77504	-28.94771	2.819468	14.77504	-66.44773	-68.72715
9	-140.0827	8.466205	-16.12643	-140.0827	8.466205	26.2046
10	-76.36501	-23.24134	34.33287	-76.36501	-23.24134	-81.87386
11	-23.24106	76.36511	-81.87401	-23.24106	38.86509	4.548508
12	-23.24113	38.8651	4.548553	-23.24113	-36.1349	8.64385
13	-23.24079	-36.13485	8.643798	-23.24079	-73.63487	-73.6836
14	-73.63496	23.24124	-42.52258	-73.63496	23.24124	73.68365

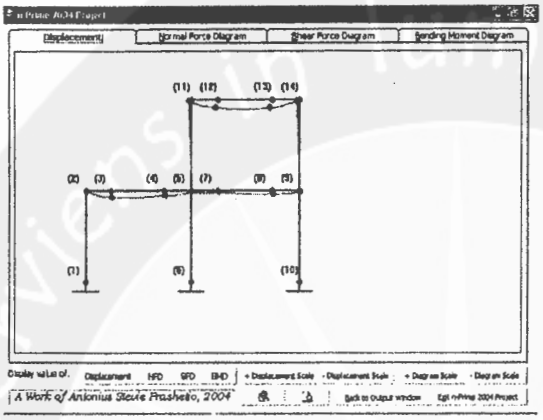
Support Reactions

Joint ID	Fx	Fy	Mz
1	16.95021	66.07266	-26.21541
6	-8.48399	243.8447	16.02795
10	-8.466205	140.0827	16.12643

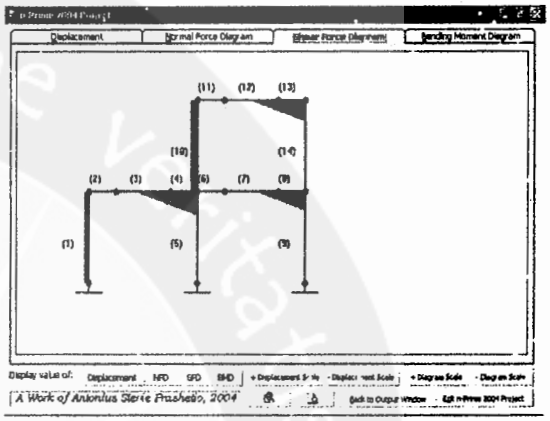
serviet Illumine Veritatis

Tampilan Grafis *Joint Displacement*, NFD, SFD, dan BMD kasus 3

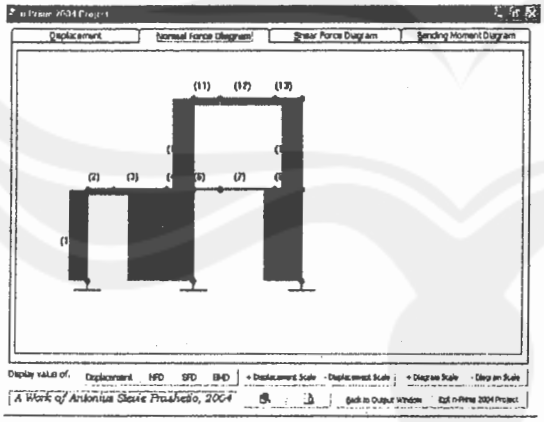
Joint Displacement



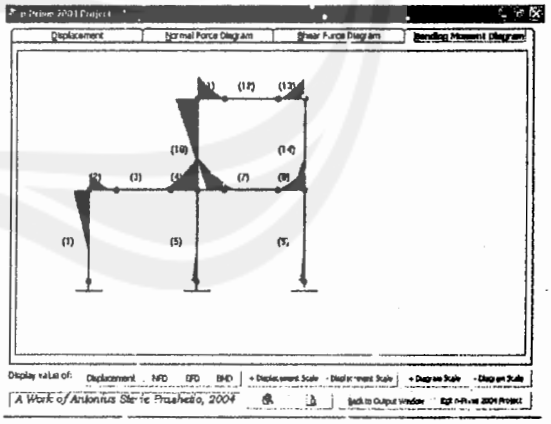
Shear Force Diagram



Normal Force Diagram



Bending Moment Diagram



Input Tables SAP2000 Kasus 3
Nama *file*: val3a.txt

SAP2000 v7.42 File: KASUS3 KN-m Units PAGE 1
5/20/04 18:00:45

Antonius Stevie Prashetio

S T A T I C L O A D C A S E S

STATIC CASE	CASE TYPE	SELF WT FACTOR
LOAD1	DEAD	0.0000

J O I N T D A T A

JOINT	GLOBAL-X	GLOBAL-Y	GLOBAL-Z	RESTRAINTS	ANGLE-A	ANGLE-B	ANGLE-C
1	0.00000	2.50000	0.00000	1 1 1 1 1 1	0.000	0.000	0.000
2	0.00000	2.50000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
3	1.50000	2.50000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
4	4.50000	2.50000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
5	6.00000	2.50000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
6	6.00000	2.50000	0.00000	1 1 1 1 1 1	0.000	0.000	0.000
7	7.50000	2.50000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
8	10.50000	2.50000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
9	12.00000	2.50000	5.00000	0 0 0 0 0 0	0.000	0.000	0.000
10	12.00000	2.50000	0.00000	1 1 1 1 1 1	0.000	0.000	0.000
11	6.00000	2.50000	10.00000	0 0 0 0 0 0	0.000	0.000	0.000
12	7.50000	2.50000	10.00000	0 0 0 0 0 0	0.000	0.000	0.000
13	10.50000	2.50000	10.00000	0 0 0 0 0 0	0.000	0.000	0.000
14	12.00000	2.50000	10.00000	0 0 0 0 0 0	0.000	0.000	0.000

F R A M E E L E M E N T D A T A

FRAME	JNT-1	JNT-2	SECTION	ANGLE	RELEASES	SEGMENTS	R1	R2	FACTOR	LENGT
1	1	2	400X400	0.000	000000	2	0.000	0.000	1.000	5.0(
2	2	3	VAR1	0.000	000000	4	0.000	0.000	1.000	1.5(
3	3	4	300X300	0.000	000000	4	0.000	0.000	1.000	3.0(
4	4	5	VAR2	0.000	000000	4	0.000	0.000	1.000	1.5(
5	5	6	400X400	0.000	000000	2	0.000	0.000	1.000	5.0(
6	5	7	VAR1	0.000	000000	4	0.000	0.000	1.000	1.5(
7	7	8	300X300	0.000	000000	4	0.000	0.000	1.000	3.0(
8	8	9	VAR2	0.000	000000	4	0.000	0.000	1.000	1.5(
9	9	10	400X400	0.000	000000	2	0.000	0.000	1.000	5.00
10	5	11	400X400	0.000	000000	2	0.000	0.000	1.000	5.00
11	11	12	VAR1	0.000	000000	4	0.000	0.000	1.000	1.50
12	12	13	300X300	0.000	000000	4	0.000	0.000	1.000	3.00
13	13	14	VAR2	0.000	000000	4	0.000	0.000	1.000	1.50
14	14	9	400X400	0.000	000000	2	0.000	0.000	1.000	5.00

F R A M E S P A N D I S T R I B U T E D L O A D S Load Case LOAD1

FRAME	TYPE	DIRECTION	DISTANCE-A	VALUE-A	DISTANCE-B	VALUE-B
2	FORCE	GLOBAL-Z	0.0000	-25.0000	1.0000	-25.0000
3	FORCE	GLOBAL-Z	0.0000	-25.0000	1.0000	-25.0000
4	FORCE	GLOBAL-Z	0.0000	-25.0000	1.0000	-25.0000
6	FORCE	GLOBAL-Z	0.0000	-25.0000	1.0000	-25.0000
7	FORCE	GLOBAL-Z	0.0000	-25.0000	1.0000	-25.0000
8	FORCE	GLOBAL-Z	0.0000	-25.0000	1.0000	-25.0000
11	FORCE	GLOBAL-Z	0.0000	-25.0000	1.0000	-25.0000
12	FORCE	GLOBAL-Z	0.0000	-25.0000	1.0000	-25.0000
13	FORCE	GLOBAL-Z	0.0000	-25.0000	1.0000	-25.0000

Output Tables SAP2000 Kasus 3
Nama *file*: val3b.txt

SAP2000 v7.42 File: KASUS3 KN-m Units PAGE 1
5/20/04 18:00:57

Antonius Stevie Prashetio

J O I N T D I S P L A C E M E N T S

JOINT	LOAD	U1	U2	U3	R1	R2	R3
1	LOAD1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	LOAD1	6.053E-04	0.0000	-1.036E-04	0.0000	1.937E-03	0.0000
3	LOAD1	5.972E-04	0.0000	-3.379E-03	0.0000	2.117E-03	0.0000
4	LOAD1	5.688E-04	0.0000	-2.273E-03	0.0000	-1.860E-03	0.0000
5	LOAD1	5.607E-04	0.0000	-3.803E-04	0.0000	-6.125E-04	0.0000
6	LOAD1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7	LOAD1	5.676E-04	0.0000	-5.513E-04	0.0000	8.898E-04	0.0000
8	LOAD1	5.919E-04	0.0000	-1.659E-03	0.0000	-1.102E-03	0.0000
9	LOAD1	5.989E-04	0.0000	-2.192E-04	0.0000	-6.182E-04	0.0000
10	LOAD1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
11	LOAD1	-1.243E-03	0.0000	-4.997E-04	0.0000	2.199E-03	0.0000
12	LOAD1	-1.254E-03	0.0000	-4.413E-03	0.0000	2.768E-03	0.0000
13	LOAD1	-1.293E-03	0.0000	-4.529E-03	0.0000	-2.850E-03	0.0000
14	LOAD1	-1.304E-03	0.0000	-3.342E-04	0.0000	-2.451E-03	0.0000

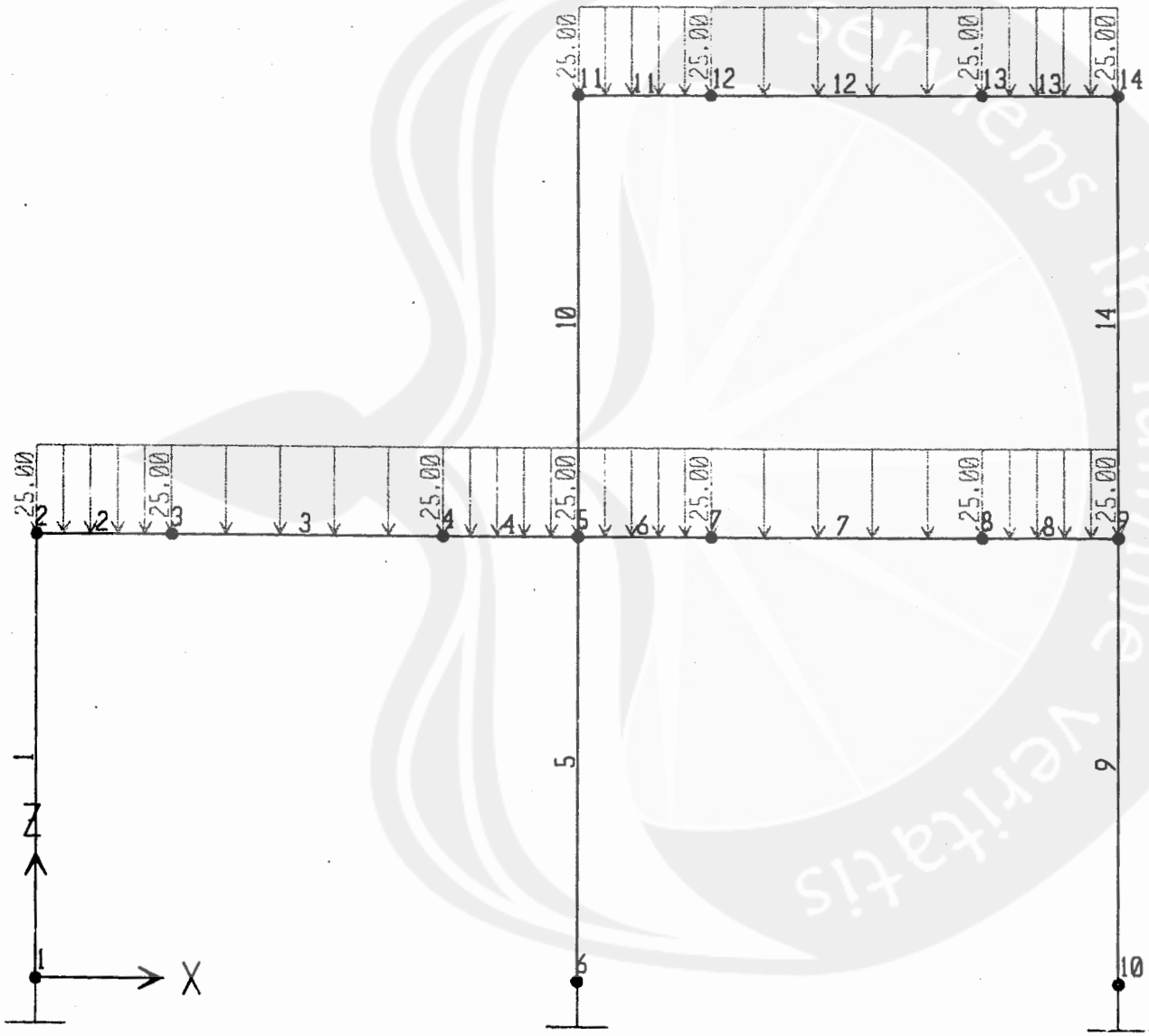
J O I N T R E A C T I O N S

JOINT	LOAD	F1	F2	F3	M1	M2	M3
1	LOAD1	17.0381	0.0000	66.2977	0.0000	26.0691	0.0000
6	LOAD1	-8.4136	0.0000	243.4085	0.0000	-15.8073	0.0000
10	LOAD1	-8.6245	0.0000	140.2937	0.0000	-16.2858	0.0000

F R A M E E L E M E N T F O R C E S

FRAME	LOAD	LOC	P	V2	V3	T	M2	M3
1	LOAD1							
		0.00	-66.30	-17.04	0.00	0.00	0.00	-26.07
		2.50	-66.30	-17.04	0.00	0.00	0.00	16.53
		5.00	-66.30	-17.04	0.00	0.00	0.00	59.12
2	LOAD1							
		0.00	-17.04	-66.30	0.00	0.00	0.00	-59.12
		3.8E-01	-17.04	-56.92	0.00	0.00	0.00	-36.02
		7.5E-01	-17.04	-47.55	0.00	0.00	0.00	-16.43
		1.13	-17.04	-38.17	0.00	0.00	0.00	-3.567E-01
		1.50	-17.04	-28.80	0.00	0.00	0.00	12.20
3	LOAD1							
		0.00	-17.04	-28.80	0.00	0.00	0.00	12.20
		7.5E-01	-17.04	-10.05	0.00	0.00	0.00	26.77

	1.50	-17.04	8.70	0.00	0.00	0.00	27.27
	2.25	-17.04	27.45	0.00	0.00	0.00	13.71
	3.00	-17.04	46.20	0.00	0.00	0.00	-13.91
4	LOAD1						
	0.00	-17.04	46.20	0.00	0.00	0.00	-13.91
	3.8E-01	-17.04	55.58	0.00	0.00	0.00	-32.99
	7.5E-01	-17.04	64.95	0.00	0.00	0.00	-55.59
	1.13	-17.04	74.33	0.00	0.00	0.00	-81.70
	1.50	-17.04	83.70	0.00	0.00	0.00	-111.33
5	LOAD1						
	0.00	-243.41	-8.41	0.00	0.00	0.00	-26.26
	2.50	-243.41	-8.41	0.00	0.00	0.00	-5.23
	5.00	-243.41	-8.41	0.00	0.00	0.00	15.81
6	LOAD1						
	0.00	14.59	-83.31	0.00	0.00	0.00	-119.11
	3.8E-01	14.59	-73.94	0.00	0.00	0.00	-69.62
	7.5E-01	14.59	-64.56	0.00	0.00	0.00	-63.65
	1.13	14.59	-55.19	0.00	0.00	0.00	-41.20
	1.50	14.59	-45.81	0.00	0.00	0.00	-22.26
7	LOAD1						
	0.00	14.59	-45.81	0.00	0.00	0.00	-22.26
	7.5E-01	14.59	-27.06	0.00	0.00	0.00	5.07
	1.50	14.59	-8.31	0.00	0.00	0.00	18.34
	2.25	14.59	10.44	0.00	0.00	0.00	17.54
	3.00	14.59	29.19	0.00	0.00	0.00	2.68
8	LOAD1						
	0.00	14.59	29.19	0.00	0.00	0.00	2.68
	3.8E-01	14.59	38.56	0.00	0.00	0.00	-10.02
	7.5E-01	14.59	47.94	0.00	0.00	0.00	-26.24
	1.13	14.59	57.31	0.00	0.00	0.00	-45.97
	1.50	14.59	66.69	0.00	0.00	0.00	-69.22
9	LOAD1						
	0.00	-140.29	-8.62	0.00	0.00	0.00	-26.84
	2.50	-140.29	-8.62	0.00	0.00	0.00	-5.28
	5.00	-140.29	-8.62	0.00	0.00	0.00	16.29
10	LOAD1						
	0.00	-76.39	-23.21	0.00	0.00	0.00	-34.03
	2.50	-76.39	-23.21	0.00	0.00	0.00	23.99
	5.00	-76.39	-23.21	0.00	0.00	0.00	82.02
11	LOAD1						
	0.00	-23.21	-76.39	0.00	0.00	0.00	-82.02
	3.8E-01	-23.21	-67.02	0.00	0.00	0.00	-55.13
	7.5E-01	-23.21	-57.64	0.00	0.00	0.00	-31.76
	1.13	-23.21	-48.27	0.00	0.00	0.00	-11.90
	1.50	-23.21	-38.89	0.00	0.00	0.00	4.44
12	LOAD1						
	0.00	-23.21	-38.89	0.00	0.00	0.00	4.44
	7.5E-01	-23.21	-20.14	0.00	0.00	0.00	26.58
	1.50	-23.21	-1.39	0.00	0.00	0.00	34.65
	2.25	-23.21	17.36	0.00	0.00	0.00	28.67
	3.00	-23.21	36.11	0.00	0.00	0.00	8.62
13	LOAD1						
	0.00	-23.21	36.11	0.00	0.00	0.00	8.62
	3.8E-01	-23.21	45.48	0.00	0.00	0.00	-6.68
	7.5E-01	-23.21	54.86	0.00	0.00	0.00	-25.50
	1.13	-23.21	64.23	0.00	0.00	0.00	-47.83
	1.50	-23.21	73.61	0.00	0.00	0.00	-73.67
14	LOAD1						
	0.00	-73.61	-23.21	0.00	0.00	0.00	-73.67
	2.50	-73.61	-23.21	0.00	0.00	0.00	-15.64
	5.00	-73.61	-23.21	0.00	0.00	0.00	42.38


25.00
10
14
5
9
Z
X
1
6
10

Input Data Struktural Kasus 4
Nama _file_: kasus4a.txt

```
10    10    6     2     24821128
1     0     0
2     11    0
3     0     4
4     3     4
5     8     4
6     11    4
7     0     8
8     3     8
9     8     8
10    11    8
1     1     3     0.4   0.4   0.4   0.4   0.4   0.4
2     2     6     0.4   0.4   0.4   0.4   0.4   0.4
3     3     4     0.4   0.4   0.4   0.6   0.6   0.6
4     4     5     0.4   0.4   0.4   0.6   0.6   0.6
5     5     6     0.4   0.4   0.4   0.6   0.6   0.6
6     3     7     0.4   0.4   0.4   0.4   0.4   0.4
7     6     10    0.4   0.4   0.4   0.4   0.4   0.4
8     7     8     0.4   0.4   0.4   0.6   0.6   0.6
9     8     9     0.4   0.4   0.4   0.6   0.6   0.6
10    9     10    0.4   0.4   0.4   0.6   0.6   0.6
1     1     1     1
2     1     1     1
```

Input Data Beban Kasus 4
Nama _file_: kasus4b.txt

```
2     6
3     100   0     0
7     200   0     0
3     0     0     50    3     0     0     0     0     0     0     0     0
8     0     0     50    3     0     0     0     0     0     0     0     0
5     50    0     0     3     0     0     0     0     0     0     0     0
10    50    0     0     3     0     0     0     0     0     0     0     0
4     50    0     50    5     100   2.5   0     0     0     0     0     0
9     50    0     50    5     100   2.5   0     0     0     0     0     0
```

***Output* Kasus 4**
Nama *file*: kasus4c.txt

n-Prime 2004 Project
[D:\Skripsi Stevie\New Folder\inputTA\kasus4c.txt]
Unit: kN - m

Frame Element Fixed-End Forces

Frame ID	R-i	M-i	R-j	M-j
3	22.49999	14.99999	52.49999	-22.49998
8	22.49999	14.99999	52.49999	-22.49998
5	52.49999	22.49998	22.49999	-14.99999
10	52.49999	22.49998	22.49999	-14.99999
4	174.9999	166.6664	174.9999	-166.6664
9	174.9999	166.6664	174.9999	-166.6664

Joint Displacements

Joint ID	Ux	Uy	Rz
1	0	0	0
2	0	0	0
3	2.365318E-02	-3.858345E-04	-7.383304E-03
4	2.368484E-02	-1.983906E-02	-4.312692E-03
5	2.373761E-02	-0.0114571	5.896929E-03
6	2.376927E-02	-6.213708E-04	-1.219281E-03
7	4.733245E-02	-5.962687E-04	-8.215324E-03
8	4.719415E-02	-2.420657E-02	-5.960109E-03
9	4.696362E-02	-1.975491E-02	6.783204E-03
10	0.0468253	-9.145391E-04	3.522826E-03

Frame Element Forces

Frame ID	Axial-i	Shear-i	Moment-i	Axial-j	Shear-j	Moment-j
1	-383.0739	88.22992	-274.1996	-383.0739	88.22992	78.72016
2	-616.925	211.781	-439.7027	-616.925	211.781	407.4212

3	62.87312	174.1453	-59.54956	62.87312	99.14536	387.8867
4	62.86943	99.14536	387.8866	62.86943	-250.8544	8.613968
5	62.86942	-250.8544	8.61396	62.86942	-325.8544	-893.9494
6	-208.9286	-74.64193	138.2696	-208.9286	-74.64193	-160.298
7	-291.0707	274.6519	-486.5281	-291.0707	274.6519	612.0795
8	-274.6385	208.9287	-160.2984	-274.6385	133.9288	391.488
9	-274.6504	133.9289	391.4879	-274.6504	-216.0709	186.1331
10	-274.6607	-216.0708	186.1333	-274.6607	-291.0708	-612.0794

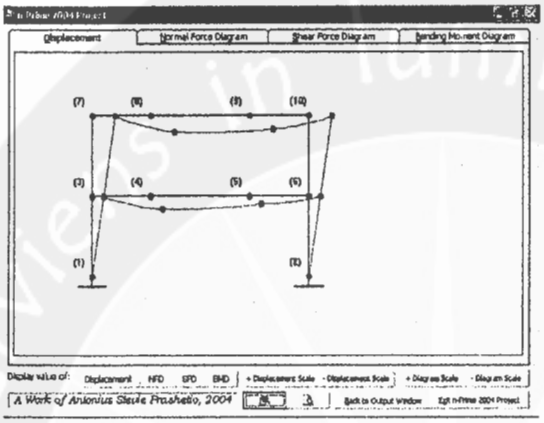
Support Reactions

Joint ID	Fx	Fy	Mz
1	-88.22992	383.0739	274.1996
2	-211.781	616.925	439.7027

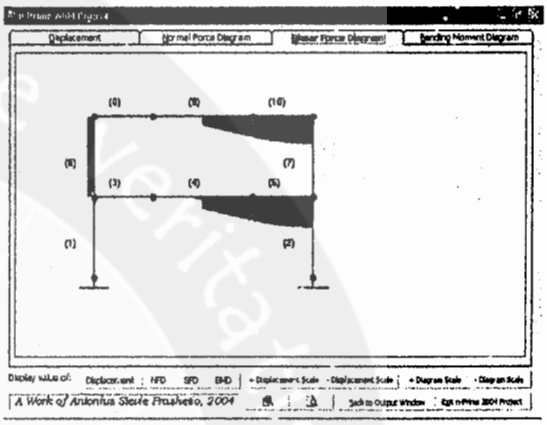
serviam lumine veritatis

Tampilan Grafis *Joint Displacement*, NFD, SFD, dan BMD kasus 4

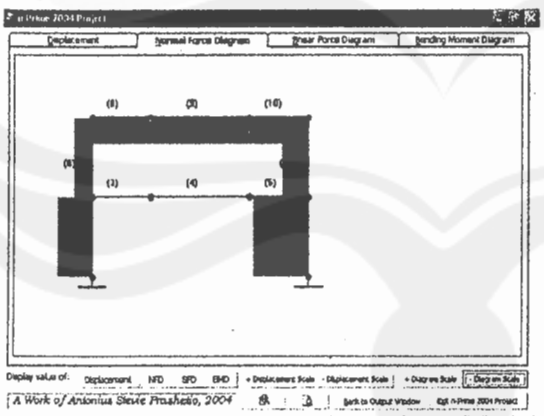
Joint Displacement



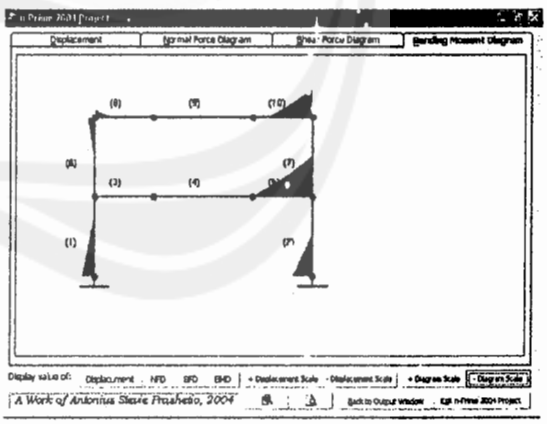
Shear Force Diagram



Normal Force Diagram



Bending Moment Diagram



Input Tables SAP2000 Kasus 4
Nama *file*: val4a.txt

SAP2000 v7.42 File: KASUS4 KN-m Units PAGE 1
5/20/04 18:07:23

Antonius Stevie Prashetio

S T A T I C L O A D C A S E S

STATIC CASE	CASE TYPE	SELF WT FACTOR
LOAD1	DEAD	0.0000

J O I N T D A T A

JOINT	GLOBAL-X	GLOBAL-Y	GLOBAL-Z	RESTRAINTS	ANGLE-A	ANGLE-B	ANGLE-C
1	0.00000	0.00000	0.00000	1 1 1 1 1 1	0.000	0.000	0.000
2	11.00000	0.00000	0.00000	1 1 1 1 1 1	0.000	0.000	0.000
3	0.00000	0.00000	4.00000	0 0 0 0 0 0	0.000	0.000	0.000
4	3.00000	0.00000	4.00000	0 0 0 0 0 0	0.000	0.000	0.000
5	8.00000	0.00000	4.00000	0 0 0 0 0 0	0.000	0.000	0.000
6	11.00000	0.00000	4.00000	0 0 0 0 0 0	0.000	0.000	0.000
7	0.00000	0.00000	8.00000	0 0 0 0 0 0	0.000	0.000	0.000
8	3.00000	0.00000	8.00000	0 0 0 0 0 0	0.000	0.000	0.000
9	8.00000	0.00000	8.00000	0 0 0 0 0 0	0.000	0.000	0.000
10	11.00000	0.00000	8.00000	0 0 0 0 0 0	0.000	0.000	0.000

F R A M E E L E M E N T D A T A

FRAME	JNT-1	JNT-2	SECTION	ANGLE	RELEASES	SEGMENTS	R1	R2	FACTOR	LENG
1	1	3	400X400	0.000	000000	2	0.000	0.000	1.000	4.0
2	2	6	400X400	0.000	000000	2	0.000	0.000	1.000	4.0
3	3	4	400X600	0.000	000000	4	0.000	0.000	1.000	3.0
4	4	5	400X600	0.000	000000	4	0.000	0.000	1.000	5.0
5	5	6	400X600	0.000	000000	4	0.000	0.000	1.000	3.0
6	3	7	400X400	0.000	000000	2	0.000	0.000	1.000	4.0
7	6	10	400X400	0.000	000000	2	0.000	0.000	1.000	4.0
8	7	8	400X600	0.000	000000	4	0.000	0.000	1.000	3.0
9	8	9	400X600	0.000	000000	4	0.000	0.000	1.000	5.0
10	9	10	400X600	0.000	000000	4	0.000	0.000	1.000	3.0

J O I N T F O R C E S Load Case LOAD1

JOINT	GLOBAL-X	GLOBAL-Y	GLOBAL-Z	GLOBAL-XX	GLOBAL-YY	GLOBAL-ZZ
7	200.000	0.000	0.000	0.000	0.000	0.000
3	100.000	0.000	0.000	0.000	0.000	0.000

F R A M E S P A N D I S T R I B U T E D L O A D S Load Case LOAD1

FRAME	TYPE	DIRECTION	DISTANCE-A	VALUE-A	DISTANCE-B	VALUE-B
8	FORCE	GLOBAL-Z	0.0000	0.0000	1.0000	-50.0000
3	FORCE	GLOBAL-Z	0.0000	0.0000	1.0000	-50.0000
5	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	0.0000
10	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	0.0000
4	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	-50.0000
9	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	-50.0000

F R A M E S P A N P O I N T L O A D S Load Case LOAD1

FRAME	TYPE	DIRECTION	DISTANCE	VALUE
4	FORCE	GLOBAL-Z	0.5000	-100.0000
9	FORCE	GLOBAL-Z	0.5000	-100.0000

Output Tables SAP2000 Kasus 4
Nama *file*: val4b.txt

SAP2000 v7.42 File: KASUS4 KN-m Units PAGE 1
5/20/04 18:07:35

Antonius Stevie Prashetio

J O I N T D I S P L A C E M E N T S

JOINT	LOAD	U1	U2	U3	R1	R2	R3
1	LOAD1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	LOAD1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
3	LOAD1	0.0242	0.0000	-3.859E-04	0.0000	7.494E-03	0.0000
4	LOAD1	0.0242	0.0000	-0.0203	0.0000	4.387E-03	0.0000
5	LOAD1	0.0242	0.0000	-0.0120	0.0000	-5.896E-03	0.0000
6	LOAD1	0.0243	0.0000	-6.213E-04	0.0000	1.170E-03	0.0000
7	LOAD1	0.0482	0.0000	-5.963E-04	0.0000	8.328E-03	0.0000
8	LOAD1	0.0481	0.0000	-0.0247	0.0000	6.021E-03	0.0000
9	LOAD1	0.0479	0.0000	-0.0203	0.0000	-6.805E-03	0.0000
10	LOAD1	0.0477	0.0000	-9.145E-04	0.0000	-3.593E-03	0.0000

J O I N T R E A C T I O N S

JOINT	LOAD	F1	F2	F3	M1	M2	M3
1	LOAD1	-88.4218	0.0000	383.1544	0.0000	-276.0510	0.0000
2	LOAD1	-211.5782	0.0000	616.8456	0.0000	-438.6468	0.0000

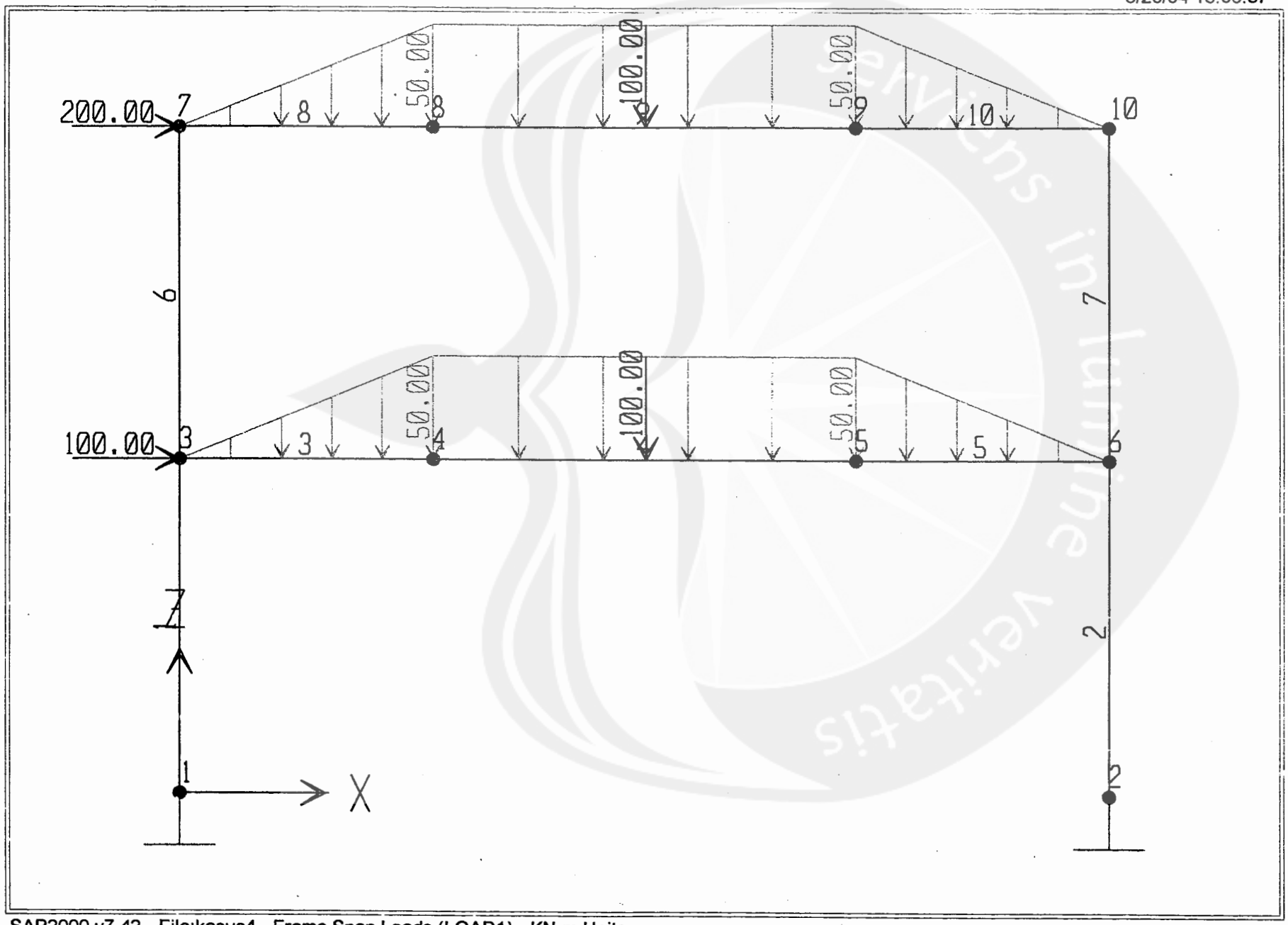
F R A M E E L E M E N T F O R C E S

FRAME	LOAD	LOC	P	V2	V3	T	M2	M3
1	LOAD1							
		0.00	-383.15	88.42	0.00	0.00	0.00	276.05
		2.00	-383.15	88.42	0.00	0.00	0.00	99.21
		4.00	-383.15	88.42	0.00	0.00	0.00	-77.64
2	LOAD1							
		0.00	-616.85	211.58	0.00	0.00	0.00	438.65
		2.00	-616.85	211.58	0.00	0.00	0.00	15.49
		4.00	-616.85	211.58	0.00	0.00	0.00	-407.67
3	LOAD1							
		0.00	61.51	-174.25	0.00	0.00	0.00	-57.51
		7.5E-01	61.51	-169.56	0.00	0.00	0.00	72.00
		1.50	61.51	-155.50	0.00	0.00	0.00	194.49
		2.25	61.51	-132.06	0.00	0.00	0.00	302.91
		3.00	61.51	-99.25	0.00	0.00	0.00	390.23
4	LOAD1							
		0.00	61.51	-99.25	0.00	0.00	0.00	390.23
		1.25	61.51	-36.75	0.00	0.00	0.00	475.23
		2.50	61.51	25.75	0.00	0.00	0.00	482.10
		3.75	61.51	188.25	0.00	0.00	0.00	285.85
		5.00	61.51	250.75	0.00	0.00	0.00	11.47

5	LOAD1						
	0.00	61.51	250.75	0.00	0.00	0.00	11.47
	7.5E-01	61.51	283.57	0.00	0.00	0.00	-189.49
	1.50	61.51	307.00	0.00	0.00	0.00	-411.54
	2.25	61.51	321.07	0.00	0.00	0.00	-647.65
	3.00	61.51	325.75	0.00	0.00	0.00	-890.79
6	LOAD1						
	0.00	-208.91	-73.09	0.00	0.00	0.00	-135.15
	2.00	-208.91	-73.09	0.00	0.00	0.00	11.03
	4.00	-208.91	-73.09	0.00	0.00	0.00	157.21
7	LOAD1						
	0.00	-291.09	273.09	0.00	0.00	0.00	485.12
	2.00	-291.09	273.09	0.00	0.00	0.00	-63.05
	4.00	-291.09	273.09	0.00	0.00	0.00	-609.23
8	LOAD1						
	0.00	-273.09	-208.91	0.00	0.00	0.00	-157.21
	7.5E-01	-273.09	-204.22	0.00	0.00	0.00	-1.70
	1.50	-273.09	-190.16	0.00	0.00	0.00	146.77
	2.25	-273.09	-166.72	0.00	0.00	0.00	281.19
	3.00	-273.09	-133.91	0.00	0.00	0.00	394.51
9	LOAD1						
	0.00	-273.09	-133.91	0.00	0.00	0.00	394.51
	1.25	-273.09	-71.41	0.00	0.00	0.00	522.83
	2.50	-273.09	-8.91	0.00	0.00	0.00	573.03
	3.75	-273.09	153.59	0.00	0.00	0.00	420.10
	5.00	-273.09	216.09	0.00	0.00	0.00	189.04
10	LOAD1						
	0.00	-273.09	216.09	0.00	0.00	0.00	189.04
	7.5E-01	-273.09	248.91	0.00	0.00	0.00	14.08
	1.50	-273.09	272.34	0.00	0.00	0.00	-181.97
	2.25	-273.09	286.41	0.00	0.00	0.00	-392.09
	3.00	-273.09	291.09	0.00	0.00	0.00	-609.23

SAP2000

5/20/04 18:06:37



SAP2000 v7.42 - File:kasus4 - Frame Span Loads (LOAD1) - KN-m Units

LAMPIRAN 4
Halaman .

Input Data Struktural Kasus 5
Nama _file_: kasus5a.txt

```
10    10    6     2     24821128
1     0     0
2     11    0
3     0     4
4     3     4
5     8     4
6     11    4
7     0     8
8     3     8
9     8     8
10    11    8
1     1     3     0.4   0.4   0.4   0.4   0.4   0.4
2     2     6     0.4   0.4   0.4   0.4   0.4   0.4
3     3     4     0.4   0.35  0.3   0.6   0.5   0.4
4     4     5     0.3   0.3   0.3   0.4   0.4   0.4
5     5     6     0.3   0.35  0.4   0.4   0.5   0.6
6     3     7     0.4   0.4   0.4   0.4   0.4   0.4
7     6     10    0.4   0.4   0.4   0.4   0.4   0.4
8     7     8     0.4   0.35  0.3   0.6   0.5   0.4
9     8     9     0.3   0.3   0.3   0.4   0.4   0.4
10    9     10    0.3   0.35  0.4   0.4   0.5   0.6
1     1     1     1
2     1     1     1
```

Input Data Beban Kasus 5
Nama _file_: kasus5b.txt

```
2     6
3     100   0     0
7     200   0     0
3     0     0     3     0     0     0     0     0     0     0     0     0
8     0     0     50    3     0     0     0     0     0     0     0     0
5     50    0     0     3     0     0     0     0     0     0     0     0
10    50    0     0     3     0     0     0     0     0     0     0     0
4     50    0     50    5     100   2.5   0     0     0     0     0     0
9     50    0     50    5     100   2.5   0     0     0     0     0     0
```

***Output* Kasus 5**
Nama *file*: kasus5c.txt

n-Prime 2004 Project
[D:\Skripsi Stevie\New Folder\inputTA\kasus5c.txt]
Unit: kN - m

Frame Element Fixed-End Forces

Frame ID	R-i	M-i	R-j	M-j
3	23.89688	19.87886	51.10318	-23.18833
8	23.89688	19.87886	51.10318	-23.18833
5	50.25166	16.38726	24.74835	-15.63229
10	50.25166	16.38726	24.74835	-15.63229
4	174.9999	166.6665	174.9999	-166.6665
9	174.9999	166.6665	174.9999	-166.6665

Joint Displacements

Joint ID	Ux	Uy	Rz
1	0	0	0
2	0	0	0
3	2.718979E-02	-3.903079E-04	-1.019806E-02
4	2.729813E-02	-3.543406E-02	-1.059233E-02
5	2.752033E-02	-1.743334E-02	1.048142E-02
6	2.759737E-02	-6.16898E-04	-2.087869E-03
7	5.725112E-02	-5.990017E-04	-1.313772E-02
8	5.695244E-02	-4.721146E-02	-1.510493E-02
9	5.633987E-02	-3.674794E-02	1.479996E-02
10	5.612749E-02	-9.118071E-04	6.199839E-03

Frame Element Forces

Frame ID	Axial-i	Shear-i	Moment-i	Axial-j	Shear-j	Moment-j
1	-387.5153	67.45059	-269.9025	-387.5153	67.45059	-0.1000462
2	-612.4842	232.5405	-492.7201	-612.4842	232.5405	437.442

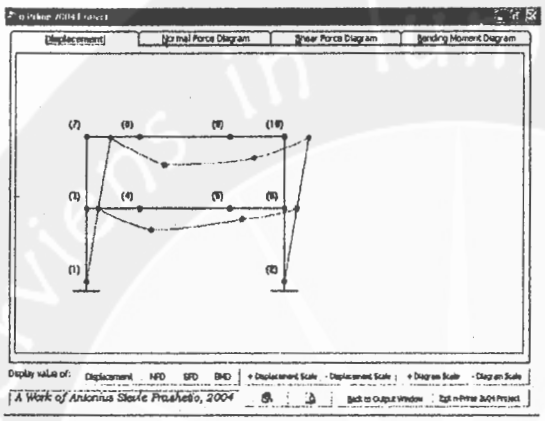
3	132.3677	180.3146	-291.0129	132.3677	105.3146	174.9307
4	132.3642	105.3146	174.9306	132.3642	-244.6852	-173.496
5	132.3696	-244.6852	-173.4962	132.3696	-319.6852	-1057.552
6	-207.2007	-164.9138	290.9125	-207.2007	-164.9138	-368.7426
7	-292.799	364.9109	-620.1097	-292.799	364.9109	839.5341
8	-364.9067	207.2006	-368.7424	-364.9067	132.2006	177.8592
9	-364.9112	132.2007	177.859	-364.9112	-217.7991	-36.13716
10	-364.9164	-217.799	-36.13758	-364.9164	-292.799	-839.5342

Support Reactions

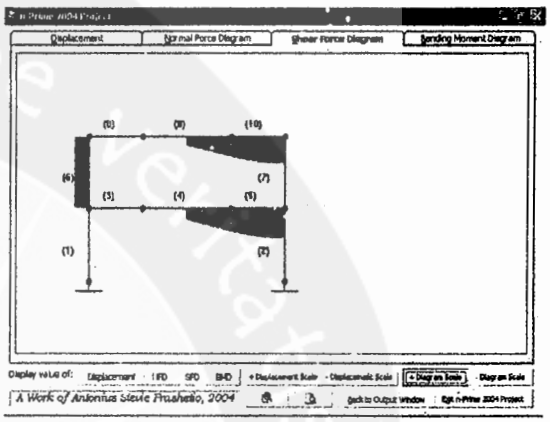
Joint ID	Fx	Fy	Mz
1	-67.45059	387.5153	269.9025
2	-232.5405	612.4842	492.7201

Tampilan Grafis *Joint Displacement*, NFD, SFD, dan BMD kasus 5

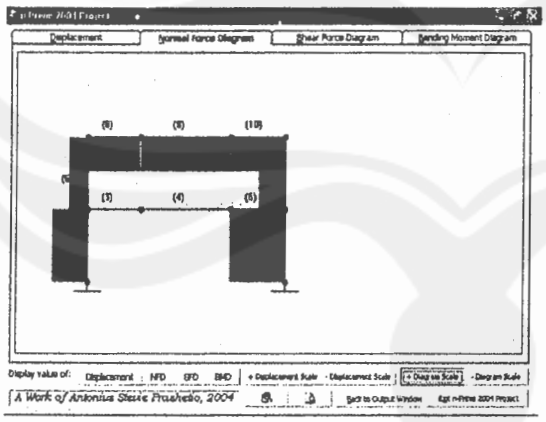
Joint Displacement



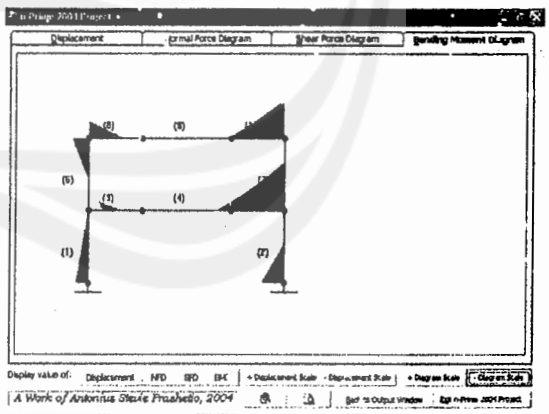
Shear Force Diagram



Normal Force Diagram



Bending Moment Diagram



Input Tables SAP2000 Kasus 5
Nama *file*: val5a.txt

SAP2000 v7.42 File: KASUS5 KN-m Units PAGE 1
5/20/04 18:14:08

Antonius Stevie Prashetio

S T A T I C L O A D C A S E S

STATIC CASE	CASE TYPE	SELF WT FACTOR
LOAD1	DEAD	0.0000

J O I N T D A T A

JOINT	GLOBAL-X	GLOBAL-Y	GLOBAL-Z	RESTRAINTS	ANGLE-A	ANGLE-B	ANGLE-C
1	0.00000	0.00000	0.00000	1 1 1 1 1 1	0.000	0.000	0.000
2	11.00000	0.00000	0.00000	1 1 1 1 1 1	0.000	0.000	0.000
3	0.00000	0.00000	4.00000	0 0 0 0 0 0	0.000	0.000	0.000
4	3.00000	0.00000	4.00000	0 0 0 0 0 0	0.000	0.000	0.000
5	8.00000	0.00000	4.00000	0 0 0 0 0 0	0.000	0.000	0.000
6	11.00000	0.00000	4.00000	0 0 0 0 0 0	0.000	0.000	0.000
7	0.00000	0.00000	8.00000	0 0 0 0 0 0	0.000	0.000	0.000
8	3.00000	0.00000	8.00000	0 0 0 0 0 0	0.000	0.000	0.000
9	8.00000	0.00000	8.00000	0 0 0 0 0 0	0.000	0.000	0.000
10	11.00000	0.00000	8.00000	0 0 0 0 0 0	0.000	0.000	0.000

F R A M E E L E M E N T D A T A

FRAME	JNT-1	JNT-2	SECTION	ANGLE	RELEASES	SEGMENTS	R1	R2	FACTOR	LENG
1	1	3	400X400	0.000	000000	2	0.000	0.000	1.000	4.0
2	2	6	400X400	0.000	000000	2	0.000	0.000	1.000	4.0
3	3	4	4T3LIN	0.000	000000	4	0.000	0.000	1.000	3.0
4	4	5	300X400	0.000	000000	4	0.000	0.000	1.000	5.0
5	5	6	3T4LIN	0.000	000000	4	0.000	0.000	1.000	3.0
6	3	7	400X400	0.000	000000	2	0.000	0.000	1.000	4.0
7	6	10	400X400	0.000	000000	2	0.000	0.000	1.000	4.0
8	7	8	4T3LIN	0.000	000000	4	0.000	0.000	1.000	3.0
9	8	9	300X400	0.000	000000	4	0.000	0.000	1.000	5.0
10	9	10	3T4LIN	0.000	000000	4	0.000	0.000	1.000	3.0

J O I N T F O R C E S Load Case LOAD1

JOINT	GLOBAL-X	GLOBAL-Y	GLOBAL-Z	GLOBAL-XX	GLOBAL-YY	GLOBAL-ZZ
7	200.000	0.000	0.000	0.000	0.000	0.000
3	100.000	0.000	0.000	0.000	0.000	0.000

F R A M E S P A N D I S T R I B U T E D L O A D S Load Case LOAD1

FRAME	TYPE	DIRECTION	DISTANCE-A	VALUE-A	DISTANCE-B	VALUE-B
8	FORCE	GLOBAL-Z	0.0000	0.0000	1.0000	-50.0000
3	FORCE	GLOBAL-Z	0.0000	0.0000	1.0000	-50.0000
5	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	0.0000
10	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	0.0000
4	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	-50.0000
9	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	-50.0000

F R A M E S P A N P O I N T L O A D S Load Case LOAD1

FRAME	TYPE	DIRECTION	DISTANCE	VALUE
4	FORCE	GLOBAL-Z	0.5000	-100.0000
9	FORCE	GLOBAL-Z	0.5000	-100.0000

Output Tables SAP2000 Kasus 5
Nama *file*: val5b.txt

SAP2000 v7.42 File: KASUS5 KN-m Units PAGE 1
5/20/04 18:14:18

Antonius Stevie Prashetio

J O I N T D I S P L A C E M E N T S

JOINT	LOAD	U1	U2	U3	R1	R2	R3
1	LOAD1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	LOAD1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
3	LOAD1	0.0298	0.0000	-3.929E-04	0.0000	0.0113	0.0000
4	LOAD1	0.0299	0.0000	-0.0375	0.0000	0.0106	0.0000
5	LOAD1	0.0301	0.0000	-0.0159	0.0000	-0.0121	0.0000
6	LOAD1	0.0302	0.0000	-6.143E-04	0.0000	3.148E-03	0.0000
7	LOAD1	0.0645	0.0000	-6.017E-04	0.0000	0.0145	0.0000
8	LOAD1	0.0643	0.0000	-0.0489	0.0000	0.0147	0.0000
9	LOAD1	0.0637	0.0000	-0.0353	0.0000	-0.0156	0.0000
10	LOAD1	0.0634	0.0000	-9.091E-04	0.0000	-5.283E-03	0.0000

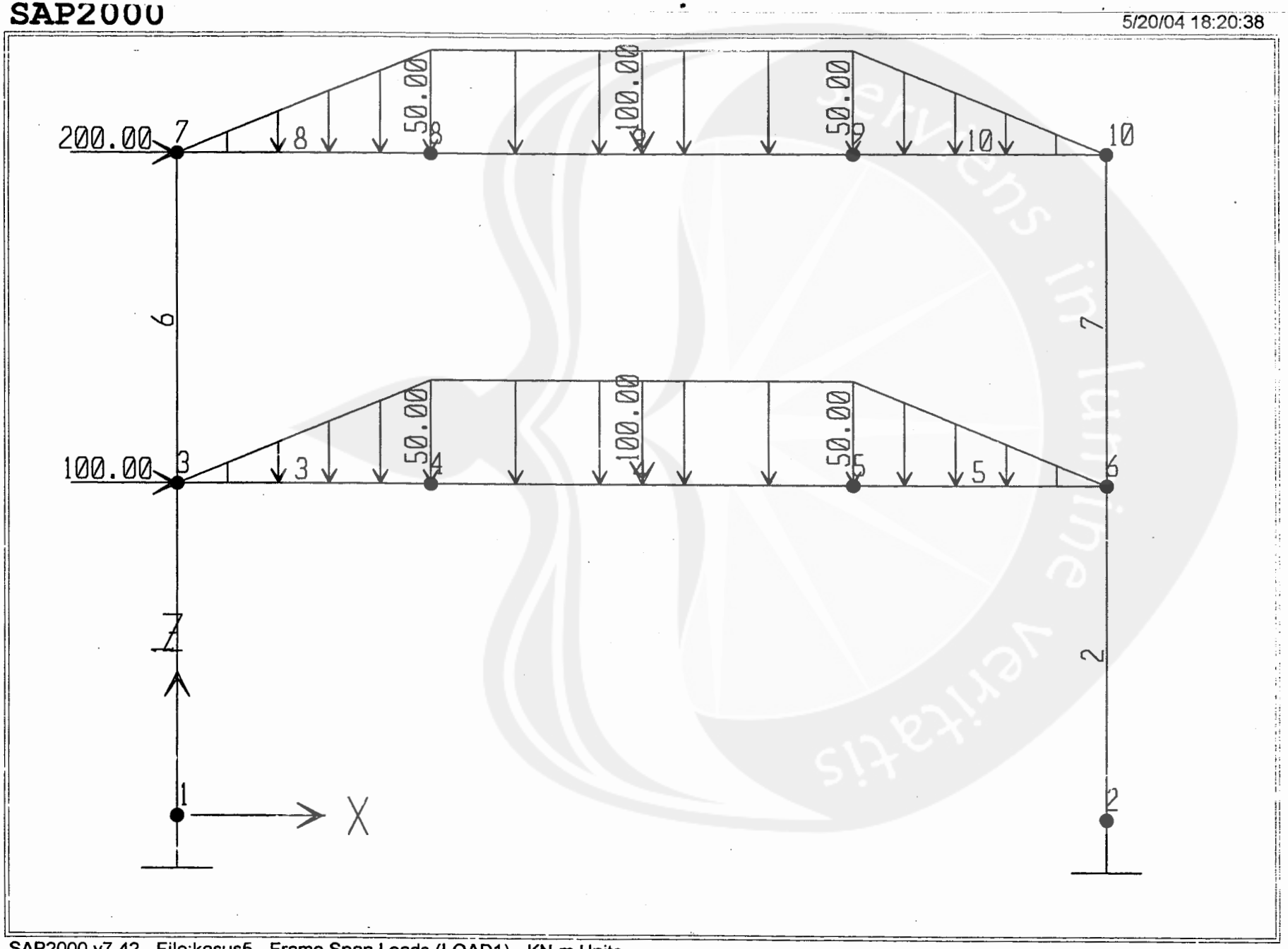
J O I N T R E A C T I O N S

JOINT	LOAD	F1	F2	F3	M1	M2	M3
1	LOAD1	-69.5514	0.0000	390.0931	0.0000	-288.4549	0.0000
2	LOAD1	-230.4486	0.0000	609.9069	0.0000	-502.5694	0.0000

F R A M E E L E M E N T F O R C E S

FRAME	LOAD	LOC	P	V2	V3	T	M2	M3
1	LOAD1							
		0.00	-390.09	69.55	0.00	0.00	0.00	288.45
		2.00	-390.09	69.55	0.00	0.00	0.00	149.35
		4.00	-390.09	69.55	0.00	0.00	0.00	10.25
2	LOAD1							
		0.00	-609.91	230.45	0.00	0.00	0.00	502.57
		2.00	-609.91	230.45	0.00	0.00	0.00	41.67
		4.00	-609.91	230.45	0.00	0.00	0.00	-419.23
3	LOAD1							
		0.00	131.38	-182.76	0.00	0.00	0.00	-291.68
		7.5E-01	131.38	-178.08	0.00	0.00	0.00	-155.78
		1.50	131.38	-164.01	0.00	0.00	0.00	-26.91
		2.25	131.38	-140.58	0.00	0.00	0.00	87.90
		3.00	131.38	-107.76	0.00	0.00	0.00	181.61
4	LOAD1							
		0.00	131.38	-107.76	0.00	0.00	0.00	181.61
		1.25	131.38	-45.26	0.00	0.00	0.00	277.25
		2.50	131.38	17.24	0.00	0.00	0.00	294.77
		3.75	131.38	179.74	0.00	0.00	0.00	109.17
		5.00	131.38	242.24	0.00	0.00	0.00	-154.57

5	LOAD1						
	0.00	131.38	242.24	0.00	0.00	0.00	-154.57
	7.5E-01	131.38	275.05	0.00	0.00	0.00	-349.13
	1.50	131.38	298.49	0.00	0.00	0.00	-564.79
	2.25	131.38	312.55	0.00	0.00	0.00	-794.52
	3.00	131.38	317.24	0.00	0.00	0.00	-1031.27
6	LOAD1						
	0.00	-207.33	-161.83	0.00	0.00	0.00	-281.43
	2.00	-207.33	-161.83	0.00	0.00	0.00	42.23
	4.00	-207.33	-161.83	0.00	0.00	0.00	365.88
7	LOAD1						
	0.00	-292.67	361.83	0.00	0.00	0.00	612.05
	2.00	-292.67	361.83	0.00	0.00	0.00	-111.61
	4.00	-292.67	361.83	0.00	0.00	0.00	-835.27
8	LOAD1						
	0.00	-361.83	-207.33	0.00	0.00	0.00	-365.88
	7.5E-01	-361.83	-202.64	0.00	0.00	0.00	-211.56
	1.50	-361.83	-188.58	0.00	0.00	0.00	-64.27
	2.25	-361.83	-165.14	0.00	0.00	0.00	68.96
	3.00	-361.83	-132.33	0.00	0.00	0.00	181.10
9	LOAD1						
	0.00	-361.83	-132.33	0.00	0.00	0.00	181.10
	1.25	-361.83	-69.83	0.00	0.00	0.00	307.45
	2.50	-361.83	-7.33	0.00	0.00	0.00	355.67
	3.75	-361.83	155.17	0.00	0.00	0.00	200.77
	5.00	-361.83	217.67	0.00	0.00	0.00	-32.26
10	LOAD1						
	0.00	-361.83	217.67	0.00	0.00	0.00	-32.26
	7.5E-01	-361.83	250.48	0.00	0.00	0.00	-208.40
	1.50	-361.83	273.92	0.00	0.00	0.00	-405.64
	2.25	-361.83	287.98	0.00	0.00	0.00	-616.94
	3.00	-361.83	292.67	0.00	0.00	0.00	-835.27


SAP2000
5/20/04 18:20:38
200.00
100.00
50.00
100.00
50.00
50.00
100.00
50.00
Z
X
SAP2000 v7.42 - File:kasus5 - Frame Span Loads (LOAD1) - KN-m Units

***Input* Data Struktural Kasus 6**
Nama *file*: kasus6a.txt

```
10    10    6     2     24821128
1     0     0
2     11    0
3     0     4
4     3     4
5     8     4
6     11    4
7     0     8
8     3     8
9     8     8
10    11    8
1     1     3     0.4   0.4   0.4   0.4   0.4   0.4
2     2     6     0.4   0.4   0.4   0.4   0.4   0.4
3     3     4     0.4   0.32  0.3   0.6   0.45  0.4
4     4     5     0.3   0.3   0.3   0.4   0.4   0.4
5     5     6     0.3   0.32  0.4   0.4   0.45  0.6
6     3     7     0.4   0.4   0.4   0.4   0.4   0.4
7     6     10    0.4   0.4   0.4   0.4   0.4   0.4
8     7     8     0.4   0.35  0.3   0.6   0.5   0.4
9     8     9     0.3   0.3   0.3   0.4   0.4   0.4
10    9     10    0.3   0.35  0.4   0.4   0.5   0.6
1     1     1     1
2     1     1     1
```

***Input* Data Beban Kasus 6**
Nama *file*: kasus6b.txt

```
2     6
3     100   0     0
7     200   0     0
3     0     0     50    3     0     0     0     0     0     0     0     0
8     0     0     50    3     0     0     0     0     0     0     0     0
5     50    0     0     3     0     0     0     0     0     0     0     0
10    50    0     0     3     0     0     0     0     0     0     0     0
4     50    0     50    5     100   2.5   0     0     0     0     0     0
9     50    0     50    5     100   2.5   0     0     0     0     0     0
```

***Output* Kasus 6**
Nama *file*: kasus6c.txt

n-Prime 2004 Project
[D:\Skripsi Stevie\New Folder\inputTA\kasus6c.txt]
Unit: kN - m

Frame Element Fixed-End Forces

Frame ID	R-i	M-i	R-j	M-j
3	25.11529	21.52877	49.88467	-21.18284
8	23.89688	19.87886	51.10318	-23.18833
5	51.10073	16.94163	23.89928	-13.63945
10	50.25166	16.38726	24.74835	-15.63229
4	174.9999	166.6665	174.9999	-166.6665
9	174.9999	166.6665	174.9999	-166.6665

Joint Displacements

Joint ID	Ux	Uy	Rz
1	0	0	0
2	0	0	0
3	2.819281E-02	-3.915277E-04	-1.067359E-02
4	2.830497E-02	-3.668658E-02	-1.065793E-02
5	2.852773E-02	-1.882586E-02	1.071658E-02
6	2.860717E-02	-6.156782E-04	-2.61854E-03
7	5.948902E-02	-5.99276E-04	-1.327043E-02
8	5.919049E-02	-4.737429E-02	-1.507688E-02
9	5.857823E-02	-3.660611E-02	1.479529E-02
10	5.836595E-02	-9.115327E-04	6.106833E-03

Frame Element Forces

Frame ID	Axial-i	Shear-i	Moment-i	Axial-j	Shear-j	Moment-j
1	-388.7264	67.96655	-277.2294	-388.7264	67.96655	-5.363156
2	-611.2731	232.0288	-498.7216	-611.2731	232.0288	429.3935
3	132.7001	182.4646	-300.4489	132.7001	107.4646	171.9448

4	132.6993	107.4645	171.9448	132.6993	-242.5353	-165.7323
5	132.6921	-242.5354	-165.7323	132.6921	-317.5354	-1043.338
6	-206.2619	-164.7313	295.0857	-206.2619	-164.7313	-363.8394
7	-293.7377	364.7254	-613.9449	-293.7377	364.7254	844.9567
8	-364.7292	206.2619	-363.8396	-364.7292	131.2619	179.9457
9	-364.727	131.262	179.9458	-364.727	-218.7378	-38.74371
10	-364.7308	-218.7377	-38.7438	-364.7308	-293.7377	-844.9565

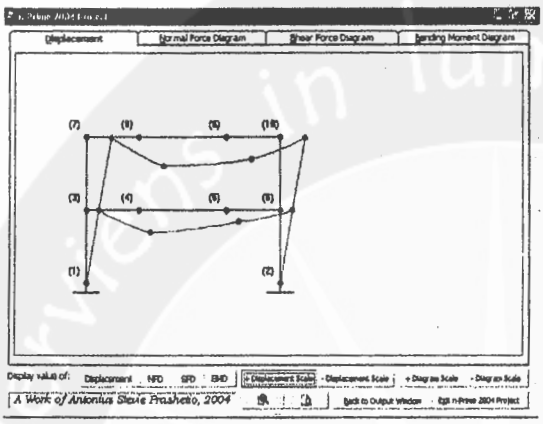
Support Reactions

Joint ID	Fx	Fy	Mz
1	-67.96655	388.7264	277.2294
2	-232.0288	611.2731	498.7216

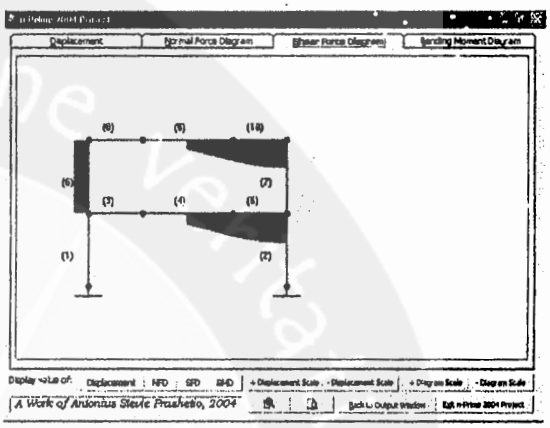
serviepin lumine veritatis

Tampilan Grafis *Joint Displacement*, NFD, SFD, dan BMD kasus 6

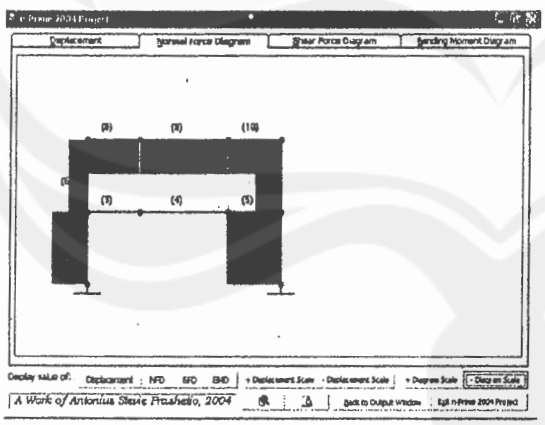
Joint Displacement



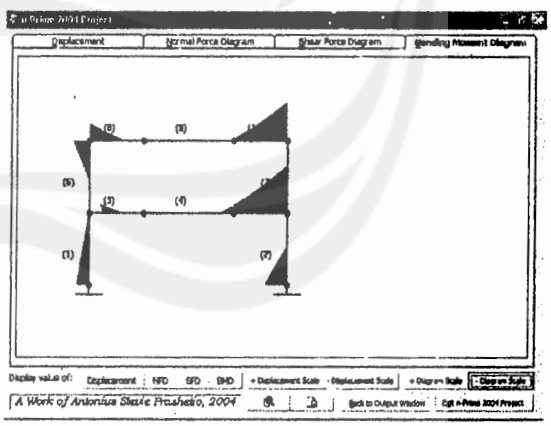
Shear Force Diagram



Normal Force Diagram



Bending Moment Diagram



Input Tables SAP2000 Kasus 6
Nama *file*: val6a.txt

SAP2000 v7.42 File: KASUS6 KN-m Units PAGE 1
5/20/04 18:19:33

Artonius Stevie Prashetio

S T A T I C L O A D C A S E S

STATIC CASE	CASE TYPE	SELF WT FACTOR
LOAD1	DEAD	0.0000

J O I N T D A T A

JOINT	GLOBAL-X	GLOBAL-Y	GLOBAL-Z	RESTRAINTS	ANGLE-A	ANGLE-B	ANGLE-C
1	0.00000	0.00000	0.00000	1 1 1 1 1 1	0.000	0.000	0.000
2	11.00000	0.00000	0.00000	1 1 1 1 1 1	0.000	0.000	0.000
3	0.00000	0.00000	4.00000	0 0 0 0 0 0	0.000	0.000	0.000
4	3.00000	0.00000	4.00000	0 0 0 0 0 0	0.000	0.000	0.000
5	8.00000	0.00000	4.00000	0 0 0 0 0 0	0.000	0.000	0.000
6	11.00000	0.00000	4.00000	0 0 0 0 0 0	0.000	0.000	0.000
7	0.00000	0.00000	8.00000	0 0 0 0 0 0	0.000	0.000	0.000
8	3.00000	0.00000	8.00000	0 0 0 0 0 0	0.000	0.000	0.000
9	8.00000	0.00000	8.00000	0 0 0 0 0 0	0.000	0.000	0.000
10	11.00000	0.00000	8.00000	0 0 0 0 0 0	0.000	0.000	0.000

F R A M E E L E M E N T D A T A

FRAME	JNT-1	JNT-2	SECTION	ANGLE	RELEASES	SEGMENTS	R1	R2	FACTOR	LENGTH
1	1	3	400X400	0.000	000000	2	0.000	0.000	1.000	4.000
2	2	6	400X400	0.000	000000	2	0.000	0.000	1.000	4.000
3	3	4	4T3PAR3	0.000	000000	4	0.000	0.000	1.000	3.000
4	4	5	300X400	0.000	000000	4	0.000	0.000	1.000	5.000
5	5	6	3T4PAR3	0.000	000000	4	0.000	0.000	1.000	3.000
6	3	7	400X400	0.000	000000	2	0.000	0.000	1.000	4.000
7	6	10	400X400	0.000	000000	2	0.000	0.000	1.000	4.000
8	7	8	4T3LIN	0.000	000000	4	0.000	0.000	1.000	3.000
9	8	9	300X400	0.000	000000	4	0.000	0.000	1.000	5.000
10	9	10	3T4LIN	0.000	000000	4	0.000	0.000	1.000	3.000

J O I N T F O R C E S Load Case LOAD1

JOINT	GLOBAL-X	GLOBAL-Y	GLOBAL-Z	GLOBAL-XX	GLOBAL-YY	GLOBAL-ZZ
7	200.000	0.000	0.000	0.000	0.000	0.000
3	100.000	0.000	0.000	0.000	0.000	0.000

F R A M E S P A N D I S T R I B U T E D L O A D S Load Case LOAD1

FRAME	TYPE	DIRECTION	DISTANCE-A	VALUE-A	DISTANCE-B	VALUE-B
8	FORCE	GLOBAL-Z	0.0000	0.0000	1.0000	-50.0000
3	FORCE	GLOBAL-Z	0.0000	0.0000	1.0000	-50.0000
5	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	0.0000
10	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	0.0000
4	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	-50.0000
9	FORCE	GLOBAL-Z	0.0000	-50.0000	1.0000	-50.0000

F R A M E S P A N P O I N T L O A D S Load Case LOAD1

FRAME	TYPE	DIRECTION	DISTANCE	VALUE
4	FORCE	GLOBAL-Z	0.5000	-100.0000
9	FORCE	GLOBAL-Z	0.5000	-100.0000

Output Tables SAP2000 Kasus 6
Nama *file*: val6b.txt

SAP2000 v7.42 File: KASUS6 KN-m Units PAGE 1
5/20/04 18:19:51

Antonius Stevie Prashetio

J O I N T D I S P L A C E M E N T S

JOINT	LOAD	U1	U2	U3	R1	R2	R3
1	LOAD1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	LOAD1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
3	LOAD1	0.0332	0.0000	-3.971E-04	0.0000	0.0127	0.0000
4	LOAD1	0.0333	0.0000	-0.0424	0.0000	0.0115	0.0000
5	LOAD1	0.0335	0.0000	-0.0176	0.0000	-0.0150	0.0000
6	LOAD1	0.0336	0.0000	-6.102E-04	0.0000	5.192E-03	0.0000
7	LOAD1	0.0720	0.0000	-6.026E-04	0.0000	0.0150	0.0000
8	LOAD1	0.0718	0.0000	-0.0498	0.0000	0.0148	0.0000
9	LOAD1	0.0712	0.0000	-0.0351	0.0000	-0.0157	0.0000
10	LOAD1	0.0709	0.0000	-9.082E-04	0.0000	-5.073E-03	0.0000

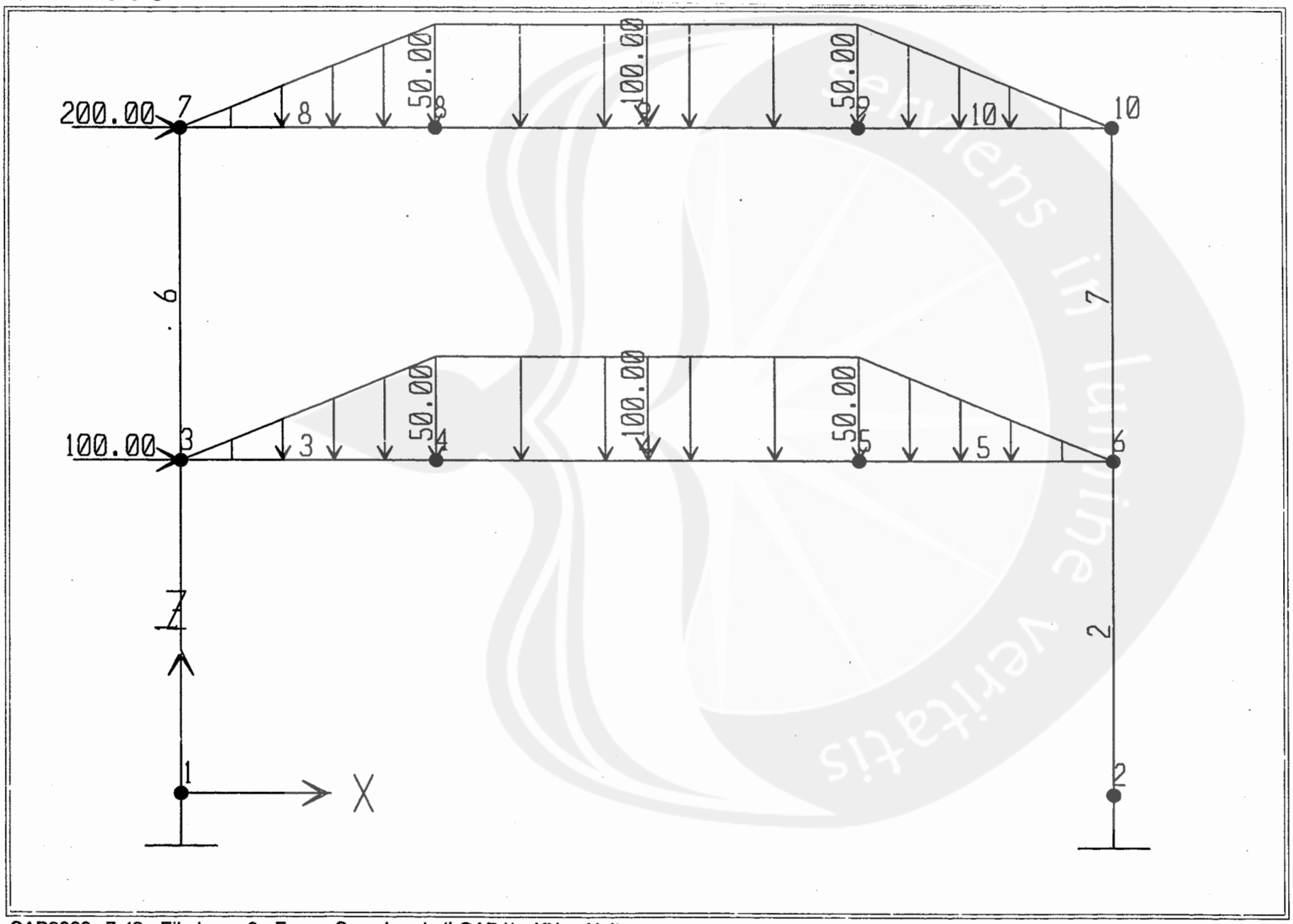
J O I N T R E A C T I O N S

JOINT	LOAD	F1	F2	F3	M1	M2	M3
1	LOAD1	-75.8064	0.0000	394.2096	0.0000	-319.1915	0.0000
2	LOAD1	-224.1936	0.0000	605.7904	0.0000	-517.1141	0.0000

F R A M E E L E M E N T F O R C E S

FRAME	LOAD	LOC	P	V2	V3	T	M2	M3
1	LOAD1							
		0.00	-394.21	75.81	0.00	0.00	0.00	319.19
		2.00	-394.21	75.81	0.00	0.00	0.00	167.58
		4.00	-394.21	75.81	0.00	0.00	0.00	15.97
2	LOAD1							
		0.00	-605.79	224.19	0.00	0.00	0.00	517.11
		2.00	-605.79	224.19	0.00	0.00	0.00	68.73
		4.00	-605.79	224.19	0.00	0.00	0.00	-379.66
3	LOAD1							
		0.00	133.77	-190.15	0.00	0.00	0.00	-301.41
		7.5E-01	133.77	-185.47	0.00	0.00	0.00	-159.97
		1.50	133.77	-171.40	0.00	0.00	0.00	-25.56
		2.25	133.77	-147.97	0.00	0.00	0.00	94.79
		3.00	133.77	-115.15	0.00	0.00	0.00	194.05
4	LOAD1							
		0.00	133.77	-115.15	0.00	0.00	0.00	194.05
		1.25	133.77	-52.65	0.00	0.00	0.00	298.93
		2.50	133.77	9.85	0.00	0.00	0.00	325.69
		3.75	133.77	172.35	0.00	0.00	0.00	149.32
		5.00	133.77	234.85	0.00	0.00	0.00	-105.18

5	LOAD1						
	0.00	133.77	234.85	0.00	0.00	0.00	-105.18
	7.5E-01	133.77	267.66	0.00	0.00	0.00	-294.20
	1.50	133.77	291.10	0.00	0.00	0.00	-504.32
	2.25	133.77	305.16	0.00	0.00	0.00	-728.50
	3.00	133.77	309.85	0.00	0.00	0.00	-959.71
6	LOAD1						
	0.00	-204.05	-157.97	0.00	0.00	0.00	-285.45
	2.00	-204.05	-157.97	0.00	0.00	0.00	30.48
	4.00	-204.05	-157.97	0.00	0.00	0.00	346.42
7	LOAD1						
	0.00	-295.95	357.97	0.00	0.00	0.00	580.05
	2.00	-295.95	357.97	0.00	0.00	0 00	-135.88
	4.00	-295.95	357.97	0.00	0.00	0.00	-851.81
8	LOAD1						
	0.00	-357.97	-204.05	0.00	0.00	0.00	-346.42
	7.5E-01	-357.97	-199.37	0.00	0.00	0.00	-194.55
	1.50	-357.97	-185.30	0.00	0.00	0.00	-49.71
	2.25	-357.97	-161.87	0.00	0.00	0.00	31.07
	3.00	-357.97	-129.05	0.00	0.00	0.00	190.75
9	LOAD1						
	0.00	-357.97	-129.05	0.00	0.00	0.00	190.75
	1.25	-357.97	-66.55	0.00	0.00	0.00	313.00
	2.50	-357.97	-4.05	0.00	0.00	0.00	357.14
	3.75	-357.97	158.45	0.00	0.00	0.00	198.14
	5.00	-357.97	220.95	0.00	0.00	0.00	-38.98
10	LOAD1						
	0.00	-357.97	220.95	0.00	0.00	0.00	-38.98
	7.5E-01	-357.97	253.76	0.00	0.00	0.00	-217.58
	1.50	-357.97	277.20	0.00	0.00	0.00	-417.27
	2.25	-357.97	291.26	0.00	0.00	0.00	-631.03
	3.00	-357.97	295.95	0.00	0.00	0.00	-851.81



SAP2000 v7.42 - File:kasus6 - Frame Span Loads (LOAD1) - KN-m Units

Lampiran Source Codes "n-Prime 2004 Project"

Input Data Struktural

```
Dim nomorfile As Integer
Sub simpan()
M = Text1.text
NJ = Text2.text
NR = Text3.text
NRJ = Text4.text
E = Text5.text
Print #1, M; Tab(9); NJ; Tab(18); NR; Tab(27); NRJ; Tab(36); E
For K = 1 To NJ
   j = (frmContoh1.MSFlexGrid1.TextMatrix(K, 0))
   xa = (frmContoh1.MSFlexGrid1.TextMatrix(K, 1))
   xb = (frmContoh1.MSFlexGrid1.TextMatrix(K, 2))
   Print #1, j; Tab(9); xa; Tab(18); xb
Next K
For j = 1 To M
   i = (frmContoh1.MSFlexGrid6.TextMatrix(j, 0))
   xa = (frmContoh1.MSFlexGrid6.TextMatrix(j, 1))
   xb = (frmContoh1.MSFlexGrid6.TextMatrix(j, 2))
   xc = (frmContoh1.MSFlexGrid6.TextMatrix(j, 3))
   xd = (frmContoh1.MSFlexGrid6.TextMatrix(j, 4))
   xe = (frmContoh1.MSFlexGrid6.TextMatrix(j, 5))
   xf = (frmContoh1.MSFlexGrid6.TextMatrix(j, 6))
   xg = (frmContoh1.MSFlexGrid6.TextMatrix(j, 7))
   xh = (frmContoh1.MSFlexGrid6.TextMatrix(j, 8))
   Print #1, i; Tab(9); xa; Tab(18); xb; Tab(27); xc; Tab(36); xd; Tab(45); xe; Tab(54); xf; Tab(63); xg; Tab(72); xh
Next j
For j = 1 To NRJ
   xa = (frmContoh1.MSFlexGrid3.TextMatrix(j, 1))
   xb = (frmContoh1.MSFlexGrid3.TextMatrix(j, 2))
   xc = (frmContoh1.MSFlexGrid3.TextMatrix(j, 3))
   K = (frmContoh1.MSFlexGrid3.TextMatrix(j, 0))
   Print #1, K; Tab(9); xa; Tab(18); xb; Tab(27); xc
Next j
End Sub
Private Sub Command2_Click()
frmGambarPortal.Show
End Sub
Private Sub Command3_Click()
bukafile
mnuFrame.Enabled = True
mnuLoad.Enabled = True
Command2.Enabled = True
End Sub
Private Sub Form_Activate()
With CboUnit
   .AddItem "lb - in"
   .AddItem "lb - ft"
   .AddItem "kip - in"
   .AddItem "kip - ft"
   .AddItem "kN - mm"
   .AddItem "kN - cm"
   .AddItem "kN - m"
   .AddItem "N - mm"
   .AddItem "N - cm"
   .AddItem "N - m"
   .AddItem "kgf - mm"
   .AddItem "kgf - cm"
   .AddItem "kgf - m"
   .AddItem "T - mm"
   .AddItem "T - cm"
   .AddItem "T - m"
End With
With MSFlexGrid1
   .ColWidth(0) = 1200
   .TextMatrix(0, 0) = "Joint ID"
   .TextMatrix(0, 1) = "X"
   .TextMatrix(0, 2) = "Y"
```

```
End With
With MSFlexGrid6
   .ColWidth(0) = 1200
   .TextMatrix(0, 0) = "Frame ID"
   .TextMatrix(0, 1) = "Start Joint"
   .TextMatrix(0, 2) = "End Joint"
   .TextMatrix(0, 3) = "B1"
   .TextMatrix(0, 4) = "B2"
   .TextMatrix(0, 5) = "B3"
   .TextMatrix(0, 6) = "H1"
   .TextMatrix(0, 7) = "H2"
   .TextMatrix(0, 8) = "H3"
End With
With MSFlexGrid3
   .ColWidth(0) = 1200
   .ColWidth(1) = 800
   .ColWidth(2) = 800
   .ColWidth(3) = 800
   .TextMatrix(0, 0) = "Joint ID"
   .TextMatrix(0, 1) = "R-Ux"
   .TextMatrix(0, 2) = "R-Uy"
   .TextMatrix(0, 3) = "R-Rz"
End With
With MSFlexGrid4
   .ColWidth(0) = 1200
   .ColWidth(1) = 1400
   .ColWidth(2) = 1400
   .TextMatrix(0, 0) = "Frame ID"
   .TextMatrix(0, 1) = "Start Joint"
   .TextMatrix(0, 2) = "End Joint"
End With
With MSFlexGrid6
   .ColWidth(0) = 1200
   .ColWidth(1) = 1400
   .ColWidth(2) = 1400
   .TextMatrix(0, 0) = "Frame ID"
   .TextMatrix(0, 1) = "Start Joint"
   .TextMatrix(0, 2) = "End Joint"
   .TextMatrix(0, 3) = "B1"
   .TextMatrix(0, 4) = "B2"
   .TextMatrix(0, 5) = "B3"
   .TextMatrix(0, 6) = "H1"
   .TextMatrix(0, 7) = "H2"
   .TextMatrix(0, 8) = "H3"
End With
Text2.SetFocus
End Sub
Private Sub mnuAbout_Click()
frmAbout.Show
End Sub
Sub bukafile()
On Error GoTo other1
dlgfile.Filter = "Text Document (*.txt)| *.txt"
dlgfile.DialogTitle = "Open File..."
dlgfile.ShowOpen
nomorfile = FreeFile
Open dlgfile.FileName For Input As nomorfile
sdata
Close nomorfile
frmContoh1.Caption = "n-Prime 2004 Project - [Joint Identifications, Frames Properties] - " + dlgfile.FileName
MSFlexGrid4.Visible = False
other1:
Close nomorfile
frmContoh1.Show
End Sub
Sub sdata()
Input #nomorfile, M, NJ, NR, NRJ, E
Text1.text = M
Text2.text = NJ
Text3.text = NR
Text4.text = NRJ
```

```
Text5.text = E
MSFlexGrid1.Rows = NJ + 1
For K = 1 To NJ
   Input #nomorfile, j, xa, xb
   With MSFlexGrid1
      .TextMatrix(K, 0) = j
      .TextMatrix(K, 1) = xa
      .TextMatrix(K, 2) = xb
   End With
Next K
MSFlexGrid6.Rows = M + 1
For j = 1 To M
   Input #nomorfile, i, xa, xb, xc, xd, xe, xf, xg, xh
   With MSFlexGrid6
      .TextMatrix(j, 0) = i
      .TextMatrix(j, 1) = xa
      .TextMatrix(j, 2) = xb
      .TextMatrix(j, 3) = xc
      .TextMatrix(j, 4) = xd
      .TextMatrix(j, 5) = xe
      .TextMatrix(j, 6) = xf
      .TextMatrix(j, 7) = xg
      .TextMatrix(j, 8) = xh
      .Visible = True
   End With
Next j
MSFlexGrid3.Rows = NRJ + 1
For j = 1 To NRJ
   Input #nomorfile, K, xa, xb, xc
   With MSFlexGrid3
      .TextMatrix(j, 0) = K
      .TextMatrix(j, 1) = xa
      .TextMatrix(j, 2) = xb
      .TextMatrix(j, 3) = xc
   End With
Next j
End Sub
Private Sub mnuExit_Click()
End
End Sub
Private Sub mnuFrame_Click()
frmGambarPortal.Show
End Sub
Private Sub mnuLoad_Click()
Form2.Show
End Sub
Private Sub mnuOpen_Click()
bukafile
mnuFrame.Enabled = True
mnuLoad.Enabled = True
Command2.Enabled = True
End Sub
Private Sub mnuSPInput_Click()
On Error GoTo debug1
dlgfile.Filter = "Text Document (*.txt)|*.txt"
dlgfile.DialogTitle = "Create Structural-Parameters Input File..."
dlgfile.DefaultExt = "txt"
dlgfile.ShowSave
Open dlgfile.FileName For Output As #1
simpan
Close #1
Exit Sub
debug1:
Close #1
frmContoh1.Show
End Sub
Private Sub Text1_KeyDown(KeyCode As Integer, Shift As Integer)
If KeyCode = 13 Then
   M = Val(Text1.text)
   MSFlexGrid4.Rows = M + 1
   MSFlexGrid6.Rows = M + 1
```

```
    For j = 1 To M
      promp = "INPUT START-JOINT OF FRAME " + Str(j)
      xa = Val(InputBox(promp, judul))
      promp = "INPUT END-JOINT OF FRAME " + Str(j)
      xb = Val(InputBox(promp, judul))
      MSFlexGrid4.TextMatrix(j, 0) = j
      MSFlexGrid6.TextMatrix(j, 0) = j
      MSFlexGrid4.TextMatrix(j, 1) = xa
      MSFlexGrid6.TextMatrix(j, 1) = xa
      MSFlexGrid4.TextMatrix(j, 2) = xb
      MSFlexGrid6.TextMatrix(j, 2) = xb
    Next j
  Text5.SetFocus
 End If
 End Sub
 Private Sub Text2_KeyDown(KeyCode As Integer, Shift As Integer)
 If KeyCode = 13 Then
    NJ = Val(Text2.text)
    MSFlexGrid1.Rows = NJ + 1
    judul = "Input Joint Coordinate"
    For K = 1 To NJ
      promp = "INPUT ABSCISSA X OF JOINT (" + Str(K) + ")"
      x = Val(InputBox(promp, judul))
      promp = "INPUT ORDINATE Y OF JOINT (" + Str(K) + ")"
      y = Val(InputBox(promp, judul))
      With MSFlexGrid1
        .TextMatrix(K, 0) = K
        .TextMatrix(K, 1) = x
        .TextMatrix(K, 2) = y
      End With
    Next K
    Text3.SetFocus
 End If
 End Sub
 Private Sub Text3_KeyDown(KeyCode As Integer, Shift As Integer)
 If KeyCode = 13 Then Text4.SetFocus
 End Sub
 Private Sub Text4_KeyDown(KeyCode As Integer, Shift As Integer)
 If KeyCode = 13 Then
    NRJ = Val(Text4.text)
    MSFlexGrid3.Rows = NRJ + 1
    judul = "Input Joint Restraints"
    For j = 1 To NRJ
      promp = "INPUT RESTRAINED-JOINT ID"
      K = Val(InputBox(promp, judul))
      promp = "TRANSLATION X OF JOINT (" + Str(K) + ")"
      xa = Val(InputBox(promp, judul))
      promp = "TRANSLATION Y OF JOINT (" + Str(K) + ")"
      xb = Val(InputBox(promp, judul))
      promp = "ROTATION Z OF JOINT (" + Str(K) + ")"
      xc = Val(InputBox(promp, judul))
      With MSFlexGrid3
        .TextMatrix(j, 0) = K
        .TextMatrix(j, 1) = xa
        .TextMatrix(j, 2) = xb
        .TextMatrix(j, 3) = xc
      End With
    Next j
    Text1.SetFocus
 End If
 End Sub
 Private Sub Text5_Change()
 mnuFrame.Enabled = True
 mnuLoad.Enabled = True
 mnuSPInput.Enabled = True
 Command2.Enabled = True
 End Sub
```

Membuat Gambar *Undeformed-Frame*

```
Option Explicit
Const PEMBEDA As Integer = 1
Public baris
Dim pengali  As Integer
Dim maxy As Long
Dim magnify As Integer
Dim x1 As Single
Dim x2 As Single
Dim y1 As Single
Dim y2 As Single
Dim i, j
Dim jumluh_batang As String
Dim jumlah_joint As String
Dim b, c, d, p, q, jumlah, E, f, g, h As Integer
Dim a(25, 25), z(25, 25), disp(25, 25), K(4, 4), titik1, titik2, kumpul(4)
Private Sub DrawAll()
Picture1.Cls
E = 0
f = 0
g = 0
h = 0
jumlah_joint = (frmContoh1.Text2.text)
For i = 1 To jumlah_joint
   For j = 1 To 2
      a(i, j) = frmContoh1.MSFlexGrid1.TextMatrix(i, j)
   Next j
Next i
jumlah = (frmContoh1.Text4.text)
For i = 1 To jumlah
   For j = 1 To 3
      K(i, j) = frmContoh1.MSFlexGrid3.TextMatrix(i, j)
   Next j
   kumpul(i) = frmContoh1.MSFlexGrid3.TextMatrix(i, 0)
   p = a(kumpul(i), 1) + 4
   q = a(kumpul(i), 2) + 4
   If K(i, 1) = 1 And K(i, 2) = 1 And K(i, 3) = 1 Then
      Garis Picture1, p * pengali, q * pengali, p * pengali, (q - 0.5) * pengali, &HFF0000
      Garis Picture1, (p - 0.75) * pengali, (q - 0.5) * pengali, (p + 0.75) * pengali, (q - 0.5) * pengali, &HFF0000
   ElseIf K(i, 1) = 1 And K(i, 2) = 1 And K(i, 3) = 0 Then
      Garis Picture1, (p - 0.75) * pengali, (q - 0.75) * pengali, p * pengali, q * pengali, &HFF0000
      Garis Picture1, p * pengali, q * pengali, (p + 0.75) * pengali, (q - 0.75) * pengali, &HFF0000
      Garis Picture1, (p + 0.75) * pengali, (q - 0.75) * pengali, (p - 0.75) * pengali, (q - 0.75) * pengali, &HFF0000
   Else
      Lingkar Picture1, p * pengali, (q - 0.25) * pengali, 0.3 * pengali, &HFF0000
      Garis Picture1, (p - 0.75) * pengali, (q - 0.6) * pengali, (p + 0.75) * pengali, (q - 0.6) * pengali, &HFF0000
   End If
Next i
jumlah_batang = (frmContoh1.Text1.text)
For i = 1 To jumlah_batang
   titik1 = (frmContoh1.MSFlexGrid6.TextMatrix(i, 1))
   titik2 = (frmContoh1.MSFlexGrid6.TextMatrix(i, 2))
   E = a(titik1, 1) + 4
   f = a(titik1, 2) + 4
   g = a(titik2, 1) + 4
   h = a(titik2, 2) + 4
   Garis Picture1, E * pengali, f * pengali, g * pengali, h * pengali, vbBlack & i
   Picture1.FillStyle = 0
   Lingkar Picture1, E * pengali, f * pengali, 0.2 * pengali, &HC000& & titik1
   Lingkar Picture1, g * pengali, h * pengali, 0.2 * pengali, &HC000& & titik2
   Picture1.FillStyle = 1
Next i
End Sub
Private Sub Garis(pic As PictureBox, x1 As Single, y1 As Single, x2 As Single, y2 As Single, warna As Long)
pic.Line (Round(x1), Round(maxy - y1))-(Round(x2), Round(maxy - y2)), warna
End Sub
Private Sub Lingkar(pic As PictureBox, x As Single, y As Single, rad As Single, warna As Long)
pic.Circle (Round(x), Round(maxy - y)), rad, warna
End Sub
Private Sub titik(pic As PictureBox, x As Single, y As Single, warna As Long)
pic.PSet (Round(x), Round(maxy - y)), warna
```

```
End Sub
Private Sub Command1_Click()
pengali = pengali + pengali / magnify
DrawAll
End Sub
Private Sub Command2_Click()
pengali = pengali - pengali / magnify
DrawAll
End Sub
Private Sub Command3_Click()
frmContoh1.Show
End Sub
Private Sub Command4_Click()
Form2.Show
Unload Me
End Sub
Private Sub Form_Activate()
mnuLoad.Enabled = True
Label7.Caption = " Unit: " + frmContoh1.CboUnit.text
maxy = Picture1.Height
Picture1.BackColor = vbWhite
Picture1.DrawWidth = 2
DrawAll
End Sub
Private Sub Form_Load()
pengali = 300
magnify = 10
baris = 0
End Sub
Private Sub mnuExit_Click()
End
End Sub
Private Sub mnuJoint_Click()
frmContoh1.Show
End Sub
Private Sub mnuLoad_Click()
Form2.Show
Unload Me
End Sub
Private Sub mnuRun_Click()
frmContoh2.Show
End Sub
Private Sub mnuZoomIn_Click()
pengali = pengali + pengali / magnify
DrawAll
End Sub
Private Sub mnuZoomOut_Click()
pengali = pengali - pengali / magnify
DrawAll
End Sub
Private Sub Picture1_Click()
If Picture1.MousePointer = 99 Then
   Form3.Show
End If
End Sub
Private Sub Picture1_MouseMove(Button As Integer, Shift As Integer, x As Single, y As Single)
Dim jumlah_batang As Integer
Dim i As Integer
jumlah_batang = (frmContoh1.Text1.text)
Label1.Caption = ""
For i = 1 To jumlah_batang
   If Picture1.Point(x, y) = vbBlack & i Then
      baris = i
      Picture1.MousePointer = 99
      Label1.Caption = "Frame Number: " & i
      Exit Sub
   End If
Next i
For i = 1 To jumlah_joint
   If Picture1.Point(x, y) = &HC000& & i Then
      Picture1.MousePointer = 1
```

```
      Label1.Caption = "Joint Number: " & i
      Exit Sub
    End If
Next i
Picture1.MousePointer = 0
End Sub
```

***Input* Dimensi Batang**

```
Private Sub Command1_Click()
frmGambarPortal.Show
frmContoh1.MSFlexGrid6.Row = frmGambarPortal.baris
frmContoh1.MSFlexGrid6.Col = 3
frmContoh1.MSFlexGrid6.text = Form3.Text3
frmContoh1.MSFlexGrid6.Col = 4
frmContoh1.MSFlexGrid6.text = Form3.Text2
frmContoh1.MSFlexGrid6.Col = 5
frmContoh1.MSFlexGrid6.text = Form3.Text1
frmContoh1.MSFlexGrid6.Col = 6
frmContoh1.MSFlexGrid6.text = Form3.Text4
frmContoh1.MSFlexGrid6.Col = 7
frmContoh1.MSFlexGrid6.text = Form3.Text5
frmContoh1.MSFlexGrid6.Col = 8
frmContoh1.MSFlexGrid6.text = Form3.Text6
frmContoh1.MSFlexGrid6.Visible = True
frmContoh1.MSFlexGrid4.Visible = False
Unload Me
End Sub
Private Sub Command2_Click()
Unload Me
End Sub
Private Sub Form_Activate()
Label8.Caption = " Unit: " + frmContoh1.CboUnit.text
End Sub
```

Input Data Beban

```
Dim nomorfile As Integer
Private Sub ChkJoint_Click()
If ChkJoint.Value = 1 Then
   Text1.Enabled = True
   Text1.text = ""
   Text1.SetFocus
ElseIf ChkJoint.Value = 0 Then
   Text1.Enabled = False
   Text1.text = "0"
End If
End Sub
Private Sub ChkTrap_Click()
If ChkTrap.Value = 1 Then
   Text2.Enabled = True
   Text2.text = ""
   Text2.SetFocus
ElseIf ChkTrap.Value = 0 Then
   Text2.Enabled = False
   Text2.text = "0"
End If
End Sub
Sub ldata()
Input #nomorfile, NLJ, nlm
Text1.text = NLJ
Text2.text = nlm
If NLJ = 0 Then GoTo 2
MSFlexGrid1.Rows = NLJ + 1
For j = 1 To NLJ
   Input #nomorfile, K, xa, xb, xc
   With MSFlexGrid1
      .TextMatrix(j, 0) = K
      .TextMatrix(j, 1) = xa
      .TextMatrix(j, 2) = xb
      .TextMatrix(j, 3) = xc
   End With
Next j
```

```
2: If nlm = 0 Then GoTo 4
MSFlexGrid2.Rows = nlm + 1
For j = 1 To nlm
   Input #nomorfile, i, qa, ra, qb, rb, qc, rc, qd, rd, qe, re, qf, rf
   With MSFlexGrid2
      .TextMatrix(j, 0) = i
      .TextMatrix(j, 1) = qa
      .TextMatrix(j, 2) = ra
      .TextMatrix(j, 3) = qb
      .TextMatrix(j, 4) = rb
      .TextMatrix(j, 5) = qc
      .TextMatrix(j, 6) = rc
      .TextMatrix(j, 7) = qd
      .TextMatrix(j, 8) = rd
      .TextMatrix(j, 9) = qe
      .TextMatrix(j, 10) = re
      .TextMatrix(j, 11) = qf
      .TextMatrix(j, 12) = rf
   End With
Next j
4: End Sub
Sub simpan()
NLJ = Text1.text
nlm = Text2.text
Print #1, NLJ; Tab(9); nlm
For K = 1 To NLJ
   j = (MSFlexGrid1.TextMatrix(K, 0))
   xa = (MSFlexGrid1.TextMatrix(K, 1))
   xb = (MSFlexGrid1.TextMatrix(K, 2))
   xc = (MSFlexGrid1.TextMatrix(K, 3))
   Print #1, j; Tab(9); xa; Tab(18); xb; Tab(27); xc
Next K
For j = 1 To nlm
   i = (MSFlexGrid2.TextMatrix(j, 0))
   xa = (MSFlexGrid2.TextMatrix(j, 1))
   xb = (MSFlexGrid2.TextMatrix(j, 2))
   xc = (MSFlexGrid2.TextMatrix(j, 3))
   xd = (MSFlexGrid2.TextMatrix(j, 4))
   xe = (MSFlexGrid2.TextMatrix(j, 5))
   xf = (MSFlexGrid2.TextMatrix(j, 6))
   xg = (MSFlexGrid2.TextMatrix(j, 7))
   xh = (MSFlexGrid2.TextMatrix(j, 8))
   xi = (MSFlexGrid2.TextMatrix(j, 9))
   xj = (MSFlexGrid2.TextMatrix(j, 10))
   xk = (MSFlexGrid2.TextMatrix(j, 11))
   xl = (MSFlexGrid2.TextMatrix(j, 12))
   Print #1, i; Tab(9); xa; Tab(18); xb; Tab(27); xc; Tab(36); xd; Tab(45); xe; Tab(54); xf; Tab(63); xg; Tab(72); xh; Tab(81); xi;
Tab(100); xj; Tab(109); xk; Tab(118); xl
Next j
End Sub
Private Sub Command1_Click()
frmContoh1.Show
End Sub
Private Sub Command2_Click()
frmContoh2.Show
End Sub
Private Sub Command3_Click()
frmGambarPortal.Show
End Sub
Private Sub Command4_Click()
On Error GoTo other2
dlgfile.Filter = "Text Document (*.txt)| *.txt"
dlgfile.DialogTitle = "Open"
dlgfile.ShowOpen
nomorfile = FreeFile
Open dlgfile.FileName For Input As nomorfile
ldata
Close nomorfile
frmContoh1.Caption = dlgfile.FileName
other2:
Close nomorfile
```

```
Form2.Show
End Sub
Private Sub Form_Activate()
Label1.Caption = " Unit: " + frmContoh1.CboUnit.text
With MSFlexGrid1
   .ColWidth(0) = 1200
   .ColWidth(1) = 1300
   .ColWidth(2) = 1300
   .ColWidth(3) = 1300
   .TextMatrix(0, 0) = "Joint ID"
   .TextMatrix(0, 1) = "Point X"
   .TextMatrix(0, 2) = "Point Y"
   .TextMatrix(0, 3) = "Moment Z"
End With
With MSFlexGrid2
   .ColWidth(0) = 1000
   .ColWidth(2) = 600
   .ColWidth(4) = 600
   .ColWidth(5) = 1000
   .ColWidth(6) = 600
   .ColWidth(7) = 1000
   .ColWidth(8) = 600
   .ColWidth(9) = 1000
   .ColWidth(10) = 600
   .ColWidth(11) = 1000
   .ColWidth(12) = 600
   .TextMatrix(0, 0) = "Frame ID"
   .TextMatrix(0, 1) = "Trap-1"
   .TextMatrix(0, 2) = "d-t1"
   .TextMatrix(0, 3) = "Trap-2"
   .TextMatrix(0, 4) = "d-t2"
   .TextMatrix(0, 5) = "Point-1"
   .TextMatrix(0, 6) = "d-p1"
   .TextMatrix(0, 7) = "Point-2"
   .TextMatrix(0, 8) = "d-p2"
   .TextMatrix(0, 9) = "Point-3"
   .TextMatrix(0, 10) = "d-p3"
   .TextMatrix(0, 11) = "Point-4"
   .TextMatrix(0, 12) = "d-p4"
End With
End Sub
Private Sub mnuCreate_Click()
On Error GoTo other3
dlgfile.Filter = "Text Document (*.txt)|*.txt"
Form2.Caption = "n-Prime 2004 Project - [Assign Frame and Joint Loading] " & dlgfile.FileName
dlgfile.DialogTitle = "Create Loading Data File..."
dlgfile.DefaultExt = "txt"
dlgfile.ShowSave
Open dlgfile.FileName For Output As #1
simpan
Close #1
Exit Sub
other3:
Form2.Show
End Sub
Private Sub mnuExit_Click()
End
End Sub
Private Sub mnuFrame_Click()
frmGambarPortal.Show
End Sub
Private Sub mnuJoint_Click()
frmContoh1.Show
End Sub
Private Sub mnuRun_Click()
frmContoh2.Show
End Sub
Private Sub mnuLoading_Click()
On Error GoTo other2
dlgfile.Filter = "Text Document (*.txt)| *.txt"
dlgfile.DialogTitle = "Open"
```

```
dlgfile.ShowOpen
nomorfile = FreeFile
Open dlgfile.FileName For Input As nomorfile
ldata
Close nomorfile
frmContoh1.Caption = dlgfile.FileName
other2:
Close nomorfile
Form2.Show
End Sub
Private Sub Text1_KeyDown(KeyCode As Integer, Shift As Integer)
If KeyCode = 13 Then
  NLJ = Val(Text1.text)
  MSFlexGrid1.Rows = NLJ + 1
  judul = "Input Loads on Joint"
  For j = 1 To NLJ
    promp = "INPUT JOINT ID"
    K = Val(InputBox(promp, judul))
    promp = "INPUT POINT LOAD X ON JOINT (" + Str(j) + ")"
    xa = Val(InputBox(promp, judul))
    promp = "INPUT POINT LOAD Y ON JOINT (" + Str(j) + ")"
    xb = Val(InputBox(promp, judul))
    promp = "INPUT MOMENT Z ON JOINT (" + Str(j) + ")"
    xc = Val(InputBox(promp, judul))
    With MSFlexGrid1
      .TextMatrix(j, 0) = K
      .TextMatrix(j, 1) = xa
      .TextMatrix(j, 2) = xb
      .TextMatrix(j, 3) = xc
    End With
  Next j
End If
End Sub
Private Sub Text2_KeyDown(KeyCode As Integer, Shift As Integer)
If KeyCode = 13 Then
  nlm = Val(Text2.text)
  MSFlexGrid2.Rows = nlm + 1
  judul = "Input Loads on Frame"
  For j = 1 To nlm
    promp = "INPUT FRAME ID"
    i = Val(InputBox(promp, judul))
    promp = "INPUT TRAPEZOIDAL SPAN LOAD 1 ON FRAME (" + Str(j) + ")"
    xa = Val(InputBox(promp, judul))
    promp = "INPUT DISTANCE OF TRAPEZOIDAL SPAN LOAD 1 ON FRAME (" + Str(j) + ")"
    xb = Val(InputBox(promp, judul))
    promp = "INPUT TRAPEZOIDAL SPAN LOAD 2 ON FRAME (" + Str(j) + ")"
    xc = Val(InputBox(promp, judul))
    promp = "INPUT DISTANCE OF TRAPEZOIDAL LOAD 2 ON FRAME (" + Str(j) + ")"
    xd = Val(InputBox(promp, judul))
    promp = "INPUT POINT LOAD 1 ON FRAME (" + Str(j) + ")"
    xe = Val(InputBox(promp, judul))
    promp = "INPUT DISTANCE OF POINT LOAD 1 ON FRAME (" + Str(j) + ")"
    xf = Val(InputBox(promp, judul))
    promp = "INPUT POINT LOAD 2 ON FRAME (" + Str(j) + ")"
    xg = Val(InputBox(promp, judul))
    promp = "INPUT DISTANCE OF POINT LOAD 2 ON FRAME (" + Str(j) + ")"
    xh = Val(InputBox(promp, judul))
    promp = "INPUT POINT LOAD 3 ON FRAME (" + Str(j) + ")"
    xi = Val(InputBox(promp, judul))
    promp = "INPUT DISTANCE OF POINT LOAD 3 ON FRAME (" + Str(j) + ")"
    xj = Val(InputBox(promp, judul))
    promp = "INPUT POINT LOAD 4 ON FRAME (" + Str(j) + ")"
    xk = Val(InputBox(promp, judul))
    promp = "INPUT DISTANCE OF POINT LOAD 4 ON FRAME (" + Str(j) + ")"
    xl = Val(InputBox(promp, judul))
    With MSFlexGrid2
      .TextMatrix(j, 0) = i
      .TextMatrix(j, 1) = xa
      .TextMatrix(j, 2) = xb
      .TextMatrix(j, 3) = xc
      .TextMatrix(j, 4) = xd
```

```
        .TextMatrix(j, 5) = xe
        .TextMatrix(j, 6) = xf
        .TextMatrix(j, 7) = xg
        .TextMatrix(j, 8) = xh
        .TextMatrix(j, 9) = xi
        .TextMatrix(j, 10) = xj
        .TextMatrix(j, 11) = xk
        .TextMatrix(j, 12) = xl
      End With
    Next j
End If
End Sub
```

Menghitung *Joint Displacement*, NF, SF, dan BM

```
Dim M As Integer, NJ As Integer, NR As Integer, NRJ As Integer, E As Single, NPL As Single
Dim SM() As Single, Kx As Single, Bw1() As Single, Bw2() As Single, Bw3() As Single, H1() As Single
Dim H2() As Single, H3() As Single, p() As Single, q() As Single, v() As Single, w() As Single, ab() As Single
Dim bc() As Single, cd() As Single, de() As Single, ef() As Single, fg() As Single, gh() As Single
Dim DF() As Single, DJ() As Single, NDJ As Integer, ND As Integer, z() As Single
Dim x() As Single, y() As Single, b() As Single, j As Integer, K As Integer
Dim SFF() As Single, c As Single, AC() As Single, i As Integer
Dim SCM1 As Single, SCM11 As Single, SCM12 As Single, SCM14 As Single, SCM22 As Single
Dim xa As Single, xb As Single, xc As Single, xd As Single, xe As Single   'q1() As Single
Dim xf As Single, xg As Single, xh As Single, SCM24 As Single, SCM44 As Single
Dim JJ() As Single, JK() As Single, EL() As Single, CX() As Single, Ki As Single
Dim CY() As Single, NB1 As Integer, XCL As Single, YCL As Single, SMS() As Single
Dim JRL() As Integer, N1 As Integer, ID() As Integer, a() As Single, IM() As Integer
Dim N As Integer, NB As Integer, JE As Integer, nomorfile As Integer
Dim I1 As Integer, I2 As Integer, IR As Integer, IC As Integer, JR As Integer
Dim TEMP As Single, SUM As Single, ITEM As Integer, J1 As Integer, J2 As Integer
Dim J3 As Integer, K3 As Integer, AMD() As Single, AM() As Single, MD As Integer
Dim u() As Single, K1 As Integer, K2 As Integer, NLJ As Integer, nlm As Integer, NLMP As Integer
Dim AML() As Single, LML() As Single, AJ() As Single, AR() As Single, AE() As Single
Dim q1() As Single, q2() As Single, r1() As Single, r4() As Single, r2() As Single, r3() As Single
Dim q3() As Single, r5() As Single, r6() As Single
Dim gauss As Integer
Dim xgauss As Single, cgauss As Single
Dim ba1() As Double, ha1() As Single, inertiaa1() As Single, intega1() As Single, valuea1() As Single
Dim bb1() As Single, hb1() As Single, inertiab1() As Single, integb1() As Single, valueb1() As Single
Dim bc1() As Single, hc1() As Single, inertiac1() As Single, integc1() As Single, valuec1() As Single
Dim bd1() As Single, hd1() As Single, inertiad1() As Single, integd1() As Single, valued1() As Single
Dim be1() As Single, he1() As Single, inertiae1() As Single, intege1() As Single, valuee1() As Single
Dim bf1() As Single, hf1() As Single, inertiaf1() As Single, integf1() As Single, valuef1() As Single
Dim bg1() As Single, hg1() As Single, inertiag1() As Single, integg1() As Single, valueg1() As Single
Dim bh1() As Single, hh1() As Single, inertiah1() As Single, integh1() As Single, valueh1() As Single
Dim bi1() As Single, hi1() As Single, inertiai1() As Single, integi1() As Single, valuei1() As Single
Dim bj1() As Single, hj1() As Single, inertiaj1() As Single, integj1() As Single, valuej1() As Single
Dim bk1() As Single, hk1() As Single, inertiak1() As Single, integk1() As Single, valuek1() As Single
Dim bl1() As Single, hl1() As Single, inertial1() As Single, integl1() As Single, valuel1() As Single
Dim eone() As Single, fone() As Single
Dim momenttrap1() As Single, reactiontrap1() As Single
Dim ba2() As Single, ha2() As Single, inertiaa2() As Single, intega2() As Single, valuea2() As Single
Dim bb2() As Single, hb2() As Single, inertiab2() As Single, integb2() As Single, valueb2() As Single
Dim bc2() As Single, hc2() As Single, inertiac2() As Single, integc2() As Single, valuec2() As Single
Dim bd2() As Single, hd2() As Single, inertiad2() As Single, integd2() As Single, valued2() As Single
Dim be2() As Single, he2() As Single, inertiae2() As Single, intege2() As Single, valuee2() As Single
Dim bf2() As Single, hf2() As Single, inertiaf2() As Single, integf2() As Single, valuef2() As Single
Dim bg2() As Single, hg2() As Single, inertiag2() As Single, integg2() As Single, valueg2() As Single
Dim bh2() As Single, hh2() As Single, inertiah2() As Single, integh2() As Single, valueh2() As Single
Dim bi2() As Single, hi2() As Single, inertiai2() As Single, integi2() As Single, valuei2() As Single
Dim bj2() As Single, hj2() As Single, inertiaj2() As Single, integj2() As Single, valuej2() As Single
Dim bk2() As Single, hk2() As Single, inertiak2() As Single, integk2() As Single, valuek2() As Single
Dim bl2() As Single, hl2() As Single, inertial2() As Single, integl2() As Single, valuel2() As Single
Dim etwo() As Single, ftwo() As Single
Dim momenttrap2() As Single, reactiontrap2() As Single
Dim bap1() As Single, hap1() As Single, inertiaap1() As Single, integap1() As Single, valueap1() As Single
Dim bbp1() As Single, hbp1() As Single, inertiabp1() As Single, integbp1() As Single, valuebp1() As Single
Dim bcp1() As Single, hcp1() As Single, inertiacp1() As Single, integcp1() As Single, valuecp1() As Single
Dim bdp1() As Single, hdp1() As Single, inertiadp1() As Single, integdp1() As Single, valuedp1() As Single
Dim bep1() As Single, hep1() As Single, inertiaep1() As Single, integep1() As Single, valueep1() As Single
Dim bfp1() As Single, hfp1() As Single, inertiafp1() As Single, integfp1() As Single, valuefp1() As Single
```

```
Dim bgp1() As Single, hgp1() As Single, inertiagp1() As Single, integgp1() As Single, valuegp1() As Single
Dim bhp1() As Single, hhp1() As Single, inertiahp1() As Single, integhp1() As Single, valuehp1() As Single
Dim bkp1() As Single, hkp1() As Single, inertiakp1() As Single, integkp1() As Single, valuekp1() As Single
Dim momentpoint1() As Single, reactionpoint1() As Single
Dim momentpoint1total() As Single, reactionpoint1total() As Single
Dim bap2() As Single, hap2() As Single, inertiaap2() As Single, integap2() As Single, valuеap2() As Single
Dim bbp2() As Single, hbp2() As Single, inertiabp2() As Single, integbp2() As Single, valuebp2() As Single
Dim bcp2() As Single, hcp2() As Single, inertiacp2() As Single, integcp2() As Single, valuecp2() As Single
Dim bdp2() As Single, hdp2() As Single, inertiadp2() As Single, integdp2() As Single, valuedp2() As Single
Dim bep2() As Single, hep2() As Single, inertiaep2() As Single, integep2() As Single, valueep2() As Single
Dim bfp2() As Single, hfp2() As Single, inertiafp2() As Single, integfp2() As Single, valuefp2() As Single
Dim bgp2() As Single, hgp2() As Single, inertiagp2() As Single, integgp2() As Single, valuegp2() As Single
Dim bhp2() As Single, hhp2() As Single, inertiahp2() As Single, integhp2() As Single, valuehp2() As Single
Dim bkp2() As Single, hkp2() As Single, inertiakp2() As Single, integkp2() As Single, valuekp2() As Single
Dim momentpoint2() As Single, reactionpoint2() As Single
Dim momentpoint2total() As Single, reactionpoint2total() As Single
Private Sub CmdPrev_Click()
FRGambar.Show
End Sub
Private Sub Command2_Click()
End
End Sub
Private Sub Form_Load()
lagi:
Label1.Caption = " Unit: " + frmContoh1.CboUnit.text
sdata
stiff
Call banfac(N, NB, SFF, c)
ReDim u(N, NB)
ReDim AMD(6)
ReDim AM(6)
ldata
loads
Call bansol(N, NB, SFF, AC, DF)
ReDim AR(3 * (M + M))
On Error GoTo debug9
dlgfile.Filter = "Text Document (*.txt)|*.txt"
dlgfile.DialogTitle = "Create Ouput File..."
dlgfile.DefaultExt = "txt"
dlgfile.ShowSave
Open dlgfile.FileName For Output As #1
resul
Close #1
frmContoh2.Caption = "n-Prime 2004 Project - [Output File: " & dlgfile.FileName & "]"
Exit Sub
debug9:
Close #1
frmContoh2.Show
GoTo lagi
End Sub
Sub sdata()
M = (frmContoh1.Text1.text)
NJ = (frmContoh1.Text2.text)
NR = (frmContoh1.Text3.text)
NRJ = (frmContoh1.Text4.text)
E = (frmContoh1.Text5.text)
NDJ = 3
ND = NDJ * NJ
N = ND - NR
ReDim AML(6, M)
ReDim DF(N)
ReDim DJ(ND)
ReDim z(N)
ReDim x(NJ)
ReDim y(NJ)
ReDim b(N)
For K = 1 To NJ
   j = (frmContoh1.MSFlexGrid1.TextMatrix(K, 0))
   xa = (frmContoh1.MSFlexGrid1.TextMatrix(K, 1))
   xb = (frmContoh1.MSFlexGrid1.TextMatrix(K, 2))
   x(j) = xa
```

```
    y(j) = xb
Next K
MD = 2 * NDJ
NB = 0
ReDim JJ(M)
ReDim JK(M)
ReDim EL(M)
ReDim CX(M)
ReDim CY(M)
ReDim Bw1(M)
ReDim Bw2(M)
ReDim Bw3(M)
ReDim H1(M)
ReDim H2(M)
ReDim H3(M)
ReDim w(M)
ReDim v(M)
ReDim p(M)
ReDim q(M)
ReDim ab(M)
ReDim bc(M)
ReDim cd(M)
ReDim de(M)
ReDim ef(M)
ReDim fg(M)
ReDim gh(M)
For j = 1 To M
    i = (frmContoh1.MSFlexGrid6.TextMatrix(j, 0))
    xa = (frmContoh1.MSFlexGrid6.TextMatrix(j, 1))
    xb = (frmContoh1.MSFlexGrid6.TextMatrix(j, 2))
    xc = (frmContoh1.MSFlexGrid6.TextMatrix(j, 3))
    xd = (frmContoh1.MSFlexGrid6.TextMatrix(j, 4))
    xe = (frmContoh1.MSFlexGrid6.TextMatrix(j, 5))
    xf = (frmContoh1.MSFlexGrid6.TextMatrix(j, 6))
    xg = (frmContoh1.MSFlexGrid6.TextMatrix(j, 7))
    xh = (frmContoh1.MSFlexGrid6.TextMatrix(j, 8))
    JJ(i) = xa
    JK(i) = xb
    Bw1(i) = xc
    Bw2(i) = xd
    Bw3(i) = xe
    H1(i) = xf
    H2(i) = xg
    H3(i) = xh
    NBI = NDJ * (Abs(JK(i) - JJ(i)) + 1)
    If NBI > NB Then NB = NBI
    XCL = x(JK(i)) - x(JJ(i))
    YCL = y(JK(i)) - y(JJ(i))
    EL(i) = Sqr(XCL * XCL + YCL * YCL)
    CX(i) = XCL / EL(i)
    CY(i) = YCL / EL(i)
Next j
ReDim JRL(ND)
For j = 1 To NRJ
    xa = (frmContoh1.MSFlexGrid3.TextMatrix(j, 1))
    xb = (frmContoh1.MSFlexGrid3.TextMatrix(j, 2))
    xc = (frmContoh1.MSFlexGrid3.TextMatrix(j, 3))
    K = (frmContoh1.MSFlexGrid3.TextMatrix(j, 0))
    JRL(3 * K - 2) = xa
    JRL(3 * K - 1) = xb
    JRL(3 * K) = xc
Next j
N1 = 0
ReDim ID(ND)
For j = 1 To ND
    N1 = N1 + JRL(j)
    If JRL(j) > 0 Then
        ID(j) = N + N1
    Else
        ID(j) = j - N1
    End If
```

```
Next j
ReDim a(N, NB)
End Sub
Sub stiff()
ReDim SFF(N, NB)
ReDim IM(6)
ReDim SMS(6, 6)
For i = 1 To M
   w(i) = 2 * (Bw1(i) + Bw2(i) - 2 * Bw3(i))
   v(i) = Bw1(i) - Bw2(i) + w(i)
   p(i) = 2 * (H1(i) + H2(i) - 2 * H3(i))
   q(i) = H1(i) - H2(i) + p(i)
   ab(i) = H1(i) ^ 3
   bc(i) = -3 * (H1(i) ^ 2) * q(i)
   cd(i) = 3 * H1(i) * q(i) ^ 2 + 3 * p(i) * H1(i) ^ 2
   de(i) = -(q(i) ^ 3 + 6 * p(i) * H1(i) * q(i))
   ef(i) = 3 * p(i) * q(i) ^ 2 + 3 * H1(i) * p(i) ^ 2
   fg(i) = -3 * q(i) * p(i) ^ 2
   gh(i) = p(i) ^ 3
   Kx = E / (12 * EL(i) ^ 3)
   Ki = E / EL(i) * (Bw1(i) * (H1(i) - 0.5 * q(i) + p(i) / 3) - v(i) * (0.5 * H1(i) - q(i) / 3 + p(i) / 4) + w(i) * (H1(i) / 3 - q(i) / 4 + p(i) /
5))
   SMS(1, 1) = Ki * CX(i) ^ 2 + Kx * CY(i) ^ 2 * (Bw1(i) * (12 * ab(i) + 6 * bc(i) + 4.8 * cd(i) + 4.2 * de(i) + 26.4 * ef(i) / 7 + 24 *
fg(i) / 7 + 22 * gh(i) / 7) - v(i) * (6 * ab(i) + 4.8 * bc(i) + 4.2 * cd(i) + 26.4 * de(i) / 7 + 24 * ef(i) / 7 + 22 * fg(i) / 7 + 2.9 * gh(i)) +
w(i) * (4.8 * ab(i) + 4.2 * bc(i) + 26.4 * cd(i) / 7 + 24 * de(i) / 7 + 22 * ef(i) / 7 + 2.9 * fg(i) + 29.6 * gh(i) / 11))
   SMS(1, 2) = (Ki - (Kx * (Bw1(i) * (12 * ab(i) + 6 * bc(i) + 4.8 * cd(i) + 4.2 * de(i) + 26.4 * ef(i) / 7 + 24 * fg(i) / 7 + 22 * gh(i) /
7) - v(i) * (6 * ab(i) + 4.8 * bc(i) + 4.2 * cd(i) + 26.4 * de(i) / 7 + 24 * ef(i) / 7 + 22 * fg(i) / 7 + 2.9 * gh(i)) + w(i) * (4.8 * ab(i) +
4.2 * bc(i) + 26.4 * cd(i) / 7 + 24 * de(i) / 7 + 22 * ef(i) / 7 + 2.9 * fg(i) + 29.6 * gh(i) / 11)))) * CX(i) * CY(i)
   SMS(1, 3) = -CY(i) * Kx * EL(i) * (Bw1(i) * (6 * ab(i) + 2 * bc(i) + 1.4 * cd(i) + 1.2 * de(i) + 7.6 * ef(i) / 7 + fg(i) + 6.5 * gh(i) /
7) - v(i) * (2 * ab(i) + 1.4 * bc(i) + 1.2 * cd(i) + 7.6 * de(i) / 7 + ef(i) + 6.5 * fg(i) / 7 + 7.8 * gh(i) / 9) + w(i) * (1.4 * ab(i) + 1.2 *
bc(i) + 7.6 * cd(i) / 7 + de(i) + 6.5 * ef(i) / 7 + 7.8 * fg(i) / 9 + 80.4 * gh(i) / 99))
   SMS(1, 4) = -SMS(1, 1)
   SMS(1, 5) = -SMS(1, 2)
   SMS(1, 6) = -CY(i) * Kx * EL(i) * (Bw1(i) * (6 * ab(i) + 4 * bc(i) + 3.4 * cd(i) + 3 * de(i) + 18.8 * ef(i) / 7 + 17 * fg(i) / 7 + 15.5
* gh(i) / 7) - v(i) * (4 * ab(i) + 3.4 * bc(i) + 3 * cd(i) + 18.8 * de(i) / 7 + 17 * ef(i) / 7 + 15.5 * fg(i) / 7 + 18.3 * gh(i) / 9) + w(i) *
(3.4 * ab(i) + 3 * bc(i) + 18.8 * cd(i) / 7 + 17 * de(i) / 7 + 15.5 * ef(i) / 7 + 18.3 * fg(i) / 9 + 186 * gh(i) / 99))
   SMS(2, 2) = Ki * CY(i) ^ 2 + Kx * CX(i) ^ 2 * (Bw1(i) * (12 * ab(i) + 6 * bc(i) + 4.8 * cd(i) + 4.2 * de(i) + 26.4 * ef(i) / 7 + 24 *
fg(i) / 7 + 22 * gh(i) / 7) - v(i) * (6 * ab(i) + 4.8 * bc(i) + 4.2 * cd(i) + 26.4 * de(i) / 7 + 24 * ef(i) / 7 + 22 * fg(i) / 7 + 2.9 * gh(i)) +
w(i) * (4.8 * ab(i) + 4.2 * bc(i) + 26.4 * cd(i) / 7 + 24 * de(i) / 7 + 22 * ef(i) / 7 + 2.9 * fg(i) + 29.6 * gh(i) / 11))
   SMS(2, 3) = CX(i) * Kx * EL(i) * (Bw1(i) * (6 * ab(i) + 2 * bc(i) + 1.4 * cd(i) + 1.2 * de(i) + 7.6 * ef(i) / 7 + fg(i) + 6.5 * gh(i) /
7) - v(i) * (2 * ab(i) + 1.4 * bc(i) + 1.2 * cd(i) + 7.6 * de(i) / 7 + ef(i) + 6.5 * fg(i) / 7 + 7.8 * gh(i) / 9) + w(i) * (1.4 * ab(i) + 1.2 *
bc(i) + 7.6 * cd(i) / 7 + de(i) + 6.5 * ef(i) / 7 + 7.8 * fg(i) / 9 + 80.4 * gh(i) / 99))
   SMS(2, 4) = -SMS(1, 2)
   SMS(2, 5) = -SMS(2, 2)
   SMS(2, 6) = CX(i) * Kx * EL(i) * (Bw1(i) * (6 * ab(i) + 4 * bc(i) + 3.4 * cd(i) + 3 * de(i) + 18.8 * ef(i) / 7 + 17 * fg(i) / 7 + 15.5
* gh(i) / 7) - v(i) * (4 * ab(i) + 3.4 * bc(i) + 3 * cd(i) + 18.8 * de(i) / 7 + 17 * ef(i) / 7 + 15.5 * fg(i) / 7 + 18.3 * gh(i) / 9) + w(i) *
(3.4 * ab(i) + 3 * bc(i) + 18.8 * cd(i) / 7 + 17 * de(i) / 7 + 15.5 * ef(i) / 7 + 18.3 * fg(i) / 9 + 186 * gh(i) / 99))
   SMS(3, 3) = Kx * EL(i) ^ 2 * (Bw1(i) * (4 * ab(i) + bc(i) + 1.6 * cd(i) / 3 + 0.4 * de(i) + 2.4 * ef(i) / 7 + 13 * fg(i) / 42 + 2 * gh(i)
/ 7) - v(i) * (ab(i) + 1.6 * bc(i) / 3 + 0.4 * cd(i) + 2.4 * de(i) / 7 + 13 * ef(i) / 42 + 2 * fg(i) / 7 + 2.4 * gh(i) / 9) + w(i) * (1.6 * ab(i) /
3 + 0.4 * bc(i) + 2.4 * cd(i) / 7 + 13 * de(i) / 42 + 2 * ef(i) / 7 + 2.4 * fg(i) / 9 + 24.8 * gh(i) / 99))
   SMS(3, 4) = -SMS(1, 3)
   SMS(3, 5) = -SMS(2, 3)
   SMS(3, 6) = Kx * EL(i) ^ 2 * (Bw1(i) * (2 * ab(i) + bc(i) + 2.6 * cd(i) / 3 + 0.8 * de(i) + 25 * ef(i) / 35 + 29 * fg(i) / 42 + 9 *
gh(i) / 14) - v(i) * (ab(i) + 2.6 * bc(i) / 3 + 0.8 * cd(i) + 26 * de(i) / 35 + 29 * ef(i) / 42 + 9 * fg(i) / 14 + 0.6 * gh(i)) + w(i) * (2.6 *
ab(i) / 3 + 0.8 * bc(i) + 26 * cd(i) / 35 + 29 * de(i) / 42 + 9 * ef(i) / 14 + 0.6 * fg(i) + 55.6 * gh(i) / 99))
   SMS(4, 4) = SMS(1, 1)
   SMS(4, 5) = SMS(1, 2)
   SMS(4, 6) = -SMS(1, 6)
   SMS(5, 5) = SMS(2, 2)
   SMS(5, 6) = -SMS(2, 6)
   SMS(6, 6) = Kx * EL(i) ^ 2 * (Bw1(i) * (4 * ab(i) + 3 * bc(i) + 7.6 * cd(i) / 3 + 2.2 * de(i) + 13.6 * ef(i) / 7 + 73 * fg(i) / 42 + 11
* gh(i) / 7) - v(i) * (3 * ab(i) + 7.6 * bc(i) / 3 + 2.2 * cd(i) + 13.6 * de(i) / 7 + 73 * ef(i) / 42 + 11 * fg(i) / 7 + 12.9 * gh(i) / 9) + w(i)
* (7.6 * ab(i) / 3 + 2.2 * bc(i) + 13.6 * cd(i) / 7 + 73 * de(i) / 42 + 11 * ef(i) / 7 + 12.9 * fg(i) / 9 + 130.4 * gh(i) / 99))
   IM(1) = 3 * JJ(i) - 2
   IM(2) = 3 * JJ(i) - 1
   IM(3) = 3 * JJ(i)
   IM(4) = 3 * JK(i) - 2
   IM(5) = 3 * JK(i) - 1
   IM(6) = 3 * JK(i)

   For j = 1 To MD
```

```
        I1 = IM(j)
        If JRL(I1) > 0 Then
          GoTo 2
        Else
          For K = j To MD
            I2 = IM(K)
            If JRL(I2) > 0 Then
              GoTo 2
            Else
              IR = ID(I1)
              IC = ID(I2)
              If IR < IC Then
                GoTo 1
              Else
                ITEM = IR
                IR = IC
                IC = ITEM
1:               IC = IC - IR + 1
                SFF(IR, IC) = SFF(IR, IC) + SMS(j, K)
              End If
            End If
2:        Next K
        End If
        Next j
      Next i
      End Sub
      Sub banfac(N As Integer, NB As Integer, a() As Single, c As Single)
      For j = 2 To N
        J1 = j - 1
        J2 = j - NB + 1
        If J2 < 1 Then J2 = 1
        If J1 = 1 Then GoTo 3
        For i = 2 To J1
          I1 = i - 1
          If I1 < J2 Then GoTo 2
          SUM = a(i, j - i + 1)
          For K = J2 To I1
            SUM = SUM - a(K, i - K + 1) * a(K, j - K + 1)
          Next K
          a(i, j - i + 1) = SUM
2:      Next i
3:      SUM = a(j, 1)
        For K = J2 To J1
          TEMP = a(K, j - K + 1) / a(K, 1)
          SUM = SUM - TEMP * a(K, j - K + 1)
          a(K, j - K + 1) = TEMP
        Next K
        a(j, 1) = SUM
      Next j
      End Sub
      Sub ldata()
      ReDim AJ(3 * M + 3)
      ReDim AE(ND)
      NLJ = (Form2.Text1.text)
      nlm = (Form2.Text2.text)
      If NLJ = 0 Then GoTo 2
      For j = 1 To NLJ
        K = Form2.MSFlexGrid1.TextMatrix(j, 0)
        xa = Form2.MSFlexGrid1.TextMatrix(j, 1)
        xb = Form2.MSFlexGrid1.TextMatrix(j, 2)
        xc = Form2.MSFlexGrid1.TextMatrix(j, 3)
        AJ(3 * K - 2) = xa
        AJ(3 * K - 1) = xb
        AJ(3 * K) = xc
      Next j
      2: If nlm = 0 Then GoTo 4
      ReDim LML(M)
      ReDim q1(i), q2(i), q3(i), r1(i), r2(i), r3(i), r4(i), r5(i), r6(i)
      ReDim q1(i), q2(i), r1(i), r4(i), r2(i), r3(i)
      ReDim q3(i), r5(i), r6(i)
      ReDim ba1(i), ha1(i), inertiaa1(i), intega1(i), valuea1(i)
```

```
ReDim bb1(i), hb1(i), inertiab1(i), integb1(i), valueb1(i)
ReDim bc1(i), hc1(i), inertiac1(i), integc1(i), valuec1(i)
ReDim bd1(i), hd1(i), inertiad1(i), integd1(i), valued1(i)
ReDim be1(i), he1(i), inertiae1(i), intege1(i), valuee1(i)
ReDim bf1(i), hf1(i), inertiaf1(i), integf1(i), valuef1(i)
ReDim bg1(i), hg1(i), inertiag1(i), integg1(i), valueg1(i)
ReDim bh1(i), hh1(i), inertiah1(i), integh1(i), valueh1(i)
ReDim bi1(i), hi1(i), inertiai1(i), integi1(i), valuei1(i)
ReDim bj1(i), hj1(i), inertiaj1(i), integj1(i), valuej1(i)
ReDim bk1(i), hk1(i), inertiak1(i), integk1(i), valuek1(i)
ReDim bl1(i), hl1(i), inertial1(i), integl1(i), valuel1(i)
ReDim eone(i), fone(i)
ReDim momenttrap1(i), reactiontrap1(i)
ReDim ba2(i), ha2(i), inertiaa2(i), intega2(i), valuea2(i)
ReDim bb2(i), hb2(i), inertiab2(i), integb2(i), valueb2(i)
ReDim bc2(i), hc2(i), inertiac2(i), integc2(i), valuec2(i)
ReDim bd2(i), hd2(i), inertiad2(i), integd2(i), valued2(i)
ReDim be2(i), he2(i), inertiae2(i), intege2(i), valuee2(i)
ReDim bf2(i), hf2(i), inertiaf2(i), integf2(i), valuef2(i)
ReDim bg2(i), hg2(i), inertiag2(i), integg2(i), valueg2(i)
ReDim bh2(i), hh2(i), inertiah2(i), integh2(i), valueh2(i)
ReDim bi2(i), hi2(i), inertiai2(i), integi2(i), valuei2(i)
ReDim bj2(i), hj2(i), inertiaj2(i), integj2(i), valuej2(i)
ReDim bk2(i), hk2(i), inertiak2(i), integk2(i), valuek2(i)
ReDim bl2(i), hl2(i), inertial2(i), integl2(i), valuel2(i)
ReDim etwo(i), ftwo(i)
ReDim momenttrap2(i), reactiontrap2(i)
ReDim bap1(i), hap1(i), inertiaap1(i), integap1(i), valueap1(i)
ReDim bbp1(i), hbp1(i), inertiabp1(i), integbp1(i), valuebp1(i)
ReDim bcp1(i), hcp1(i), inertiacp1(i), integcp1(i), valuecp1(i)
ReDim bdp1(i), hdp1(i), inertiadp1(i), integdp1(i), valuedp1(i)
ReDim bep1(i), hep1(i), inertiaep1(i), integep1(i), valueep1(i)
ReDim bfp1(i), hfp1(i), inertiafp1(i), integfp1(i), valuefp1(i)
ReDim bgp1(i), hgp1(i), inertiagp1(i), integgp1(i), valuegp1(i)
ReDim bhp1(i), hhp1(i), inertiahp1(i), integhp1(i), valuehp1(i)
ReDim bkp1(i), hkp1(i), inertiakp1(i), integkp1(i), valuekp1(i)
ReDim momentpoint1(i), reactionpoint1(i)
ReDim momentpoint1total(i), reactionpoint1total(i)
ReDim bap2(i), hap2(i), inertiaap2(i), integap2(i), valueap2(i)
ReDim bbp2(i), hbp2(i), inertiabp2(i), integbp2(i), valuebp2(i)
ReDim bcp2(i), hcp2(i), inertiacp2(i), integcp2(i), valuecp2(i)
ReDim bdp2(i), hdp2(i), inertiadp2(i), integdp2(i), valuedp2(i)
ReDim bep2(i), hep2(i), inertiaep2(i), integep2(i), valueep2(i)
ReDim bfp2(i), hfp2(i), inertiafp2(i), integfp2(i), valuefp2(i)
ReDim bgp2(i), hgp2(i), inertiagp2(i), integgp2(i), valuegp2(i)
ReDim bhp2(i), hhp2(i), inertiahp2(i), integhp2(i), valuehp2(i)
ReDim bkp2(i), hkp2(i), inertiakp2(i), integkp2(i), valuekp2(i)
ReDim momentpoint2(i), reactionpoint2(i)
ReDim momentpoint2total(i), reactionpoint2total(i)
For j = 1 To nlm
  'Input #nomorfile, i, xa
  i = Form2.MSFlexGrid2.TextMatrix(j, 0)
  q1(i) = Form2.MSFlexGrid2.TextMatrix(j, 1)
  r1(i) = Form2.MSFlexGrid2.TextMatrix(j, 2)  'a
  q2(i) = Form2.MSFlexGrid2.TextMatrix(j, 3)
  r2(i) = Form2.MSFlexGrid2.TextMatrix(j, 4)  'c
  r4(i) = EL(i) - r1(i)   'b = 1 - a
  r3(i) = EL(i) - r2(i)   'd = 1 - c
  If q1(i) = 0 And q2(i) = 0 Then
    momenttrap1(i) = 0
    reactiontrap1(i) = 0
    momenttrap2(i) = 0
    reactiontrap2(i) = 0
    GoTo continue
  End If
  For gauss = 1 To 15
    Select Case gauss
      Case 1
        xgauss = -0.987992518020485
        cgauss = 3.07532419961173E-02
      Case 2
```

```
            xgauss = -0.937273392400706
            cgauss = 7.03660474881081E-02
          Case 3
            xgauss = -0.848206583410427
            cgauss = 0.107159220467172
          Case 4
            xgauss = -0.72441773136017
            cgauss = 0.139570677926154
          Case 5
            xgauss = -0.570972172608539
            cgauss = 0.166269205816994
          Case 6
            xgauss = -0.394151347077563
            cgauss = 0.186161000015562
          Case 7
            xgauss = -0.201194093997435
            cgauss = 0.198431485327112
          Case 8
            xgauss = 0
            cgauss = 0.202578241925561
          Case 9
            xgauss = 0.201194093997435
            cgauss = 0.198431485327112
          Case 10
            xgauss = 0.394151347077563
            cgauss = 0.186161000015562
          Case 11
            xgauss = 0.570972172608539
            cgauss = 0.166269205816994
          Case 12
            xgauss = 0.72441773136017
            cgauss = 0.139570677926154
          Case 13
            xgauss = 0.848206583410427
            cgauss = 0.107159220467172
          Case 14
            xgauss = 0.937273392400706
            cgauss = 7.03660474881081E-02
          Case 15
            xgauss = 0.987992518020485
            cgauss = 3.07532419961173E-02
        End Select
        ba1(i) = Bw1(i) + (-3 * Bw1(i) - Bw2(i) + 4 * Bw3(i)) * (EL(i) + EL(i) * xgauss) / (2 * EL(i)) + 2 * (Bw1(i) + Bw2(i) - 2 *
Bw3(i)) * ((EL(i) + EL(i) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        ha1(i) = H1(i) + (-3 * H1(i) - H2(i) + 4 * H3(i)) * (EL(i) + EL(i) * xgauss) / (2 * EL(i)) + 2 * (H1(i) + H2(i) - 2 * H3(i)) *
((EL(i) + EL(i) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        inertiaa1(i) = ba1(i) * (ha1(i) ^ 3) / 12
        intega1(i) = cgauss * 0.5 * EL(i) / inertiaa1(i)
        valuea1(i) = valuea1(i) + intega1(i)
        ba2(i) = Bw2(i) + (-3 * Bw2(i) - Bw1(i) + 4 * Bw3(i)) * (EL(i) + EL(i) * xgauss) / (2 * EL(i)) + 2 * (Bw2(i) + Bw1(i) - 2 *
Bw3(i)) * ((EL(i) + EL(i) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        ha2(i) = H2(i) + (-3 * H2(i) - H1(i) + 4 * H3(i)) * (EL(i) + EL(i) * xgauss) / (2 * EL(i)) + 2 * (H2(i) + H1(i) - 2 * H3(i)) *
((EL(i) + EL(i) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        inertiaa2(i) = ba2(i) * (ha2(i) ^ 3) / 12
        intega2(i) = cgauss * 0.5 * EL(i) / inertiaa2(i)
        valuea2(i) = valuea2(i) + intega2(i)
        integb1(i) = cgauss * 0.5 * EL(i) * (0.5 * (EL(i) + EL(i) * xgauss)) / inertiaa1(i)
        valueb1(i) = valueb1(i) + integb1(i)
        integb2(i) = cgauss * 0.5 * EL(i) * (0.5 * (EL(i) + EL(i) * xgauss)) / inertiaa2(i)
        valueb2(i) = valueb2(i) + integb2(i)
        valuec1(i) = valueb1(i)
        valuec2(i) = valueb2(i)
        integd1(i) = cgauss * 0.5 * EL(i) * ((0.5 * (EL(i) + EL(i) * xgauss)) ^ 2) / inertiaa1(i)
        valued1(i) = valued1(i) + integd1(i)
        integd2(i) = cgauss * 0.5 * EL(i) * ((0.5 * (EL(i) + EL(i) * xgauss)) ^ 2) / inertiaa2(i)
        valued2(i) = valued2(i) + integd2(i)
        be1(i) = Bw1(i) + (-3 * Bw1(i) - Bw2(i) + 4 * Bw3(i)) * ((r2(i) + r1(i)) + (r2(i) - r1(i)) * xgauss) / (2 * EL(i)) + 2 * (Bw1(i)
+ Bw2(i) - 2 * Bw3(i)) * (((r2(i) + r1(i)) + (r2(i) - r1(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        he1(i) = H1(i) + (-3 * H1(i) - H2(i) + 4 * H3(i)) * ((r2(i) + r1(i)) + (r2(i) - r1(i)) * xgauss) / (2 * EL(i)) + 2 * (H1(i) + H2(i) -
2 * H3(i)) * (((r2(i) + r1(i)) + (r2(i) - r1(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        inertiae1(i) = be1(i) * (he1(i) ^ 3) / 12
```

```
        intege1(i) = cgauss * (((0.5 * ((r2(i) + r1(i)) + (r2(i) - r1(i)) * xgauss)) - r1(i)) ^ 2) * (0.5 * (r2(i) - r1(i))) / inertiae1(i)
        valuee1(i) = valuee1(i) + intege1(i)
        be2(i) = Bw2(i) + (-3 * Bw2(i) - Bw1(i) + 4 * Bw3(i)) * ((r4(i) + r3(i)) + (r4(i) - r3(i)) * xgauss) / (2 * EL(i)) + 2 * (Bw2(i)
+ Bw1(i) - 2 * Bw3(i)) * (((r4(i) + r3(i)) + (r4(i) - r3(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        he2(i) = H2(i) + (-3 * H2(i) - H1(i) + 4 * H3(i)) * ((r4(i) + r3(i)) + (r4(i) - r3(i)) * xgauss) / (2 * EL(i)) + 2 * (H2(i) + H1(i) -
2 * H3(i)) * (((r4(i) + r3(i)) + (r4(i) - r3(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        inertiae2(i) = be2(i) * (he2(i) ^ 3) / 12
        intege2(i) = cgauss * (((0.5 * ((r4(i) + r3(i)) + (r4(i) - r3(i)) * xgauss)) - r3(i)) ^ 2) * (0.5 * (r4(i) - r3(i))) / inertiae2(i)
        valuee2(i) = valuee2(i) + intege2(i)
        integf1(i) = cgauss * (((0.5 * ((r2(i) + r1(i)) + (r2(i) - r1(i)) * xgauss)) - r1(i)) ^ 3) * (0.5 * (r2(i) - r1(i))) / inertiae1(i)
        valuef1(i) = valuef1(i) + integf1(i)
        integf2(i) = cgauss * (((0.5 * ((r4(i) + r3(i)) + (r4(i) - r3(i)) * xgauss)) - r3(i)) ^ 3) * (0.5 * (r4(i) - r3(i))) / inertiae2(i)
        valuef2(i) = valuef2(i) + integf2(i)
        bg1(i) = Bw1(i) + (-3 * Bw1(i) - Bw2(i) + 4 * Bw3(i)) * ((EL(i) + r2(i)) + (EL(i) - r2(i)) * xgauss) / (2 * EL(i)) + 2 *
(Bw1(i) + Bw2(i) - 2 * Bw3(i)) * (((EL(i) + r2(i)) + (EL(i) - r2(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        hg1(i) = H1(i) + (-3 * H1(i) - H2(i) + 4 * H3(i)) * ((EL(i) + r2(i)) + (EL(i) - r2(i)) * xgauss) / (2 * EL(i)) + 2 * (H1(i) +
H2(i) - 2 * H3(i)) * (((EL(i) + r2(i)) + (EL(i) - r2(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        inertiag1(i) = bg1(i) * (hg1(i) ^ 3) / 12
        integg1(i) = cgauss * 0.5 * ((EL(i) + r2(i)) + (EL(i) - r2(i)) * xgauss) * 0.5 * (EL(i) - r2(i)) / inertiag1(i)
        valueg1(i) = valueg1(i) + integg1(i)
        bg2(i) = Bw2(i) + (-3 * Bw2(i) - Bw1(i) + 4 * Bw3(i)) * ((EL(i) + r4(i)) + (EL(i) - r4(i)) * xgauss) / (2 * EL(i)) + 2 *
(Bw2(i) + Bw1(i) - 2 * Bw3(i)) * (((EL(i) + r4(i)) + (EL(i) - r4(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        hg2(i) = H2(i) + (-3 * H2(i) - H1(i) + 4 * H3(i)) * ((EL(i) + r4(i)) + (EL(i) - r4(i)) * xgauss) / (2 * EL(i)) + 2 * (H2(i) +
H1(i) - 2 * H3(i)) * (((EL(i) + r4(i)) + (EL(i) - r4(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        inertiag2(i) = bg2(i) * (hg2(i) ^ 3) / 12
        integg2(i) = cgauss * 0.5 * ((EL(i) + r4(i)) + (EL(i) - r4(i)) * xgauss) * 0.5 * (EL(i) - r4(i)) / inertiag2(i)
        valueg2(i) = valueg2(i) + integg2(i)
        integh1(i) = cgauss * 0.5 * (EL(i) - r2(i)) / inertiag1(i)
        valueh1(i) = valueh1(i) + integh1(i)
        integh2(i) = cgauss * 0.5 * (EL(i) - r4(i)) / inertiag2(i)
        valueh2(i) = valueh2(i) + integh2(i)
        integi1(i) = cgauss * 0.5 * ((r2(i) + r1(i)) + (r2(i) - r1(i)) * xgauss) * (((0.5 * ((r2(i) + r1(i)) + (r2(i) - r1(i)) * xgauss)) - r1(i))
^ 2) * (0.5 * (r2(i) - r1(i))) / inertiae1(i)
        valuei1(i) = valuei1(i) + integi1(i)
        integi2(i) = cgauss * 0.5 * ((r4(i) + r3(i)) + (r4(i) - r3(i)) * xgauss) * (((0.5 * ((r4(i) + r3(i)) + (r4(i) - r3(i)) * xgauss)) - r3(i))
^ 2) * (0.5 * (r4(i) - r3(i))) / inertiae2(i)
        valuei2(i) = valuei2(i) + integi2(i)
        integj1(i) = cgauss * 0.5 * ((r2(i) + r1(i)) + (r2(i) - r1(i)) * xgauss) * (((0.5 * ((r2(i) + r1(i)) + (r2(i) - r1(i)) * xgauss)) - r1(i))
^ 3) * (0.5 * (r2(i) - r1(i))) / inertiae1(i)
        valuej1(i) = valuej1(i) + integj1(i)
        integj2(i) = cgauss * 0.5 * ((r4(i) + r3(i)) + (r4(i) - r3(i)) * xgauss) * (((0.5 * ((r4(i) + r3(i)) + (r4(i) - r3(i)) * xgauss)) - r3(i))
^ 3) * (0.5 * (r4(i) - r3(i))) / inertiae2(i)
        valuej2(i) = valuej2(i) + integj2(i)
        bk1(i) = Bw1(i) + (-3 * Bw1(i) - Bw2(i) + 4 * Bw3(i)) * ((EL(i) + r2(i)) + (EL(i) - r2(i)) * xgauss) / (2 * EL(i)) + 2 *
(Bw1(i) + Bw2(i) - 2 * Bw3(i)) * (((EL(i) + r2(i)) + (EL(i) - r2(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        hk1(i) = H1(i) + (-3 * H1(i) - H2(i) + 4 * H3(i)) * ((EL(i) + r2(i)) + (EL(i) - r2(i)) * xgauss) / (2 * EL(i)) + 2 * (H1(i) +
H2(i) - 2 * H3(i)) * (((EL(i) + r2(i)) + (EL(i) - r2(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        inertiak1(i) = bk1(i) * (hk1(i) ^ 3) / 12
        integk1(i) = cgauss * ((0.5 * ((EL(i) + r2(i)) + (EL(i) - r2(i)) * xgauss)) ^ 2) * 0.5 * (EL(i) - r2(i)) / inertiag1(i)
        valuek1(i) = valuek1(i) + integk1(i)
        bk2(i) = Bw2(i) + (-3 * Bw2(i) - Bw1(i) + 4 * Bw3(i)) * ((EL(i) + r4(i)) + (EL(i) - r4(i)) * xgauss) / (2 * EL(i)) + 2 *
(Bw2(i) + Bw1(i) - 2 * Bw3(i)) * (((EL(i) + r4(i)) + (EL(i) - r4(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        hk2(i) = H2(i) + (-3 * H2(i) - H1(i) + 4 * H3(i)) * ((EL(i) + r4(i)) + (EL(i) - r4(i)) * xgauss) / (2 * EL(i)) + 2 * (H2(i) +
H1(i) - 2 * H3(i)) * (((EL(i) + r4(i)) + (EL(i) - r4(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
        inertiak2(i) = bk2(i) * (hk2(i) ^ 3) / 12
        integk2(i) = cgauss * ((0.5 * ((EL(i) + r4(i)) + (EL(i) - r4(i)) * xgauss)) ^ 2) * 0.5 * (EL(i) - r4(i)) / inertiag2(i)
        valuek2(i) = valuek2(i) + integk2(i)
        valuel1(i) = valueg1(i)
        valuel2(i) = valueg2(i)
    Next gauss
    If q1(i) < q2(i) Then
        eone(i) = 0.5 * q1(i) * valuee1(i) + ((q2(i) - q1(i)) * valuef1(i) / (r2(i) - r1(i))) / 6 + q1(i) * (r2(i) - r1(i)) * valueg1(i) - 0.5 *
(r1(i) + r2(i)) * (r2(i) - r1(i)) * q1(i) * valueh1(i) + 0.5 * (q2(i) - q1(i)) * (r2(i) - r1(i)) * valueg1(i) - ((q2(i) - q1(i)) * (r2(i) - r1(i)) *
(r1(i) + 2 * r2(i))) / 6 * valueh1(i)
        fone(i) = 0.5 * q1(i) * valuei1(i) + ((q2(i) - q1(i)) * valuej1(i) / (r2(i) - r1(i))) / 6 + q1(i) * (r2(i) - r1(i)) * valuek1(i) - 0.5 *
(r1(i) + r2(i)) * (r2(i) - r1(i)) * q1(i) * valuel1(i) + 0.5 * (q2(i) - q1(i)) * (r2(i) - r1(i)) * valuek1(i) - ((q2(i) - q1(i)) * (r2(i) - r1(i)) *
(r1(i) + 2 * r2(i))) / 6 * valuel1(i)
        momenttrap1(i) = -1 * (-valueb1(i) * fone(i) + valued1(i) * eone(i)) / (valuea1(i) * valued1(i) - valueb1(i) * valuec1(i))
        reactiontrap1(i) = (valuea1(i) * fone(i) - valuec1(i) * eone(i)) / (valuea1(i) * valued1(i) - valueb1(i) * valuec1(i))
```

```
        etwo(i) = 0.5 * q2(i) * valuee2(i) + ((q1(i) - q2(i)) * valuef2(i) / (r4(i) - r3(i))) / 6 + q1(i) * (r4(i) - r3(i)) * valueg2(i) - 0.5 *
(r3(i) + r4(i)) * (r4(i) - r3(i)) * q1(i) * valueh2(i) + 0.5 * (q2(i) - q1(i)) * (r4(i) - r3(i)) * valueg2(i) - ((q2(i) - q1(i)) * (r4(i) - r3(i)) *
(r4(i) + 2 * r3(i))) / 6 * valueh2(i)
        ftwo(i) = 0.5 * q2(i) * valuei2(i) + ((q1(i) - q2(i)) * valuej2(i) / (r4(i) - r3(i))) / 6 + q1(i) * (r4(i) - r3(i)) * valuek2(i) - 0.5 *
(r3(i) + r4(i)) * (r4(i) - r3(i)) * q1(i) * valuel2(i) + 0.5 * (q2(i) - q1(i)) * (r4(i) - r3(i)) * valuek2(i) - ((q2(i) - q1(i)) * (r4(i) - r3(i)) *
(r4(i) + 2 * r3(i))) / 6 * valuel2(i)
        momenttrap2(i) = (-valueb2(i) * ftwo(i) + valued2(i) * etwo(i)) / (valuea2(i) * valued2(i) - valueb2(i) * valuec2(i))
        reactiontrap2(i) = (valuea2(i) * ftwo(i) - valuec2(i) * etwo(i)) / (valuea2(i) * valued2(i) - valueb2(i) * valuec2(i))
    Else
        eone(i) = 0.5 * q1(i) * valuee1(i) + ((q2(i) - q1(i)) * valuef1(i) / (r2(i) - r1(i))) / 6 + q2(i) * (r2(i) - r1(i)) * valueg1(i) - 0.5 *
(r1(i) + r2(i)) * (r2(i) - r1(i)) * q2(i) * valueh1(i) + 0.5 * (q1(i) - q2(i)) * (r2(i) - r1(i)) * valueg1(i) - ((q1(i) - q2(i)) * (r2(i) - r1(i)) *
(2 * r1(i) + r2(i))) / 6 * valueh1(i)
        fone(i) = 0.5 * q1(i) * valuei1(i) + ((q2(i) - q1(i)) * valuej1(i) / (r2(i) - r1(i))) / 6 + q2(i) * (r2(i) - r1(i)) * valuek1(i) - 0.5 *
(r1(i) + r2(i)) * (r2(i) - r1(i)) * q2(i) * valuel1(i) + 0.5 * (q1(i) - q2(i)) * (r2(i) - r1(i)) * valuek1(i) - ((q1(i) - q2(i)) * (r2(i) - r1(i)) *
(2 * r1(i) + r2(i))) / 6 * valuel1(i)
        momenttrap1(i) = -1 * (-valueb1(i) * fone(i) + valued1(i) * eone(i)) / (valuea1(i) * valued1(i) - valueb1(i) * valuec1(i))
        reactiontrap1(i) = (valuea1(i) * fone(i) - valuec1(i) * eone(i)) / (valuea1(i) * valued1(i) - valueb1(i) * valuec1(i))
        etwo(i) = 0.5 * q2(i) * valuee2(i) + ((q1(i) - q2(i)) * valuef2(i) / (r4(i) - r3(i))) / 6 + q2(i) * (r4(i) - r3(i)) * valueg2(i) - 0.5 *
(r3(i) + r4(i)) * (r4(i) - r3(i)) * q2(i) * valueh2(i) + 0.5 * (q1(i) - q2(i)) * (r4(i) - r3(i)) * valueg2(i) - ((q1(i) - q2(i)) * (r4(i) - r3(i)) *
(2 * r4(i) + r3(i))) / 6 * valueh2(i)
        ftwo(i) = 0.5 * q2(i) * valuei2(i) + ((q1(i) - q2(i)) * valuej2(i) / (r4(i) - r3(i))) / 6 + q2(i) * (r4(i) - r3(i)) * valuek2(i) - 0.5 *
(r3(i) + r4(i)) * (r4(i) - r3(i)) * q2(i) * valuel2(i) + 0.5 * (q1(i) - q2(i)) * (r4(i) - r3(i)) * valuek2(i) - ((q1(i) - q2(i)) * (r4(i) - r3(i)) *
(2 * r4(i) + r3(i))) / 6 * valuel2(i)
        momenttrap2(i) = (-valueb2(i) * ftwo(i) + valued2(i) * etwo(i)) / (valuea2(i) * valued2(i) - valueb2(i) * valuec2(i))
        reactiontrap2(i) = (valuea2(i) * ftwo(i) - valuec2(i) * etwo(i)) / (valuea2(i) * valued2(i) - valueb2(i) * valuec2(i))
    End If
continue:
For NPL = 0 To 3
    q3(i) = Form2.MSFlexGrid2.TextMatrix(j, (5 + 2 * NPL))
    r5(i) = Form2.MSFlexGrid2.TextMatrix(j, (6 + 2 * NPL))
    r6(i) = EL(i) - r5(i)
    If q3(i) = 0 Then GoTo continued
    valueap1(i) = 0
    valuebp1(i) = 0
    valuecp1(i) = 0
    valuedp1(i) = 0
    valueep1(i) = 0
    valuefp1(i) = 0
    valuegp1(i) = 0
    valuehp1(i) = 0
    valuekp1(i) = 0
    valueap2(i) = 0
    valuebp2(i) = 0
    valuecp2(i) = 0
    valuedp2(i) = 0
    valueep2(i) = 0
    valuefp2(i) = 0
    valuegp2(i) = 0
    valuehp2(i) = 0
    valuekp2(i) = 0
    For gauss = 1 To 15
        Select Case gauss
            Case 1
                xgauss = -0.987992518020485
                cgauss = 3.07532419961173E-02
            Case 2
                xgauss = -0.937273392400706
                cgauss = 7.03660474881081E-02
            Case 3
                xgauss = -0.848206583410427
                cgauss = 0.107159220467172
            Case 4
                xgauss = -0.72441773136017
                cgauss = 0.139570677926154
            Case 5
                xgauss = -0.570972172608539
                cgauss = 0.166269205816994
            Case 6
                xgauss = -0.394151347077563
                cgauss = 0.186161000015562
            Case 7
```

```
            xgauss = -0.201194093997435
            cgauss = 0.198431485327112
          Case 8
            xgauss = 0
            cgauss = 0.202578241925561
          Case 9
            xgauss = 0.201194093997435
            cgauss = 0.198431485327112
          Case 10
            xgauss = 0.394151347077563
            cgauss = 0.186161000015562
          Case 11
            xgauss = 0.570972172608539
            cgauss = 0.166269205816994
          Case 12
            xgauss = 0.72441773136017
            cgauss = 0.139570677926154
          Case 13
            xgauss = 0.848206583410427
            cgauss = 0.107159220467172
          Case 14
            xgauss = 0.937273392400706
            cgauss = 7.03660474881081E-02
          Case 15
            xgauss = 0.987992518020485
            cgauss = 3.07532419961173E-02
        End Select
          bap1(i) = Bw1(i) + (-3 * Bw1(i) - Bw2(i) + 4 * Bw3(i)) * (EL(i) + EL(i) * xgauss) / (2 * EL(i)) + 2 * (Bw1(i) + Bw2(i) - 2 *
Bw3(i)) * ((EL(i) + EL(i) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
          hap1(i) = H1(i) + (-3 * H1(i) - H2(i) + 4 * H3(i)) * (EL(i) + EL(i) * xgauss) / (2 * EL(i)) + 2 * (H1(i) + H2(i) - 2 * H3(i)) *
((EL(i) + EL(i) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
          inertiaap1(i) = bap1(i) * (hap1(i) ^ 3) / 12
          integap1(i) = cgauss * 0.5 * EL(i) / inertiaap1(i)
          valueap1(i) = valueap1(i) + integap1(i)
          bap2(i) = Bw2(i) + (-3 * Bw2(i) - Bw1(i) + 4 * Bw3(i)) * (EL(i) + EL(i) * xgauss) / (2 * EL(i)) + 2 * (Bw2(i) + Bw1(i) - 2 *
Bw3(i)) * ((EL(i) + EL(i) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
          hap2(i) = H2(i) + (-3 * H2(i) - H1(i) + 4 * H3(i)) * (EL(i) + EL(i) * xgauss) / (2 * EL(i)) + 2 * (H2(i) + H1(i) - 2 * H3(i)) *
((EL(i) + EL(i) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
          inertiaap2(i) = bap2(i) * (hap2(i) ^ 3) / 12
          integap2(i) = cgauss * 0.5 * EL(i) / inertiaap2(i)
          valueap2(i) = valueap2(i) + integap2(i)
          integbp1(i) = cgauss * 0.5 * EL(i) * (0.5 * (EL(i) + EL(i) * xgauss)) / inertiaap1(i)
          valuebp1(i) = valuebp1(i) + integbp1(i)
          integbp2(i) = cgauss * 0.5 * EL(i) * (0.5 * (EL(i) + EL(i) * xgauss)) / inertiaap2(i)
          valuebp2(i) = valuebp2(i) + integbp2(i)
          valuecp1(i) = valuebp1(i)
          valuecp2(i) = valuebp2(i)
          integdp1(i) = cgauss * 0.5 * EL(i) * ((0.5 * (EL(i) + EL(i) * xgauss)) ^ 2) / inertiaap1(i)
          valuedp1(i) = valuedp1(i) + integdp1(i)
          integdp2(i) = cgauss * 0.5 * EL(i) * ((0.5 * (EL(i) + EL(i) * xgauss)) ^ 2) / inertiaap2(i)
          valuedp2(i) = valuedp2(i) + integdp2(i)
          bgp1(i) = Bw1(i) + (-3 * Bw1(i) - Bw2(i) + 4 * Bw3(i)) * ((EL(i) + r5(i)) + (EL(i) - r5(i)) * xgauss) / (2 * EL(i)) + 2 *
(Bw1(i) + Bw2(i) - 2 * Bw3(i)) * (((EL(i) + r5(i)) + (EL(i) - r5(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
          hgp1(i) = H1(i) + (-3 * H1(i) - H2(i) + 4 * H3(i)) * ((EL(i) + r5(i)) + (EL(i) - r5(i)) * xgauss) / (2 * EL(i)) + 2 * (H1(i) +
H2(i) - 2 * H3(i)) * (((EL(i) + r5(i)) + (EL(i) - r5(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
          inertiagp1(i) = bgp1(i) * (hgp1(i) ^ 3) / 12
          integgp1(i) = cgauss * 0.5 * ((EL(i) + r5(i)) + (EL(i) - r5(i)) * xgauss) * 0.5 * (EL(i) - r5(i)) / inertiagp1(i)
          valuegp1(i) = valuegp1(i) + integgp1(i)
          bgp2(i) = Bw2(i) + (-3 * Bw2(i) - Bw1(i) + 4 * Bw3(i)) * ((EL(i) + r6(i)) + (EL(i) - r6(i)) * xgauss) / (2 * EL(i)) + 2 *
(Bw2(i) + Bw1(i) - 2 * Bw3(i)) * (((EL(i) + r6(i)) + (EL(i) - r6(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
          hgp2(i) = H2(i) + (-3 * H2(i) - H1(i) + 4 * H3(i)) * ((EL(i) + r6(i)) + (EL(i) - r6(i)) * xgauss) / (2 * EL(i)) + 2 * (H2(i) +
H1(i) - 2 * H3(i)) * (((EL(i) + r6(i)) + (EL(i) - r6(i)) * xgauss) ^ 2) / (4 * EL(i) * EL(i))
          inertiagp2(i) = bgp2(i) * (hgp2(i) ^ 3) / 12
          integgp2(i) = cgauss * 0.5 * ((EL(i) + r6(i)) + (EL(i) - r6(i)) * xgauss) * 0.5 * (EL(i) - r6(i)) / inertiagp2(i)
          valuegp2(i) = valuegp2(i) + integgp2(i)
          integhp1(i) = cgauss * 0.5 * (EL(i) - r5(i)) / inertiagp1(i)
          valuehp1(i) = valuehp1(i) + integhp1(i)
          integhp2(i) = cgauss * 0.5 * (EL(i) - r6(i)) / inertiagp2(i)
          valuehp2(i) = valuehp2(i) + integhp2(i)
          integkp1(i) = cgauss * ((0.5 * ((EL(i) + r5(i)) + (EL(i) - r5(i)) * xgauss)) ^ 2) * 0.5 * (EL(i) - r5(i)) / inertiagp1(i)
          valuekp1(i) = valuekp1(i) + integkp1(i)
```

```
        integkp2(i) = cgauss * ((0.5 * ((EL(i) + r6(i)) + (EL(i) - r6(i)) * xgauss)) ^ 2) * 0.5 * (EL(i) - r6(i)) / inertiagp2(i)
        valuekp2(i) = valuekp2(i) + integkp2(i)
    Next gauss
    valueep1(i) = q3(i) * valuegp1(i) - q3(i) * r5(i) * valuehp1(i)
    valuefp1(i) = q3(i) * valuekp1(i) - q3(i) * r5(i) * valuegp1(i)
    momentpoint1(i) = -1 * (-valuebp1(i) * valuefp1(i) + valuedp1(i) * valueep1(i)) / (valueap1(i) * valuedp1(i) - valuebp1(i) *
valuecp1(i))
    reactionpoint1(i) = (valueap1(i) * valuefp1(i) - valuecp1(i) * valueep1(i)) / (valueap1(i) * valuedp1(i) - valuebp1(i) * valuecp1(i))
    momentpoint1total(i) = momentpoint1total(i) + momentpoint1(i)
    reactionpoint1total(i) = reactionpoint1total(i) + reactionpoint1(i)
    valueep2(i) = q3(i) * valuegp2(i) - q3(i) * r6(i) * valuehp2(i)
    valuefp2(i) = q3(i) * valuekp2(i) - q3(i) * r6(i) * valuegp2(i)
    momentpoint2(i) = (-valuebp2(i) * valuefp2(i) + valuedp2(i) * valueep2(i)) / (valueap2(i) * valuedp2(i) - valuebp2(i) *
valuecp2(i))
    reactionpoint2(i) = (valueap2(i) * valuefp2(i) - valuecp2(i) * valueep2(i)) / (valueap2(i) * valuedp2(i) - valuebp2(i) * valuecp2(i))
    momentpoint2total(i) = momentpoint2total(i) + momentpoint2(i)
    reactionpoint2total(i) = reactionpoint2total(i) + reactionpoint2(i)
Next NPL
continued:
    AML(1, i) = 0
    AML(2, i) = reactiontrap1(i) + reactionpoint1total(i)
    AML(3, i) = momenttrap1(i) + momentpoint1total(i)
    AML(4, i) = 0
    AML(5, i) = reactiontrap2(i) + reactionpoint2total(i)
    AML(6, i) = momenttrap2(i) + momentpoint2total(i)
    LML(i) = 1
Next j
4: End Sub
Sub loads()
ReDim AC(ND)
If nlm = 0 Then GoTo 2
For i = 1 To M
    If LML(i) = 0 Then GoTo 1
    J1 = 3 * JJ(i) - 2
    J2 = 3 * JJ(i) - 1
    J3 = 3 * JJ(i)
    K1 = 3 * JK(i) - 2
    K2 = 3 * JK(i) - 1
    K3 = 3 * JK(i)
    AE(J1) = AE(J1) - CX(i) * AML(1, i) + CY(i) * AML(2, i)
    AE(J2) = AE(J2) - CY(i) * AML(1, i) - CX(i) * AML(2, i)
    AE(J3) = AE(J3) - AML(3, i)
    AE(K1) = AE(K1) - CX(i) * AML(4, i) + CY(i) * AML(5, i)
    AE(K2) = AE(K2) - CY(i) * AML(4, i) - CX(i) * AML(5, i)
    AE(K3) = AE(K3) - AML(6, i)
1: Next i
2: For j = 1 To ND
      JR = ID(j)
      AC(JR) = AJ(j) + AE(j)
  Next j
End Sub
Sub bansol(N As Integer, NB As Integer, u() As Single, b() As Single, z() As Single)
For i = 1 To N
    j = i - NB + 1
    If i <= NB Then j = 1
    SUM = b(i)
    K1 = i - 1
    If j > K1 Then GoTo 1
    For K = j To K1
      SUM = SUM - u(K, i - K + 1) * z(K)
    Next K
1:   z(i) = SUM
Next i
For i = 1 To N
    z(i) = z(i) / u(i, 1)
Next i
For I1 = 1 To N
    i = N - I1 + 1
    j = i + NB - 1
    If j > N Then j = N
```

```
    SUM = z(i)
    K2 = i + 1
    If K2 > j Then GoTo 2
    For K = K2 To j
      SUM = SUM - u(i, K - i + 1) * z(K)
    Next K
2:    z(i) = SUM
Next I1
End Sub
Sub resul()
For j = 1 To 4
    frmContoh2.MSFlexGrid4.ColWidth(0) = 1200
    frmContoh2.MSFlexGrid4.ColWidth(j) = 1500
Next j
With frmContoh2.MSFlexGrid4
    .Rows = nlm + 1
    .TextMatrix(0, 0) = "Frame ID"
    .TextMatrix(0, 1) = "Reaction i"
    .TextMatrix(0, 2) = "Moment i"
    .TextMatrix(0, 3) = "Reaction j"
    .TextMatrix(0, 4) = "Moment j"
End With
Print #1, "n-Prime 2004 Project"
Print #1, "[" & dlgfile.FileName & "]"
Print #1, "Unit: " & frmContoh1.CboUnit.text
Print #1,
Print #1, "Frame Element Fixed-End Forces"
Print #1, "Frame ID"; Tab(12); "R-i"; Tab(29); "M-i"; Tab(46); "R-j"; Tab(63); "M-j"
ReDim tampung(nlm)
For j = 1 To nlm
    tampung(j) = (Form2.MSFlexGrid2.TextMatrix(j, 0))
Next j
For j = 1 To nlm
    frmContoh2.MSFlexGrid4.TextMatrix(j, 0) = tampung(j)
    frmContoh2.MSFlexGrid4.TextMatrix(j, 1) = AML(2, tampung(j))
    frmContoh2.MSFlexGrid4.TextMatrix(j, 2) = AML(3, tampung(j))
    frmContoh2.MSFlexGrid4.TextMatrix(j, 3) = AML(5, tampung(j))
    frmContoh2.MSFlexGrid4.TextMatrix(j, 4) = AML(6, tampung(j))
    Print #1, tampung(j); Tab(12); AML(2, tampung(j)); Tab(29); AML(3, tampung(j)); Tab(46); AML(5, tampung(j)); Tab(63);
AML(6, tampung(j))
Next j
j = N + 1
For K = 1 To ND
    JE = ND - K + 1
    If JRL(JE) = 0 Then GoTo 1
    DJ(JE) = 0
    GoTo 2
1: j = j - 1
  DJ(JE) = DF(j)
2: Next K
For j = 1 To 3
    frmContoh2.MSFlexGrid1.ColWidth(0) = 1200
    frmContoh2.MSFlexGrid1.ColWidth(j) = 2000
Next j
    frmContoh2.MSFlexGrid1.Rows = NJ + 1
    frmContoh2.MSFlexGrid1.TextMatrix(0, 0) = "Joint ID"
    frmContoh2.MSFlexGrid1.TextMatrix(0, 1) = "Ux"
    frmContoh2.MSFlexGrid1.TextMatrix(0, 2) = "Uy"
    frmContoh2.MSFlexGrid1.TextMatrix(0, 3) = "Rz"
    Print #1,
    Print #1, "Joint Displacements"
    Print #1, "Joint ID"; Tab(12); "Ux"; Tab(29); "Uy"; Tab(46); "Rz"
For j = 1 To NJ
    frmContoh2.MSFlexGrid1.Row = j
    frmContoh2.MSFlexGrid1.Col = 0
    frmContoh2.MSFlexGrid1.text = j
    frmContoh2.MSFlexGrid1.Col = 1
    frmContoh2.MSFlexGrid1.text = DJ(3 * j - 2)
    frmContoh2.MSFlexGrid1.Col = 2
    frmContoh2.MSFlexGrid1.text = DJ(3 * j - 1)
    frmContoh2.MSFlexGrid1.Col = 3
```

```
    frmContoh2.MSFlexGrid1.text = DJ(3 * j)
    Print #1, j; Tab(12); DJ(3 * j - 2); Tab(29); DJ(3 * j - 1); Tab(46); DJ(3 * j)
Next j
For j = 1 To 6
    frmContoh2.MSFlexGrid2.ColWidth(0) = 1200
    frmContoh2.MSFlexGrid2.ColWidth(j) = 1500
Next j
With frmContoh2.MSFlexGrid2
    .Rows = M + 1
    .TextMatrix(0, 0) = "Frame ID"
    .TextMatrix(0, 1) = "Axial-i"
    .TextMatrix(0, 2) = "Shear-i"
    .TextMatrix(0, 3) = "Moment-i"
    .TextMatrix(0, 4) = "Axial-j"
    .TextMatrix(0, 5) = "Shear-j"
    .TextMatrix(0, 6) = "Moment-j"
End With
Print #1,
Print #1, "Frame Element Forces"
Print #1, "Frame ID"; Tab(12); "Axial-i"; Tab(29); "Shear-i"; Tab(46); "Moment-i"; Tab(63); "Axial-j"; Tab(80); "Shear-j";
Tab(97); "Moment-j"
For i = 1 To M
    J1 = 3 * JJ(i) - 2
    J2 = 3 * JJ(i) - 1
    J3 = 3 * JJ(i)
    K1 = 3 * JK(i) - 2
    K2 = 3 * JK(i) - 1
    K3 = 3 * JK(i)
    w(i) = 2 * (Bw1(i) + Bw2(i) - 2 * Bw3(i))
    v(i) = Bw1(i) - Bw2(i) + w(i)
    p(i) = 2 * (H1(i) + H2(i) - 2 * H3(i))
    q(i) = H1(i) - H2(i) + p(i)
    ab(i) = H1(i) ^ 3
    bc(i) = -3 * (H1(i) ^ 2) * q(i)
    cd(i) = 3 * H1(i) * q(i) ^ 2 + 3 * p(i) * H1(i) ^ 2
    de(i) = -(q(i) ^ 3 + 6 * p(i) * H1(i) * q(i))
    ef(i) = 3 * p(i) * q(i) ^ 2 + 3 * H1(i) * p(i) ^ 2
    fg(i) = -3 * q(i) * p(i) ^ 2
    gh(i) = p(i) ^ 3
    Kx = E / (12 * EL(i) ^ 3)
    SCM1 = E / EL(i) * (Bw1(i) * (H1(i) - 0.5 * q(i) + p(i) / 3) - v(i) * (0.5 * H1(i) - q(i) / 3 + p(i) / 4) + w(i) * (H1(i) / 3 - q(i) / 4 +
p(i) / 5))
    SCM22 = Kx * EL(i) ^ 2 * (Bw1(i) * (4 * ab(i) + bc(i) + 1.6 * cd(i) / 3 + 0.4 * de(i) + 2.4 * ef(i) / 7 + 13 * fg(i) / 42 + 2 * gh(i) /
7) - v(i) * (ab(i) + 1.6 * bc(i) / 3 + 0.4 * cd(i) + 2.4 * de(i) / 7 + 13 * ef(i) / 42 + 2 * fg(i) / 7 + 2.4 * gh(i) / 9) + w(i) * (1.6 * ab(i) / 3
+ 0.4 * bc(i) + 2.4 * cd(i) / 7 + 13 * de(i) / 42 + 2 * ef(i) / 7 + 2.4 * fg(i) / 9 + 24.8 * gh(i) / 99))
    SCM12 = Kx * EL(i) * (Bw1(i) * (6 * ab(i) + 2 * bc(i) + 1.4 * cd(i) + 1.2 * de(i) + 7.6 * ef(i) / 7 + fg(i) + 6.5 * gh(i) / 7) - v(i) *
(2 * ab(i) + 1.4 * bc(i) + 1.2 * cd(i) + 7.6 * de(i) / 7 + ef(i) + 6.5 * fg(i) / 7 + 7.8 * gh(i) / 9) + w(i) * (1.4 * ab(i) + 1.2 * bc(i) + 7.6 *
cd(i) / 7 + de(i) + 6.5 * ef(i) / 7 + 7.8 * fg(i) / 9 + 80.4 * gh(i) / 99))
    SCM11 = Kx * (Bw1(i) * (12 * ab(i) + 6 * bc(i) + 4.8 * cd(i) + 4.2 * de(i) + 26.4 * ef(i) / 7 + 24 * fg(i) / 7 + 22 * gh(i) / 7) - v(i)
* (6 * ab(i) + 4.8 * bc(i) + 4.2 * cd(i) + 26.4 * de(i) / 7 + 24 * ef(i) / 7 + 22 * fg(i) / 7 + 2.9 * gh(i)) + w(i) * (4.8 * ab(i) + 4.2 * bc(i)
+ 26.4 * cd(i) / 7 + 24 * de(i) / 7 + 22 * ef(i) / 7 + 2.9 * fg(i) + 29.6 * gh(i) / 11))
    SCM14 = Kx * EL(i) * (Bw1(i) * (6 * ab(i) + 4 * bc(i) + 3.4 * cd(i) + 3 * de(i) + 18.8 * ef(i) / 7 + 17 * fg(i) / 7 + 15.5 * gh(i) / 7)
- v(i) * (4 * ab(i) + 3.4 * bc(i) + 3 * cd(i) + 18.8 * de(i) / 7 + 17 * ef(i) / 7 + 15.5 * fg(i) / 7 + 18.3 * gh(i) / 9) + w(i) * (3.4 * ab(i) +
3 * bc(i) + 18.8 * cd(i) / 7 + 17 * de(i) / 7 + 15.5 * ef(i) / 7 + 18.3 * fg(i) / 9 + 186 * gh(i) / 99))
    SCM24 = Kx * EL(i) ^ 2 * (Bw1(i) * (2 * ab(i) + bc(i) + 2.6 * cd(i) / 3 + 0.8 * de(i) + 26 * ef(i) / 35 + 29 * fg(i) / 42 + 9 * gh(i) /
14) - v(i) * (ab(i) + 2.6 * bc(i) / 3 + 0.8 * cd(i) + 26 * de(i) / 35 + 29 * ef(i) / 42 + 9 * fg(i) / 14 + 0.6 * gh(i)) + w(i) * (2.6 * ab(i) / 3
+ 0.8 * bc(i) + 26 * cd(i) / 35 + 29 * de(i) / 42 + 9 * ef(i) / 14 + 0.6 * fg(i) + 55.6 * gh(i) / 99))
    SCM44 = Kx * EL(i) ^ 2 * (Bw1(i) * (4 * ab(i) + 3 * bc(i) + 7.6 * cd(i) / 3 + 2.2 * de(i) + 13.6 * ef(i) / 7 + 73 * fg(i) / 42 + 11 *
gh(i) / 7) - v(i) * (3 * ab(i) + 7.6 * bc(i) / 3 + 2.2 * cd(i) + 13.6 * de(i) / 7 + 73 * ef(i) / 42 + 11 * fg(i) / 7 + 12.9 * gh(i) / 9) + w(i) *
(7.6 * ab(i) / 3 + 2.2 * bc(i) + 13.6 * cd(i) / 7 + 73 * de(i) / 42 + 11 * ef(i) / 7 + 12.9 * fg(i) / 9 + 130.4 * gh(i) / 99))
    AMD(1) = SCM1 * ((DJ(J1) - DJ(K1)) * CX(i) + (DJ(J2) - DJ(K2)) * CY(i))
    AMD(2) = SCM11 * (-(DJ(J1) - DJ(K1)) * CY(i) + (DJ(J2) - DJ(K2)) * CX(i)) + SCM12 * DJ(J3) + SCM14 * DJ(K3)
    AMD(3) = -SCM12 * ((DJ(J1) - DJ(K1)) * CY(i) - (DJ(J2) - DJ(K2)) * CX(i)) + SCM22 * DJ(J3) + SCM24 * DJ(K3)
    AMD(4) = -AMD(1)
    AMD(5) = -AMD(2)
    AMD(6) = -SCM14 * ((DJ(J1) - DJ(K1)) * CY(i) - (DJ(J2) - DJ(K2)) * CX(i)) + SCM24 * DJ(J3) + SCM44 * DJ(K3)
    For j = 1 To MD
        AM(j) = AML(j, i) + AMD(j)
    Next j
    If JRL(J1) = 1 Then AR(J1) = AR(J1) + CX(i) * AMD(1) - CY(i) * AMD(2)
    If JRL(J2) = 1 Then AR(J2) = AR(J2) + CY(i) * AMD(1) + CX(i) * AMD(2)
```

```
    If JRL(J3) = 1 Then AR(J3) = AR(J3) + AMD(3)
    If JRL(K1) = 1 Then AR(K1) = AR(K1) + CX(i) * AMD(4) - CY(i) * AMD(5)
    If JRL(K2) = 1 Then AR(K2) = AR(K2) + CY(i) * AMD(4) + CX(i) * AMD(5)
    If JRL(K3) = 1 Then AR(K3) = AR(K3) + AMD(6)
    frmContoh2.MSFlexGrid2.TextMatrix(i, 0) = i
    frmContoh2.MSFlexGrid2.TextMatrix(i, 1) = -AM(1)
    frmContoh2.MSFlexGrid2.TextMatrix(i, 2) = AM(2)
    frmContoh2.MSFlexGrid2.TextMatrix(i, 3) = -AM(3)
    frmContoh2.MSFlexGrid2.TextMatrix(i, 4) = AM(4)
    frmContoh2.MSFlexGrid2.TextMatrix(i, 5) = -AM(5)
    frmContoh2.MSFlexGrid2.TextMatrix(i, 6) = AM(6)
    Print #1, i; Tab(12); -AM(1); Tab(29); AM(2); Tab(46); -AM(3); Tab(63); AM(4); Tab(80); -AM(5); Tab(97); AM(6)
Next i
For j = 1 To ND
    If JRL(j) = 0 Then GoTo 7
    AR(j) = AR(j) - AJ(j) - AE(j)
7: Next j
For j = 1 To 3
    frmContoh2.MSFlexGrid3.ColWidth(0) = 1000
    frmContoh2.MSFlexGrid3.ColWidth(j) = 1600
Next j
With frmContoh2.MSFlexGrid3
    .Rows = NRJ + 1
    .TextMatrix(0, 0) = "Joint ID"
    .TextMatrix(0, 1) = "Fx"
    .TextMatrix(0, 2) = "Fy"
    .TextMatrix(0, 3) = "Mz"
End With
Print #1,
Print #1, "Support Reactions"
Print #1, "Joint ID"; Tab(12); "Fx"; Tab(29); "Fy"; Tab(46); "Mz"
i = 1
For j = 1 To NJ
    J1 = 3 * j - 2
    J2 = 3 * j - 1
    J3 = 3 * j
    N1 = JRL(J1) + JRL(J2) + JRL(J3)
    If N1 = 0 Then GoTo 8
    frmContoh2.MSFlexGrid3.TextMatrix(i, 0) = j
    frmContoh2.MSFlexGrid3.TextMatrix(i, 1) = AR(J1)
    frmContoh2.MSFlexGrid3.TextMatrix(i, 2) = AR(J2)
    frmContoh2.MSFlexGrid3.TextMatrix(i, 3) = AR(J3)
    Print #1, j; Tab(12); AR(J1); Tab(29); AR(J2); Tab(46); AR(J3)
    i = i + 1
8:  Next j
End Sub
Private Sub mnuExit_Click()
End
End Sub
Private Sub mnuForces_Click()
FRGambar.Show
End Sub
Private Sub mnuFrame_Click()
frmGambarPortal.Show
End Sub
Private Sub mnuJoint_Click()
frmContoh1.Show
End Sub
Private Sub mnuLoad_Click()
Form2.Show
End Sub
Private Sub mnuSave_Click()
On Error GoTo debug4
dlgfile.Filter = "Text Document (*.txt)|*.txt"
dlgfile.DialogTitle = "Create Ouput File..."
dlgfile.DefaultExt = "txt"
dlgfile.ShowSave
Open dlgfile.FileName For Output As #1
resul
Close #1
frmContoh2.Caption = "n-Prime 2004 Project - [Output File: " & dlgfile.FileName & "]"
```

```
Exit Sub
debug4:
Close #1
frmContoh2.Show
End Sub
Private Sub mnuStart_Click()
frmContoh1.Show
frmContoh1.Text1.text = ""
frmContoh1.Text2.text = ""
frmContoh1.Text3.text = ""
frmContoh1.Text4.text = ""
frmContoh1.Text5.text = ""
frmContoh1.MSFlexGrid1.Clear
frmContoh1.MSFlexGrid3.Clear
frmContoh1.MSFlexGrid4.Clear
frmContoh1.MSFlexGrid6.Clear
Unload frmContoh2
Unload Form2
Form2.Text1 = ""
Form2.Text2 = ""
Form2.MSFlexGrid1.Clear
Form2.MSFlexGrid2.Clear
End Sub
```

Menggambar *Deformed-Frame*, NFD, SFD, dan BMD

```
Option Explicit
Dim pengali  As Integer
Dim pAx As Integer
Dim dispxscale As Long
Dim dispyscale As Long
Dim radius As Integer
Dim maxy As Long
Dim magnify As Integer
Dim NJ As Integer
Dim i As Single
Dim j As Single
Dim M As Integer
Private Sub DrawAll()
Dim axawal As Single
Dim axakhir As Single
Dim dispx1 As Single
Dim dispy1 As Single
Dim dispx2 As Single
Dim dispy2 As Single
'buat bentuk kurva
Dim x1 As Single
Dim x2 As Single
Dim y1 As Single
Dim y2 As Single
'*************
Dim axial As Single
Dim shear As Single
Dim moment As Single
Dim tinggi As Single
Dim jarak As Single
Dim posneg As Integer
Picture1.Cls
Picture2.Cls
Picture3.Cls
Picture4.Cls
Dim x, i, j, l, ii As Integer
Dim jumlah_batang As String
Dim p As Single, q As Single, jumlah As Single, E As Single, f As Single, g As Single, h As Single
Dim ED As Single, fD As Single, gD As Single, hD As Single
Dim a(25, 25), z(25, 25), disp(25, 25), K(4, 4), titik1, titik2, kumpul(4)
E = 0
f = 0
g = 0
h = 0
jumlah = (frmContoh1.Text2.text)
x = 0
```

```
For i = 1 To jumlah
  For j = 1 To 2
    a(i, j) = frmContoh1.MSFlexGrid1.TextMatrix(i, j)
    disp(i, j) = frmContoh2.MSFlexGrid1.TextMatrix(i, j)
  Next j
Next i
jumlah_batang = (frmContoh1.Text1.text)
jumlah = (frmContoh1.Text4.text)
For i = 1 To jumlah
  For j = 1 To 3
    K(i, j) = frmContoh1.MSFlexGrid3.TextMatrix(i, j)
  Next j
  kumpul(i) = frmContoh1.MSFlexGrid3.TextMatrix(i, 0)
  p = a(kumpul(i), 1) + 4
  q = a(kumpul(i), 2) + 4
  If K(i, 1) = 1 And K(i, 2) = 1 And K(i, 3) = 1 Then
    Garis Picture1, p * pengali, q * pengali, p * pengali, (q - 0.5) * pengali, &HFF0000
    Garis Picture1, (p - 0.75) * pengali, (q - 0.5) * pengali, (p + 0.75) * pengali, (q - 0.5) * pengali, &HFF0000
    Garis Picture2, p * pengali, q * pengali, p * pengali, (q - 0.5) * pengali, &HFF0000
    Garis Picture2, (p - 0.75) * pengali, (q - 0.5) * pengali, (p + 0.75) * pengali, (q - 0.5) * pengali, &HFF0000
    Garis Picture3, p * pengali, q * pengali, p * pengali, (q - 0.5) * pengali, &HFF0000
    Garis Picture3, (p - 0.75) * pengali, (q - 0.5) * pengali, (p + 0.75) * pengali, (q - 0.5) * pengali, &HFF0000
    Garis Picture4, p * pengali, q * pengali, p * pengali, (q - 0.5) * pengali, &HFF0000
    Garis Picture4, (p - 0.75) * pengali, (q - 0.5) * pengali, (p + 0.75) * pengali, (q - 0.5) * pengali, &HFF0000
  ElseIf K(i, 1) = 1 And K(i, 2) = 1 And K(i, 3) = 0 Then
    Garis Picture1, (p - 0.75) * pengali, (q - 0.75) * pengali, p * pengali, q * pengali, &HFF0000
    Garis Picture1, p * pengali, q * pengali, (p + 0.75) * pengali, (q - 0.75) * pengali, &HFF0000
    Garis Picture1, (p + 0.75) * pengali, (q - 0.75) * pengali, (p - 0.75) * pengali, (q - 0.75) * pengali, &HFF0000
    Garis Picture2, (p - 0.75) * pengali, (q - 0.75) * pengali, p * pengali, q * pengali, &HFF0000
    Garis Picture2, p * pengali, q * pengali, (p + 0.75) * pengali, (q - 0.75) * pengali, &HFF0000
    Garis Picture2, (p + 0.75) * pengali, (q - 0.75) * pengali, (p - 0.75) * pengali, (q - 0.75) * pengali, &HFF0000
    Garis Picture3, (p - 0.75) * pengali, (q - 0.75) * pengali, p * pengali, q * pengali, &HFF0000
    Garis Picture3, p * pengali, q * pengali, (p + 0.75) * pengali, (q - 0.75) * pengali, &HFF0000
    Garis Picture3, (p + 0.75) * pengali, (q - 0.75) * pengali, (p - 0.75) * pengali, (q - 0.75) * pengali, &HFF0000
    Garis Picture4, (p - 0.75) * pengali, (q - 0.75) * pengali, p * pengali, q * pengali, &HFF0000
    Garis Picture4, p * pengali, q * pengali, (p + 0.75) * pengali, (q - 0.75) * pengali, &HFF0000
    Garis Picture4, (p + 0.75) * pengali, (q - 0.75) * pengali, (p - 0.75) * pengali, (q - 0.75) * pengali, &HFF0000
  Else
    Lingkar Picture1, p * pengali, (q - 0.25) * pengali, 0.3 * pengali, &HFF0000
    Garis Picture1, (p - 0.75) * pengali, (q - 0.6) * pengali, (p + 0.75) * pengali, (q - 0.6) * pengali, &HFF0000
    Lingkar Picture2, p * pengali, (q - 0.25) * pengali, 0.3 * pengali, &HFF0000
    Garis Picture2, (p - 0.75) * pengali, (q - 0.6) * pengali, (p + 0.75) * pengali, (q - 0.6) * pengali, &HFF0000
    Lingkar Picture3, p * pengali, (q - 0.25) * pengali, 0.3 * pengali, &HFF0000
    Garis Picture3, (p - 0.75) * pengali, (q - 0.6) * pengali, (p + 0.75) * pengali, (q - 0.6) * pengali, &HFF0000
    Lingkar Picture4, p * pengali, (q - 0.25) * pengali, 0.3 * pengali, &HFF0000
    Garis Picture4, (p - 0.75) * pengali, (q - 0.6) * pengali, (p + 0.75) * pengali, (q - 0.6) * pengali, &HFF0000
  End If
Next i
Dim beban() As Single
Dim nlm As Single
Dim q1() As Single, r1() As Single, q2() As Single, r2() As Single, q3() As Single, r3() As Single, q4() As Single, r4() As Single,
q5() As Single, r5() As Single, q6() As Single, r6() As Single
nlm = (Form2.Text2.text)
ReDim q1(jumlah_batang), r1(jumlah_batang), q2(jumlah_batang), r2(jumlah_batang), q3(jumlah_batang), r3(jumlah_batang),
q4(jumlah_batang), r4(jumlah_batang), q5(jumlah_batang), r5(jumlah_batang), q6(jumlah_batang), r6(jumlah_batang)
ReDim beban(jumlah_batang)
For j = 1 To nlm
  i = Form2.MSFlexGrid2.TextMatrix(j, 0)
  q1(i) = Form2.MSFlexGrid2.TextMatrix(j, 1)
  r1(i) = Form2.MSFlexGrid2.TextMatrix(j, 2)
  q2(i) = Form2.MSFlexGrid2.TextMatrix(j, 3)
  r2(i) = Form2.MSFlexGrid2.TextMatrix(j, 4)
  q3(i) = Form2.MSFlexGrid2.TextMatrix(j, 5)
  r3(i) = Form2.MSFlexGrid2.TextMatrix(j, 6)
  q4(i) = Form2.MSFlexGrid2.TextMatrix(j, 7)
  r4(i) = Form2.MSFlexGrid2.TextMatrix(j, 8)
  q5(i) = Form2.MSFlexGrid2.TextMatrix(j, 9)
  r5(i) = Form2.MSFlexGrid2.TextMatrix(j, 10)
  q6(i) = Form2.MSFlexGrid2.TextMatrix(j, 11)
  r6(i) = Form2.MSFlexGrid2.TextMatrix(j, 12)
  beban(i) = 1
```

```
Next j
Dim piasver1 As Single
Dim piashor1 As Single
Dim piasver2 As Single
Dim piashor2 As Single
Dim pias As Single
Dim qsum(25, 40000)
Dim pengurang As Single
Dim pengurangtetap As Single
Dim moawal As Single
Dim lendut As Single
'bikin gambar batang-batang
For i = 1 To jumlah_batang
   x = x + 1
   titik1 = (frmContoh1.MSFlexGrid6.TextMatrix(i, 1))
   titik2 = (frmContoh1.MSFlexGrid6.TextMatrix(i, 2))
   E = a(titik1, 1) + 4
   f = a(titik1, 2) + 4
   g = a(titik2, 1) + 4
   h = a(titik2, 2) + 4
'ceritanya mau bikin gambar axial force
   axawal = frmContoh2.MSFlexGrid2.TextMatrix(i, 1)
   axakhir = frmContoh2.MSFlexGrid2.TextMatrix(i, 4)
   'batang vertikal
   If E = g Then
      If f <= h Then posneg = 1 Else posneg = -1
      For axial = (f + 0.05 * posneg) To (h - 0.05 * posneg) Step (0.01 * posneg)
         If axawal = axakhir And axawal < 0 Then Garis Picture1, E * pengali, axial * pengali, E * pengali + axawal * pAx, axial *
pengali, vbRed
         If axawal = axakhir And axawal > 0 Then Garis Picture1, E * pengali, axial * pengali, E * pengali + axawal * pAx, axial *
pengali, vbYellow
      Next axial
   'batang horizontal
   Else
      If E <= g Then posneg = 1 Else posneg = -1
      For axial = (E + 0.05 * posneg) To (g - 0.05 * posneg) Step (0.01 * posneg)
         If axawal = axakhir And axawal < 0 Then Garis Picture1, axial * pengali, f * pengali, axial * pengali, f * pengali + axawal *
pAx, vbRed
         If axawal = axakhir And axawal > 0 Then Garis Picture1, axial * pengali, f * pengali, axial * pengali, f * pengali + axawal *
pAx, vbYellow
      Next axial
   End If
   axawal = frmContoh2.MSFlexGrid2.TextMatrix(i, 2)
   axakhir = frmContoh2.MSFlexGrid2.TextMatrix(i, 5)
   If E = g Then
      If f <= h Then posneg = 1 Else posneg = -1
      For shear = (f + 0.05 * posneg) To (h - 0.05 * posneg) Step (0.01 * posneg)
         axawal = frmContoh2.MSFlexGrid2.TextMatrix(i, 2)
         pengurang = 0
         If r1(i) = r2(i) Then
            pengurang = 0
         Else
            If shear > (r1(i) + f) Then pengurang = (q1(i) + (q1(i) + (q2(i) - q1(i)) * (shear - (r1(i) + f)) / Abs(r2(i) - r1(i)))) * 0.5 *
(shear - (r1(i) + f))
            If shear > (r2(i) + f) Then pengurang = pengurangtetap
         End If
         If shear > (r3(i) + f) Then pengurang = pengurang + q3(i)
         If shear > (r4(i) + f) Then pengurang = pengurang + q4(i)
         If shear > (r5(i) + f) Then pengurang = pengurang + q5(i)
         If shear > (r6(i) + f) Then pengurang = pengurang + q6(i)
         axawal = axawal - pengurang
         If axawal < 0 Then Garis Picture2, E * pengali, shear * pengali, E * pengali + axawal * pAx, shear * pengali, vbRed
         If axawal > 0 Then Garis Picture2, E * pengali, shear * pengali, E * pengali + axawal * pAx, shear * pengali, vbYellow
         pengurangtetap = pengurang
      Next shear
   Else
      If E <= g Then posneg = 1 Else posneg = -1
      For shear = (E + 0.05 * posneg) To (g - 0.05 * posneg) Step (0.01 * posneg)
         axawal = frmContoh2.MSFlexGrid2.TextMatrix(i, 2)
         pengurang = 0
         If r1(i) = r2(i) Then
```

```
            pengurang = 0
          Else
            If shear > (r1(i) + E) Then pengurang = (q1(i) + (q1(i) + (q2(i) - q1(i)) * (shear - (r1(i) + E)) / Abs(r2(i) - r1(i)))) * 0.5 *
(shear - (r1(i) + E))
            If shear > (r2(i) + E) Then pengurang = pengurangtetap
          End If
          If shear > (r3(i) + E) Then pengurang = pengurang + q3(i)
          If shear > (r4(i) + E) Then pengurang = pengurang + q4(i)
          If shear > (r5(i) + E) Then pengurang = pengurang + q5(i)
          If shear > (r6(i) + E) Then pengurang = pengurang + q6(i)
          axawal = axawal - pengurang
          'MsgBox "shear= " & shear & Chr(13) & "pengurang= " & pengurang, vbOKOnly
          If axawal < 0 Then Garis Picture2, shear * pengali, f * pengali, shear * pengali, f * pengali + axawal * pAx, vbRed
          If axawal > 0 Then Garis Picture2, shear * pengali, f * pengali, shear * pengali, f * pengali + axawal * pAx, vbYellow
          pengurangtetap = pengurang
        Next shear
      End If
      axawal = frmContoh2.MSFlexGrid2.TextMatrix(i, 3)
      axakhir = frmContoh2.MSFlexGrid2.TextMatrix(i, 6)
      If E = g Then
        If f <= h Then posneg = 1 Else posneg = -1
        For moment = (f + 0.05 * posneg) To (h - 0.05 * posneg) Step (0.01 * posneg)
          moawal = frmContoh2.MSFlexGrid2.TextMatrix(i, 3)
          axawal = frmContoh2.MSFlexGrid2.TextMatrix(i, 2)
          pengurang = axawal * (moment - f)
          If r1(i) = r2(i) Then
            pengurang = axawal * (moment - f)
          Else
            If q1(i) <= q2(i) Then
              If moment > (r1(i) + f) Then pengurang = pengurang - q1(i) * (moment - (r1(i) + f)) * 0.5 * (moment - (r1(i) + f)) - 0.5
* (q2(i) - q1(i)) * (moment - (r1(i) + f)) * 1 / 3 * (moment - (r1(i) + f))
              If moment > (r2(i) + f) Then pengurang = pengurangtetap
            ElseIf q1(i) > q2(i) Then
              If moment > (r1(i) + f) Then pengurang = pengurang - q2(i) * (moment - (r1(i) + f)) * 0.5 * (moment - (r1(i) + f)) - 0.5
* (q1(i) - q2(i)) * (moment - (r1(i) + f)) * 2 / 3 * (moment - (r1(i) + f))
              If moment > (r2(i) + f) Then pengurang = pengurangtetap
            End If
          End If
          If moment > (r3(i) + f) Then pengurang = pengurang - q3(i) * (moment - (r3(i) + f))
          If moment > (r4(i) + f) Then pengurang = pengurang - q4(i) * (moment - (r4(i) + f))
          If moment > (r5(i) + f) Then pengurang = pengurang - q5(i) * (moment - (r5(i) + f))
          If moment > (r6(i) + f) Then pengurang = pengurang - q6(i) * (moment - (r6(i) + f))
          moawal = moawal + pengurang * posneg
          If axawal >= 0 Then
            If moawal < 0 Then Garis Picture3, E * pengali, moment * pengali, E * pengali + moawal * pAx, moment * pengali,
vbRed
            If moawal > 0 Then Garis Picture3, E * pengali, moment * pengali, E * pengali + moawal * pAx, moment * pengali,
vbYellow
          Else
            If moawal > 0 Then Garis Picture3, E * pengali, moment * pengali, E * pengali + moawal * pAx, moment * pengali,
vbYellow
            If moawal < 0 Then Garis Picture3, E * pengali, moment * pengali, E * pengali + moawal * pAx, moment * pengali,
vbRed
          End If
          pengurang = pengurangtetap
        Next moment
      Else
        If E <= g Then posneg = 1 Else posneg = -1
        For moment = (E + 0.05 * posneg) To (g - 0.05 * posneg) Step (0.01 * posneg)
          moawal = frmContoh2.MSFlexGrid2.TextMatrix(i, 3)
          axawal = frmContoh2.MSFlexGrid2.TextMatrix(i, 2)
          pengurang = axawal * (moment - E)
          If r1(i) = r2(i) Then
            pengurang = axawal * (moment - E)
          Else
            If q1(i) <= q2(i) Then
              If moment > (r1(i) + E) Then pengurang = pengurang - q1(i) * (moment - (r1(i) + E)) * 0.5 * (moment - (r1(i) + E)) -
0.5 * (q2(i) - q1(i)) * (moment - (r1(i) + E)) * 1 / 3 * (moment - (r1(i) + E))
              If moment > (r2(i) + E) Then pengurang = pengurangtetap
            ElseIf q1(i) > q2(i) Then
```

```
            If moment > (r1(i) + E) Then pengurang = pengurang - q2(i) * (moment - (r1(i) + E)) * 0.5 * (moment - (r1(i) + E)) -
0.5 * (q1(i) - q2(i)) * (moment - (r1(i) + E)) * 2 / 3 * (moment - (r1(i) + E))
            If moment > (r2(i) + E) Then pengurang = pengurangtetap
          End If
        End If
        If moment > (r3(i) + E) Then pengurang = pengurang - q3(i) * (moment - (r3(i) + E))
        If moment > (r4(i) + E) Then pengurang = pengurang - q4(i) * (moment - (r4(i) + E))
        If moment > (r5(i) + E) Then pengurang = pengurang - q5(i) * (moment - (r5(i) + E))
        If moment > (r6(i) + E) Then pengurang = pengurang - q6(i) * (moment - (r6(i) + E))
        moawal = moawal + pengurang
        If moawal < 0 Then Garis Picture3, moment * pengali, f * pengali, moment * pengali, f * pengali - moawal * pAx, vbRed
        If moawal > 0 Then Garis Picture3, moment * pengali, f * pengali, moment * pengali, f * pengali - moawal * pAx,
vbYellow
      Next moment
    End If
    Garis Picture1, E * pengali, f * pengali, g * pengali, h * pengali, vbBlack & i
    Picture1.FillStyle = 0
    Lingkar Picture1, E * pengali, f * pengali, 0.15 * pengali, &HC000&
    Lingkar Picture1, g * pengali, h * pengali, 0.15 * pengali, &HC000&
    Picture1.FillStyle = 1
    Garis Picture2, E * pengali, f * pengali, g * pengali, h * pengali, vbBlack & i
    Picture2.FillStyle = 0
    Lingkar Picture2, E * pengali, f * pengali, 0.15 * pengali, &HC000&
    Lingkar Picture2, g * pengali, h * pengali, 0.15 * pengali, &HC000&
    Picture2.FillStyle = 1
    Garis Picture3, E * pengali, f * pengali, g * pengali, h * pengali, vbBlack & i
    Picture3.FillStyle = 0
    Lingkar Picture3, E * pengali, f * pengali, 0.15 * pengali, &HC000&
    Lingkar Picture3, g * pengali, h * pengali, 0.15 * pengali, &HC000&
    Picture3.FillStyle = 1
    Garis Picture4, E * pengali, f * pengali, g * pengali, h * pengali, vbBlack & i
    Picture4.FillStyle = 0
    Lingkar Picture4, E * pengali, f * pengali, 0.15 * pengali, &HC000&
    Lingkar Picture4, g * pengali, h * pengali, 0.15 * pengali, &HC000&
    Picture4.FillStyle = 1
    dispx1 = disp(titik1, 1)
    dispy1 = disp(titik1, 2)
    ED = E + dispx1 * dispxscale
    fD = f + dispy1 * dispyscale
    dispx2 = disp(titik2, 1)
    dispy2 = disp(titik2, 2)
    gD = g + dispx2 * dispxscale
    hD = h + dispy2 * dispyscale
    If Int(f) = Int(h) Then
      lendut = 0.1
      tinggi = Abs(fD - hD) / 10
      jarak = Abs(ED - gD) / 10
      If E <= g Then posneg = 1 Else posneg = -1
      If fD > hD Then
        Garis Picture4, ED * pengali, fD * pengali, (ED + 1 * jarak) * pengali, (fD - 1 * tinggi - 0.4762 * lendut) * pengali,
vbMagenta
        Garis Picture4, (ED + 1 * jarak) * pengali, (fD - 1 * tinggi - 0.4762 * lendut) * pengali, (ED + 2 * jarak) * pengali, (fD - 2 *
tinggi - 0.8095 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 2 * jarak) * pengali, (fD - 2 * tinggi - 0.8095 * lendut) * pengali, (ED + 3 * jarak) * pengali, (fD - 3 *
tinggi - 0.9048 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 3 * jarak) * pengali, (fD - 3 * tinggi - 0.9048 * lendut) * pengali, (ED + 4 * jarak) * pengali, (fD - 4 *
tinggi - 0.9542 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 4 * jarak) * pengali, (fD - 4 * tinggi - 0.9524 * lendut) * pengali, (ED + 5 * jarak) * pengali, (fD - 5 *
tinggi - 1 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 5 * jarak) * pengali, (fD - 5 * tinggi - 1 * lendut) * pengali, (ED + 6 * jarak) * pengali, (fD - 6 * tinggi
- 0.9542 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 6 * jarak) * pengali, (fD - 6 * tinggi - 0.9542 * lendut) * pengali, (ED + 7 * jarak) * pengali, (fD - 7 *
tinggi - 0.9048 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 7 * jarak) * pengali, (fD - 7 * tinggi - 0.9048 * lendut) * pengali, (ED + 8 * jarak) * pengali, (fD - 8 *
tinggi - 0.8095 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 8 * jarak) * pengali, (fD - 8 * tinggi - 0.8095 * lendut) * pengali, (ED + 9 * jarak) * pengali, (fD - 9 *
tinggi - 0.4762 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 9 * jarak) * pengali, (fD - 9 * tinggi - 0.4762 * lendut) * pengali, gD * pengali, hD * pengali,
vbMagenta
      ElseIf fD < hD Then
```

```
        Garis Picture4, ED * pengali, fD * pengali, (ED + 1 * jarak) * pengali, (fD + 1 * tinggi - 0.4762 * lendut) * pengali,
vbMagenta
        Garis Picture4, (ED + 1 * jarak) * pengali, (fD + 1 * tinggi - 0.4762 * lendut) * pengali, (ED + 2 * jarak) * pengali, (fD + 2
* tinggi - 0.8095 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 2 * jarak) * pengali, (fD + 2 * tinggi - 0.8095 * lendut) * pengali, (ED + 3 * jarak) * pengali, (fD + 3
* tinggi - 0.9048 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 3 * jarak) * pengali, (fD + 3 * tinggi - 0.9048 * lendut) * pengali, (ED + 4 * jarak) * pengali, (fD + 4
* tinggi - 0.9542 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 4 * jarak) * pengali, (fD + 4 * tinggi - 0.9524 * lendut) * pengali, (ED + 5 * jarak) * pengali, (fD + 5
* tinggi - 1 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 5 * jarak) * pengali, (fD + 5 * tinggi - 1 * lendut) * pengali, (ED + 6 * jarak) * pengali, (fD + 6 *
tinggi - 0.9542 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 6 * jarak) * pengali, (fD + 6 * tinggi - 0.9542 * lendut) * pengali, (ED + 7 * jarak) * pengali, (fD + 7
* tinggi - 0.9048 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 7 * jarak) * pengali, (fD + 7 * tinggi - 0.9048 * lendut) * pengali, (ED + 8 * jarak) * pengali, (fD + 8
* tinggi - 0.8095 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 8 * jarak) * pengali, (fD + 8 * tinggi - 0.8095 * lendut) * pengali, (ED + 9 * jarak) * pengali, (fD + 9
* tinggi - 0.4762 * lendut) * pengali, vbMagenta
        Garis Picture4, (ED + 9 * jarak) * pengali, (fD + 9 * tinggi - 0.4762 * lendut) * pengali, gD * pengali, hD * pengali,
vbMagenta
      End If
    Else
      Garis Picture4, ED * pengali, fD * pengali, gD * pengali, hD * pengali, vbMagenta
    End If
    Picture4.FillColor = vbMagenta
    Picture4.FillStyle = 0
    Lingkar Picture4, ED * pengali, fD * pengali, 0.15 * pengali, vbMagenta
    Lingkar Picture4, gD * pengali, hD * pengali, 0.15 * pengali, vbMagenta
    Picture4.FillStyle = 1
    Picture4.FillColor = &HC000&
    If Int(Round(E)) = Int(Round(g)) Then
      drawText Picture1, (E - 1.5) * pengali, 0.5 * (f + h) * pengali, "(" & i & ")", vbBlue, 10
      drawText Picture2, (E - 1.5) * pengali, 0.5 * (f + h) * pengali, "(" & i & ")", vbBlue, 10
      drawText Picture3, (E - 1.5) * pengali, 0.5 * (f + h) * pengali, "(" & i & ")", vbBlue, 10
    Else
      drawText Picture1, 0.5 * (E + g - 1.5) * pengali, (h + 1) * pengali, "(" & i & ")", vbBlue, 10
      drawText Picture2, 0.5 * (E + g - 1.5) * pengali, (h + 1) * pengali, "(" & i & ")", vbBlue, 10
      drawText Picture3, 0.5 * (E + g - 1.5) * pengali, (h + 1) * pengali, "(" & i & ")", vbBlue, 10
    End If
    drawText Picture4, (E - 1) * pengali, (f + 1) * pengali, "(" & titik1 & ")", vbBlack, 10
    drawText Picture4, (g - 1) * pengali, (h + 1) * pengali, "(" & titik2 & ")", vbBlack, 10
Next i
End Sub
Private Sub Garis(pic As PictureBox, x1 As Single, y1 As Single, x2 As Single, y2 As Single, warna As Long)
pic.Line (Round(x1), Round(maxy - y1))-(Round(x2), Round(maxy - y2)), warna
End Sub
Private Sub Lingkar(pic As PictureBox, x As Single, y As Single, rad As Single, warna As Long)
pic.Circle (Round(x), Round(maxy - y)), rad, warna
End Sub
Private Sub drawText(pic As PictureBox, x As Single, y As Single, text As String, warna As Long, ukuran As Integer)
  pic.ForeColor = warna
  pic.FontSize = ukuran
  pic.CurrentY = maxy - y
  pic.CurrentX = x
  pic.FontBold = True
  pic.Print text
End Sub
Private Sub cmdskalamin_Click()
dispxscale = dispxscale / 4
dispyscale = dispyscale / 3
DrawAll
End Sub
Private Sub cmdskalaplus_Click()
dispxscale = dispxscale * 4
dispyscale = dispyscale * 3
DrawAll
End Sub
Private Sub Command1_Click()
Unload Me
End Sub
```

```
Private Sub Command10_Click()
M = (frmContoh1.Text1.text)
frmValue.Show
frmValue.Caption = "Value of Bending Moment Diagram"
With frmValue.MSFlexGrid1
   .Rows = M + 1
   .TextMatrix(0, 0) = "Frame ID"
   .TextMatrix(0, 1) = "Moment-i"
   .TextMatrix(0, 2) = "Moment-j"
   For i = 1 To M
      .TextMatrix(i, 0) = frmContoh2.MSFlexGrid2.TextMatrix(i, 0)
      .TextMatrix(i, 1) = frmContoh2.MSFlexGrid2.TextMatrix(i, 3)
      .TextMatrix(i, 2) = frmContoh2.MSFlexGrid2.TextMatrix(i, 6)
   Next i
End With
End Sub
Private Sub Command2_Click()
pengali = pengali + pengali / magnify
pAx = pAx + pAx / magnify
DrawAll
End Sub
Private Sub Command3_Click()
pengali = pengali - pengali / magnify
pAx = pAx - pAx / magnify
DrawAll
End Sub
Private Sub Command4_Click()
End
End Sub
Private Sub Command5_Click()
pAx = pAx + 1
DrawAll
End Sub
Private Sub Command6_Click()
If pAx < 0.5 Then pAx = 1 Else pAx = pAx - 1
DrawAll
End Sub
Private Sub Command7_Click()
NJ = frmContoh1.Text2.text
frmValue.Show
frmValue.Caption = "Value of Joint Displacement"
With frmValue.MSFlexGrid1
   .Cols = 4
   .Rows = NJ + 1
   .ColWidth(3) = 1200
   .TextMatrix(0, 0) = "Joint ID"
   .TextMatrix(0, 1) = "Ux"
   .TextMatrix(0, 2) = "Uy"
   .TextMatrix(0, 3) = "Rz"
   For i = 1 To NJ
      For j = 0 To 3
         .TextMatrix(i, j) = frmContoh2.MSFlexGrid1.TextMatrix(i, j)
      Next j
   Next i
End With
End Sub
Private Sub Command8_Click()
M = (frmContoh1.Text1.text)
frmValue.Show
frmValue.Caption = "Value of Normal Force Diagram"
With frmValue.MSFlexGrid1
   .Rows = M + 1
   .TextMatrix(0, 0) = "Frame ID"
   .TextMatrix(0, 1) = "Axial-i"
   .TextMatrix(0, 2) = "Axial-j"
   For i = 1 To M
      .TextMatrix(i, 0) = frmContoh2.MSFlexGrid2.TextMatrix(i, 0)
      .TextMatrix(i, 1) = frmContoh2.MSFlexGrid2.TextMatrix(i, 1)
      .TextMatrix(i, 2) = frmContoh2.MSFlexGrid2.TextMatrix(i, 4)
   Next i
End With
```

```
End Sub
Private Sub Command9_Click()
M = (frmContoh1.Text1.text)
frmValue.Show
frmValue.Caption = "Value of Shear Force Diagram"
With frmValue.MSFlexGrid1
   .Rows = M + 1
   .TextMatrix(0, 0) = "Frame ID"
   .TextMatrix(0, 1) = "Shear-i"
   .TextMatrix(0, 2) = "Shear-j"
   For i = 1 To M
      .TextMatrix(i, 0) = frmContoh2.MSFlexGrid2.TextMatrix(i, 0)
      .TextMatrix(i, 1) = frmContoh2.MSFlexGrid2.TextMatrix(i, 2)
      .TextMatrix(i, 2) = frmContoh2.MSFlexGrid2.TextMatrix(i, 5)
   Next i
End With
End Sub
Private Sub Form_Activate()
maxy = Picture1.Height
Picture1.BackColor = vbWhite
Picture2.BackColor = vbWhite
Picture3.BackColor = vbWhite
Picture4.BackColor = vbWhite
Picture1.DrawWidth = 2
Picture2.DrawWidth = 2
Picture3.DrawWidth = 2
Picture4.DrawWidth = 2
SSTab1.TabIndex = 0
DrawAll
End Sub
Private Sub Form_Load()
pAx = 5
pengali = 400
radius = 5
dispxscale = 100
dispyscale = 100
magnify = 10
End Sub
```

Tampilan Nilai NFD, SFD, dan BMD

```
Private Sub Command1_Click()
MSFlexGrid1.Cols = 3
Unload Me
End Sub
Private Sub Form_Load()
With MSFlexGrid1
   .ColWidth(0) = 800
   .ColWidth(1) = 1200
   .ColWidth(2) = 1200
End With
End Sub
```