YILDIZ TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

Transport Probleminin Yenilenmiş Simpleks Yöntemi

H. İbrahim Sezginman

Doktora Tezi

YILDIZ ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

DOKTORA TEZİ

TRANSPORT PROBLEMİNİN
YENİLENMİŞ SİMPLEKS
YÖNTEMİYLE
ÇÖZÜMÜ İÇİN
BİR ALGORİTMA

PROF.H.İBRAHİM SEZGİNMAN

FEN-EDEBİYAT FAKÜLTESİ

MATEMATİK BÖLÜMÜ

İSTANBUL-1984

D İ Z İ N

ÖZET ....(1)

İNGİLİZCE ÖZET ....(11)

1.BÖLÜM:

Simpleks çarpanları 1

2.BÖLÜM:

Yenilenmiş simpleks yöntem 3

3.BÖLÜM:

Transport problemi 4

4.BÖLÜM:

Transport probleminin yenilenmiş simpleks yöntemiyle

çözümü için bir algoritma 7

5.BÖLÜM:

Akış diyagramı 11

ÖZET

Transport probleminin bir lineer programlama problemi olduğu ve bu nedenle de s i m p l e k s y ö n t e m l e çözülebildiği bilinmektedir. Kısıtlardaki katsayıların sıfır ya da bir olmaları nedeniyle transport problemi değişik yöntemlerle de çözülmektedir.

Bu çalışmamda, transport probleminin yenilenmiş simpleks (revised simplex) yöntemiyle çözümü için bir algoritma oluşturulmuştur. Bu algoritmayla transport probleminin çözümü, temelde bir matrisin inversini ve iki matrisin çarpımını bulmaya dönüştürülmektedir; bu nedenle, bu algoritma transport probleminin bilgisayarlarla çözümünde oldukça kullanışlıdır.

Birinci bölümde simpleks çarpanları, ikinci bölümde yenilenmiş simpleks yöntem (revised simplex method), üçüncü bölümde transport problemi hakkında genel bilgiler verilmiş, dördüncü bölümde " transport probleminin yenilenmiş simpleks yöntemiyle çözümü için bir algoritma " oluşturularak bir problemin çözümüne uygulanmıştır ve beşinci bölümde de problemin bilgisayarlarla çözümü için bir akış diyagramı yapılmıştır.

SUMMARY

It's known that transportation problem is a linear programming problem and therefore can be solved by simplex method. Because the coefficients in restrictions are zero or one the transportation problems can be solved by using different methods.

In this thesis I have developed an algorithm to be used in solving transportation problems by revised simplex method. In fact, using this algorithm in solving a transportation problem, transforms it to find the invers of a matrix and multiply two matrices. Therefore , in solving the transportation problems by computer, this algorithm is very practical.

The first three chapters give an overview on simplex multipliers, revised simplex method, and transportation problem respectivly. In the forth chapter an ' algorithm for solving transportation problems by revised method ' is developed and applied to and example. In fifth chapter a computer flow diagram for such problem is given.

I. BÖLÜM

SİMPLEKS ÇARPANLARI (1)

$$a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{im}x_m + \ldots + a_{in}x_n = b_i \quad (1 \le i \le m) \qquad (1.1)$$

$$x_j \ge 0 \quad (1 \le j \le n) \qquad (1.2)$$

$$\min z = \sum_{j=1}^{n} c_j x_j \qquad (1.3)$$

lineer programlama problemini gözönüne alalım.

$B_o = (V_1, V_2, \ldots, V_m)$ nin bir baz oluşturduğunu kabul ederek z amaç fonksiyonundaki $x_1, x_2, \ldots, x_m$ temel değişkenlerini yoketmek isteyelim.

(1.1) koşullarındaki i ($1 \le i \le m$) numaralı satırlardan her birini s_i sayısıyla çarpıp z amaç fonksiyonuna ekleyelim:

$$\sum_{j=1}^{n} (c_j + \sum_{i=1}^{m} a_{ij}s_i)x_j = z + \sum_{i=1}^{m} b_i s_i \qquad (1.4)$$

$x_1, x_2, \ldots, x_m$ temel değişkenlerinin yok edilmesi amaçlandığından katsayıları sıfır olmalıdır:

$$\sum_{i=1}^{m} a_{ij}s_i = -c_j \quad (1 \le j \le m) \qquad (1.5)$$

Burada

---

(1) KVMital, Optimization Methods in operations research and systems analysis, Wiley Eastern Limited, 1977, S. 79

$$S = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_m \end{bmatrix}, \quad B = \begin{bmatrix} a_{11} & a_{12} \,.\,. & a_{1p} & .\,.\, a_{1m} \\ a_{21} & a_{22} & a_{2p} & .\,.\, a_{2m} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} \,.\,. & a_{mp} & .\,.\, a_{mm} \end{bmatrix}, \qquad C = \begin{bmatrix} c_1 & c_2 . \; . \; . \; . & c_m \end{bmatrix}$$

olmak üzere (1.5) sistemi matrisyel yazılabilir:

$$\sum_{i=1}^{m} a_{ij} s_i = -c_j \quad (1 \le j \le m) \longrightarrow S'B = -C \rightarrow S' = -CB^{-1} \tag{1.6}$$

(1.6) sisteminden elde edilen

$S' = [s_1 \;\; s_2 \dots\, s_m]$, $s_1, s_2, \dots, s_m$ değerleri (1.4) eşitliğinde yerleştirilerek z amaç fonksiyonunda temel değişkenler yok edilmiş olur:

$$z = -\sum_{i=1}^{m} b_i s_i + \sum_{j=m+1}^{n} \left(c_j + \sum_{i=1}^{m} a_{ij} s_i\right) x_j \tag{1.7}$$

$z^* = z + \sum_{i=1}^{m} b_i s_j$, $c_j^* = c_j + \sum_{i=1}^{m} a_{ij} s_i$ dönüşümleriyle (1.7) eşitliği

$$z^* = \sum_{j=m+1}^{n} c_j^* x_j \tag{1.8}$$

şeklini alır.

Sonuç:

1-) Temel olmayan x_j $(m+1 \le j \le n)$ değişkenlerinin değerleri

$x_{m+1} = x_{m+2} = \dots = x_n = 0$ olduğundan

$z^* = 0$ dır.

2-) c_j^* $(m+1 \le j \le n)$ lar arasından, meselâ $c_r^* < 0$ olsun; bu takdirde x_r temel değişken yapılırsa $x_r \ge 0$ olarak çözüme gireceğinden

$z^* \le 0$ olur.

c_j^* lar içinde negatif olanlar tükenmişse z^* daha küçültülemez; yani optimal çözüme ulaşılmıştır.

Tanım:

(1.6) eşitliğindeki

$S' = [s_1\ s_2 \ldots\ s_m]$, $s_1, s_2, \ldots, s_m$ sayılarına s i m p l e k s ç a r p a n l a r ı adı verilir.

2. BÖLÜM

YENİLENMİŞ SİMPLEKS YÖNTEM*

Bir aşamada baz

$$B_o = (v_1, v_2, \ldots, v_p, \ldots, v_m)$$

olsun ve v_p vektörünün çıkıp yerine v_r vektörünün girmesine karar verilerek

$$B_1 = (v_1, v_2, \ldots, v_r, \ldots, v_m)$$

bazına atlansın; konumuz B_1^{-1} matrisini B_o^{-1} cinsinden ifade etmektir.

$$B_0^{-1} B_1 = B_0^{-1} \begin{bmatrix} a_{11} & a_{12} \ldots & a_{1r} \ldots & a_{1m} \\ a_{21} & a_{22} \ldots & a_{2r} \ldots & a_{2m} \\ \ldots & \ldots & \ldots & \ldots \\ a_{m1} & a_{m2} \ldots & a_{mr} \ldots & a_{mm} \end{bmatrix} = \begin{bmatrix} 1 & 0 \ldots & a_{1r}^* & \ldots 0 \\ 0 & 1 \ldots & a_{2r}^* & \ldots 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 \ldots & a_{mr}^* & \ldots 0 \end{bmatrix} \qquad (2.1)$$

$$T = \begin{bmatrix} 1 & 0 \ldots & -a_{1r}^* / a_{pr}^* & \ldots & 0 \\ 0 & 1 \ldots & -a_{2r}^* / a_{pr}^* & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 \ldots & 1/ a_{pr}^* & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 \ldots & -a_{mr}^* / a_{pr}^* & \ldots & 1 \end{bmatrix}$$

olmak üzere (2.1) eşitliğini soldan T matrisiyle çarpalım:

---

*Revised simplex method - karşılığında kullanılmıştır.

$T(B_o^{-1} B_1) = I$ ve $T(B_o^{-1} B_1)B_1^{-1} = I B_1^{-1}$ den $B_1^{-1} = TB_o^{-1}$ (2.2)

elde edilir.

Sonuç:

1-) Baza girecek V_r^* vektörü 1.Bölümdeki sonuç 2 yardımıyla belirlenir.

2-) Bazdan çıkacak V_p vektörü

$$B_o^{-1}\left[V_o \; V_r\right] = \left[V_o^* \; V_r^*\right]$$

matrisinde V_r^* ın pozitif elemanları için oran testi uygulanarak kararlaştırılır.

3-) Bir sonraki iterasyon için B_1^{-1} matrisi B_o^{-1} olarak kabul edilir.

4-) Optimal çözüme ulaşana kadar iterasyonlar devam eder.

3. BÖLÜM

TRANSPORT PROBLEMİ

Transport model:

m tane üretim merkezi, n tane tüketim merkezi olsun

$$(1 \le i \le m) \quad \overset{(i)}{\boxed{a_i}} \xrightarrow[c_{ij}]{x_{ij}} \overset{(j)}{\boxed{b_J}} \quad (1 \le j \le n)$$

x_{ij} : (i) numaralı üretim merkezinden (kaynaktan) (j) numaralı tüketim merkezine(pazara) taşınan mal miktarını,

c_{ij} : (i) numaralı kaynaktan (j) numaralı pazara birim taşıma fiatını;

a_i : (i) numaralı kaynaktaki mal miktarını;

b_j : (j) numaralı tüketim merkezinin istemini göstermektedir.

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

koşulu altında, kaynaklardaki malların pazarlara en ucuza taşınması t r a n s - p o r t p r o b l e m i dir.

Aşağıdaki tablo yapılabilir:

(i) \ (j)	(1)	(2)	....	(j)	....	(n)	a_i ↓
(1)	x_{11} c_{11}	x_{12} c_{12}	....	x_{1j} c_{1j}	....	x_{1n} c_{1n}	a_1
(2)	x_{21} c_{21}	x_{22} c_{22}	....	x_{2j} c_{2j}	....	x_{2n} c_{2n}	a_2
⋮	⋮						⋮
(i)	x_{i1} c_{i1}	x_{i2} c_{i2}	....	x_{ij} c_{ij}	....	x_{in} c_{in}	a_i
⋮							⋮
(m)	x_{m1} c_{m1}	x_{m2} c_{m2}	....	x_{mj} c_{mj}	....	x_{mn} c_{mn}	a_m
b_j →	b_1	b_2	....	b_j	....	b_n	Σa Σb

TABLO (3.1)

(i,j) $(1 \leq i \leq m,\ 1 \leq j \leq n)$ hücrelerinin her biri için x_{ij} miktarının taşıma fiatı

$$c_{ij}x_{ij}$$

dir.

Bu nedenle, toplam taşıma fiatı

$$z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}x_{ij} \qquad (3.2)$$

dir.

Sınırlayıcı koşullar:

Tablo (3.1) in satırlarından ve sütunlarından

$$\sum_{j=1}^{n} x_{ij} = a_i \qquad (1 \leq i \leq m) \qquad (3.3)$$ satır denklemleri;

$$\sum_{i=1}^{m} x_{ij} = b_j \qquad (1 \leq j \leq n \qquad (3.4)$$ sütun denklemleri

sınırlayıcı koşullardır. Şu halde transport problemin matematik modeli

$$\left.\begin{aligned} \sum_{j=1}^{n} x_{ij} &= a_i \qquad (1 \leq i \leq m) \\ \sum_{i=1}^{m} x_{ij} &= b_j \qquad (1 \leq j \leq n) \end{aligned}\right\} \qquad (3.5)$$

$$x_{ij} \geq 0 \qquad (1 \leq i \leq m, \quad 1 \leq j \leq n) \qquad (3.6)$$

$$\text{Min } z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \qquad (3.7)$$

dir.

Uyarı:

(3.5) sınırlayıcı koşularındaki m+n tane denklem lineer bağımlıdır; çünkü

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

varsayımından dolayı, m tane satır denklemini sütun denklemlerinin n-1 tanesinden çıkarırsak sütun denklemlerinden biri elde edilir. Şu halde (3.5) koşullarındaki denklemlerden en az bir tanesi atılabilir.

Başlangıç tablosu (son denklem atılmış ve x_{ij} ler tek indisli y_j değişkenlerine çevrilmiş olarak):

	$y_1=x_{11}$	$y_2=x_{12}$	..	$y_j=x_{1j}$	...	$y_n=x_{1n}$			...	$y_{m.n}=x_{mn}$
V_0	V_1	V_2	..	V_j	...	V_n	V_{n+1}	V_{n+2}	...	$V_{m.n}$
a_1	1	1	..	1	...	1	0	0	...	0
a_2	0	0	..	0	...	0	1	1	...	0
.	.	.	..	.	...	.	.	.	...	.
a_m	0	0	..	0	...	0	0	0	...	1
b_1	1	0	..	0	...	0	1	0	...	0
b_2	0	1	..	0	...	0	0	1	...	0
.	.	.	..	.	...	.	.	.	...	.
b_{n-1}	0	0	..	0	...	0	0	0	...	0

Sorun, yapma değişkenler kullanılmadan burada bir başlangıç bazı oluşturmaktır. Bunun için Tablo (3.1) de m+n-1 tane temel değişkenli bir dağıtım yapılır; temel değişkenlerin sayısının m+n-1 den az olması durumunda dejenere çözüm olduğu bilinmektedir.

$y_1, y_2, \ldots , y_{m+n-1}$ lerin temel değişken oldukları ve dolayısıyla da

$$B_o = (V_1, V_2, \ldots , V_{m+n-1})$$

in başlangıç bazı olduğunu kabul ederek yenilenmiş simpleks yöntemi uygulayalım.

Baza girecek vektör (1. Bölüm - sonuç 2) yardımıyla; bazdan çıkacak vektör (2.bölüm sonuç 2) yardımıyla belirlenerek B_1 bazına atlanır.

(2.Bölüm - sonuç 3 ve sonuç 4)deki işlemler yapılarak optimal çözüme ulaşılır.

TRANSPORT PROBLEMİNİN YENİLENMİŞ SİMPLEKS YÖNTEMİYLE ÇÖZÜMÜ İÇİN BİR ALGORİTMA

1-) Tablo (3.1) de m+n-1 tane temel değişkenli bir dağıtım yaparak dolu hücreler yardımıyla B_o başlangıç bazını ve B_o^{-1} matrisini oluşturunuz.

2-) Baza girecek V_r vektörünü (1.Bölüm - sonuç 2) yardımıyla belirtiniz; eğer c_j^* larda negatif olanlar tükenmişse minimum (optimal) çözüme ulaşılmıştır; aksi takdirde devam ediniz.

3-) Bazdan çıkacak V_p vektörünü (2.Bölüm - sonuç 2) yardımıyla kararlaştırınız

4-) 2. Bölümdeki T matrisini bulunuz ve

$$B_1^{-1} = T\,B_o^{-1} \qquad (2.2)$$

den B_1^{-1} matrisini elde ediniz.

5-) (2) ye gidiniz ve bir sonraki iterasyon için B_o^{-1} olarak B_1^{-1} matrisini alınız ($B_o^{-1} = B_1^{-1}$).

Bu algoritmayı aşağıdaki transport problemine uygulayalım.

(j) / (i)	(1)	(2)	(3)	(4)	a_i
(1)	$c_{11}=3$	$c_{12}=2$	$c_{13}=5$	$c_{14}=4$	$a_1 = 25$
(2)	$c_{21}=4$	$c_{22}=1$	$c_{23}=7$	$c_{24}=6$	$a_2 = 35$
(3)	$c_{31}=7$	$c_{32}=8$	$c_{33}=3$	$c_{34}=5$	$a_3 = 30$
b_j	$b_1 = 10$	$b_2 = 18$	$b_3 = 20$	$b_4 = 42$	

Birinci iterasyon:

1-) Tablo (3.1) de

10	15			25
	3	20	12	35
			30	30
10	18	20	42	

dağıtımı yardımıyla

$B_o = (V_1, V_2, V_6, V_7, V_8, V_{12})$

bazı oluşur ve

$$B_o^{-1} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & -1 & -1 & -1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad \text{dir.}$$

2-) (1.6) dan $S' = \begin{bmatrix} -7 & -6 & -5 & 4 & 5 & -1 \end{bmatrix}$;

$c_3^* = -3, \quad c_4^* = -3, \quad c_5^* = 2, \quad c_9^* = 6, \quad c_{10}^* = 8, \quad c_{11}^* = -3.$

V_3, V_4, V_{11} vektörlerinden herhangi biri baza girebilir; fiatı en küçük olan V_{11} vektörü baza girsin.

$$\text{3-)} \quad \begin{bmatrix} V_0^* & V_{11}^* \end{bmatrix} = \begin{bmatrix} 10 & 0 \\ 15 & 0 \\ 3 & 0 \\ 20 & 1 \\ 12 & -1 \\ 30 & 1 \end{bmatrix} \longrightarrow V_7 \text{ bazdan çıkar.}$$

$$\text{4-)} \quad T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 \end{bmatrix} \quad \text{ve} \quad B_1^{-1} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & -1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix} \quad \text{dir.}$$

5-) (2) ye gidiniz.

İkinci iterasyon:

2-) $B_o = (V_1, V_2, V_6, V_{11}, V_8, V_{12})$ ve $B_o^{-1} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & -1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix}$

dir.

$S' = \begin{bmatrix} -7 & -6 & -5 & 4 & 5 & 2 \end{bmatrix}$; $c_3^* = 0$, $c_4^* = -3$, $c_5^* = 3$, $c_7^* = 3$, $c_9^* = 6$, $c_{10}^* = 8$.

V_4 vektörü baza girer.

3-) $\begin{bmatrix} V_0^* & V_4^* \end{bmatrix} = \begin{bmatrix} 10 & 0 \\ 15 & 1 \\ 3 & -1 \\ 20 & 0 \\ 32 & 1 \\ 10 & 0 \end{bmatrix} \longrightarrow V_2$ bazdan çıkar.

4-) $T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$ ve $B_1^{-1} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix}$

dir.

5-) (2) ye gidiniz.

Üçüncü iterasyon:

2-) $B_o = (V_1, V_4, V_6, V_{11}, V_8, V_{12})$ ve $B_o^{-1} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix}$

dir.

$S' = \begin{bmatrix} -4 & -6 & -5 & 1 & 5 & 2 \end{bmatrix}$; $c_2^* = 3$, $c_3^* = 3$, $c_5^* = -1$, $c_7^* = 3$, $c_9^* = 3$, $c_{10}^* = 8$.

V_5 vektörü baza girer.

3-) $\begin{bmatrix} V_0^* & V_5^* \end{bmatrix} = \begin{bmatrix} 10 & 1 \\ 15 & -1 \\ 18 & 0 \\ 20 & 0 \\ 17 & 1 \\ 10 & 0 \end{bmatrix}$ $\longrightarrow$ V_1 vektörü bazdan çıkar.

4-) $I = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$ ve $B_1^{-1} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & -1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix}$

dir

5-) (2) ye gidiniz.

Dördüncü iterasyon

2-) $B_o = (V_5, V_4, V_6, V_{11}, V_8, V_{12})$ ve $B_o^{-1} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & -1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix}$

dir.

$S' = \begin{bmatrix} -4 & -6 & -5 & 2 & 5 & 2 \end{bmatrix}$; $c_1^* = 1$, $c_2^* = 3$, $c_3^* = 3$, $c_7^* = 3$, $c_9^* = 4$, $c_{10}^* = 8$.

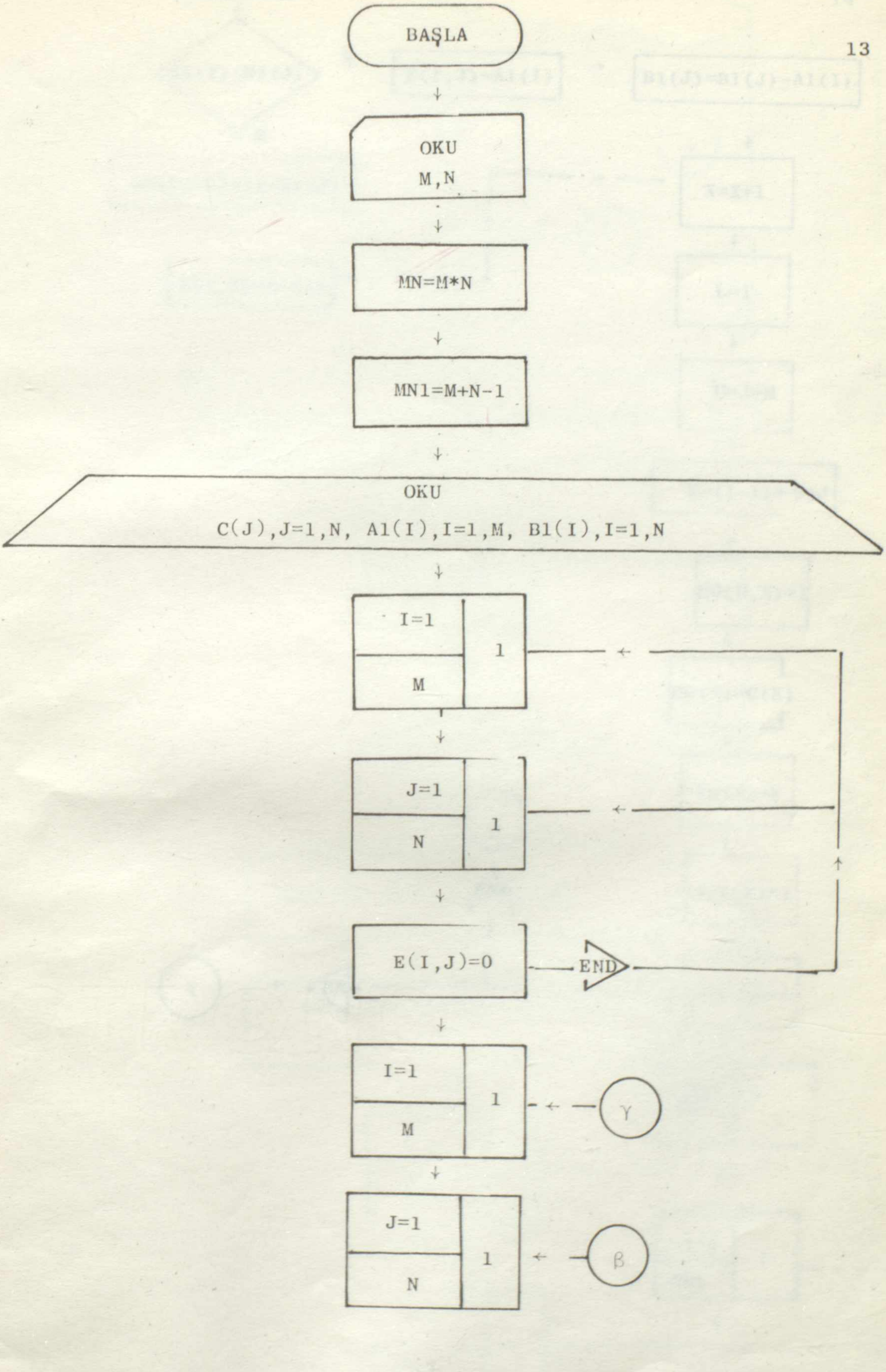
c_j^* ler içinde negatif olanlar tükendiğinden optimal çözüme ulaşılmıştır.

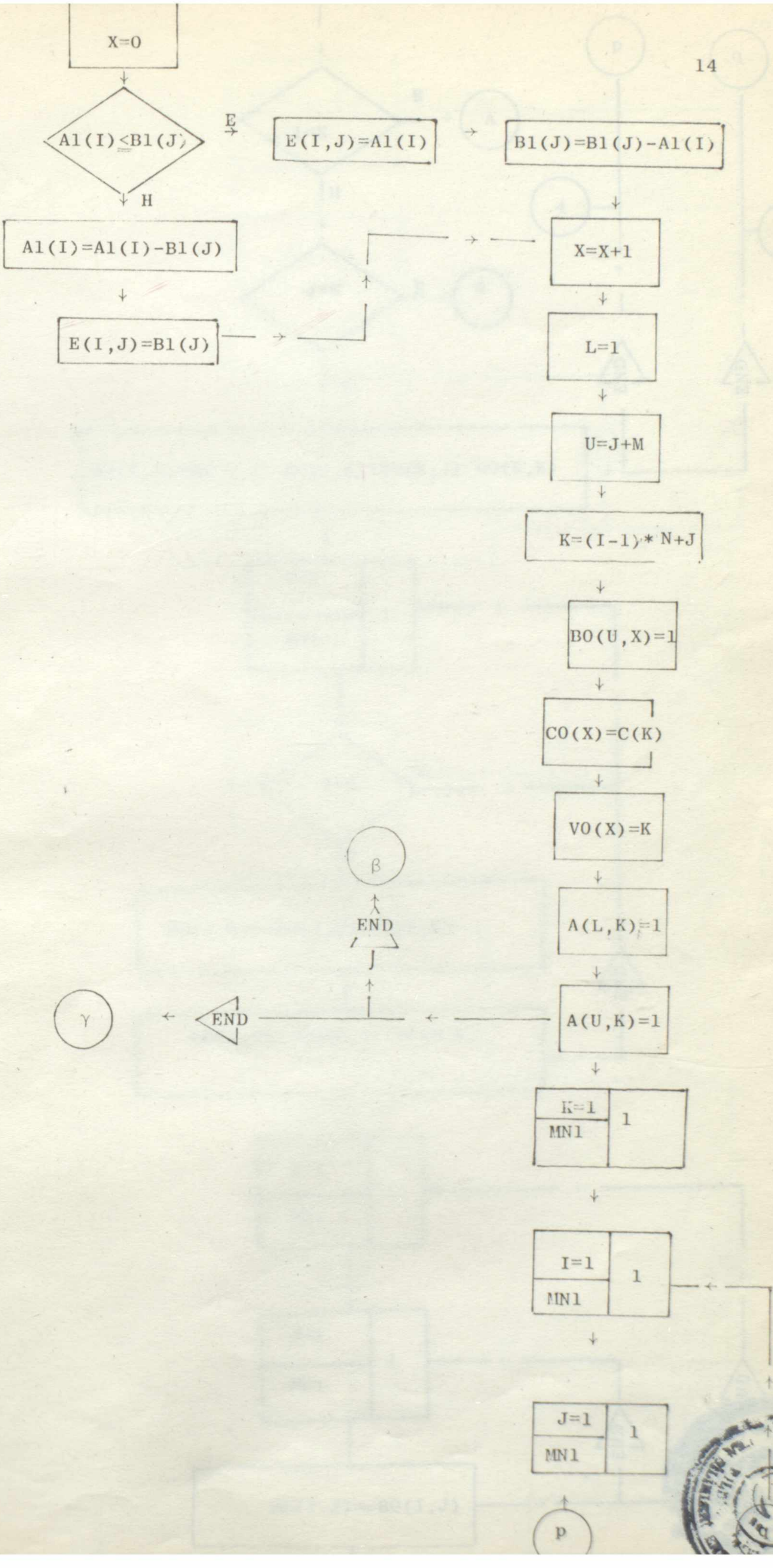
$$V_o^* = B_o^{-1} V \longrightarrow V_o^* = \begin{bmatrix} 10 \\ 25 \\ 18 \\ 20 \\ 7 \\ 10 \end{bmatrix}.$$

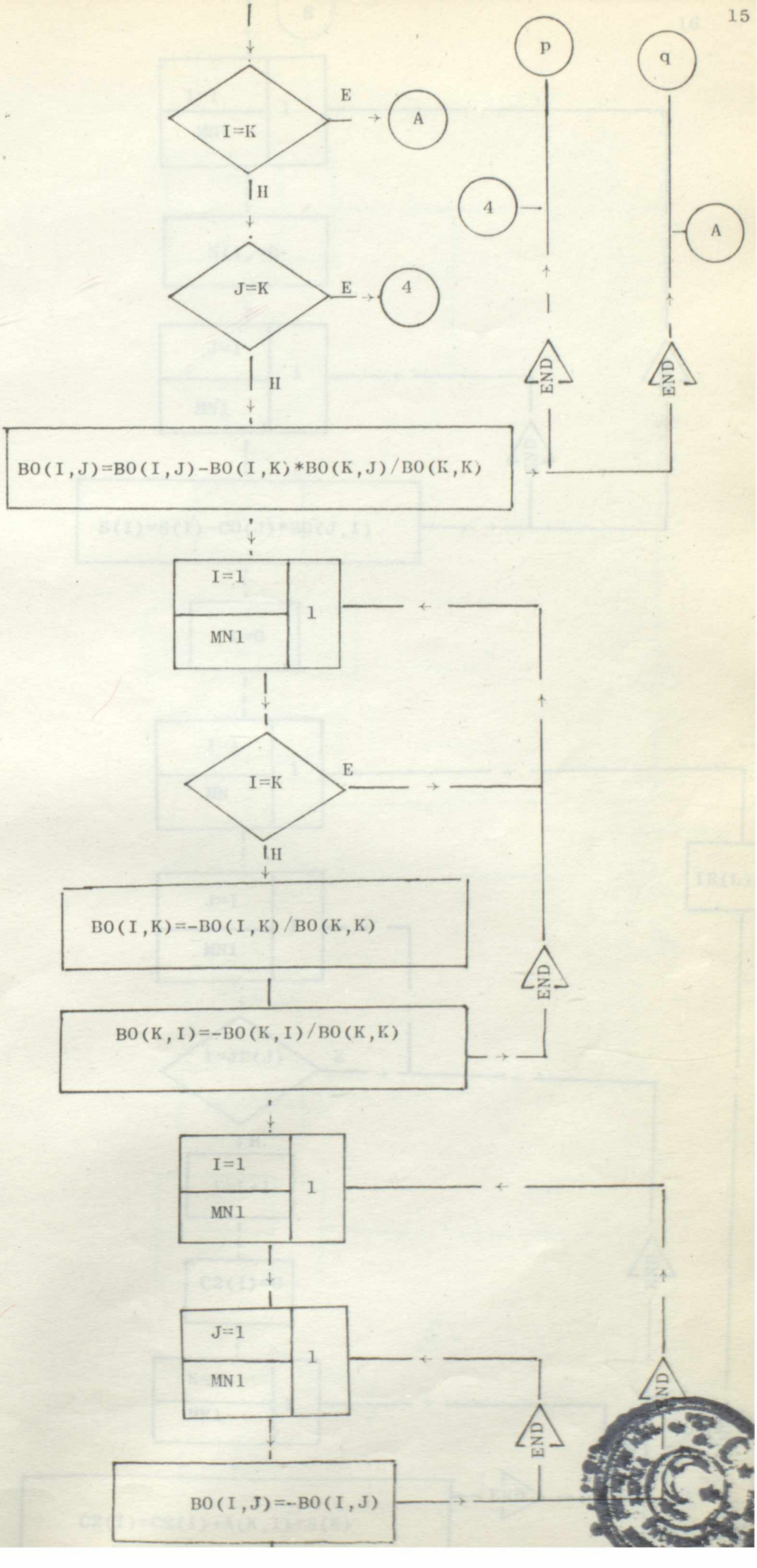
Optimal çözüm:

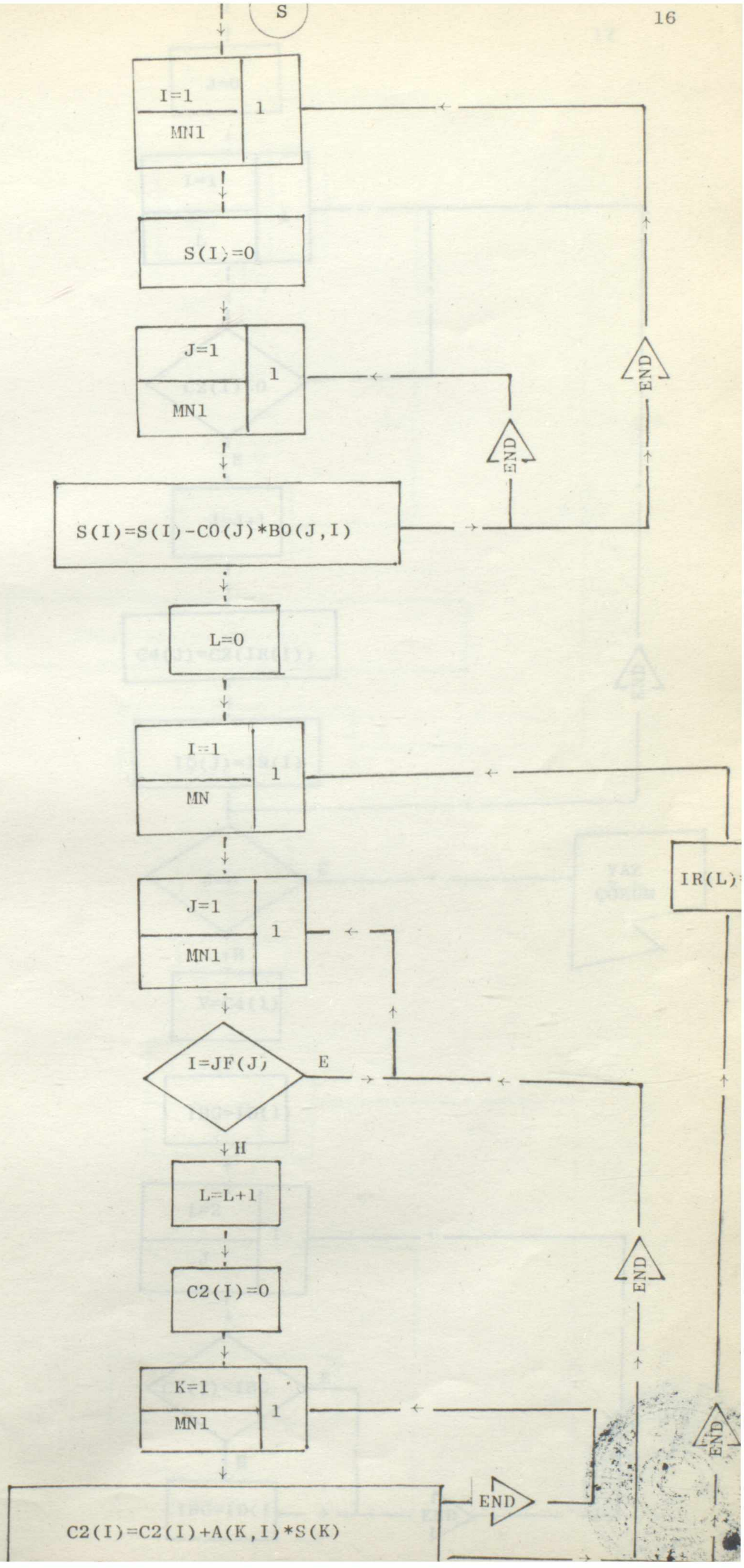
$x_{21} = 10$, $x_{14} = 25$, $x_{22} = 18$, $x_{33} = 20$, $x_{24} = 7$, $x_{34} = 10$;

Min $z = 310$.

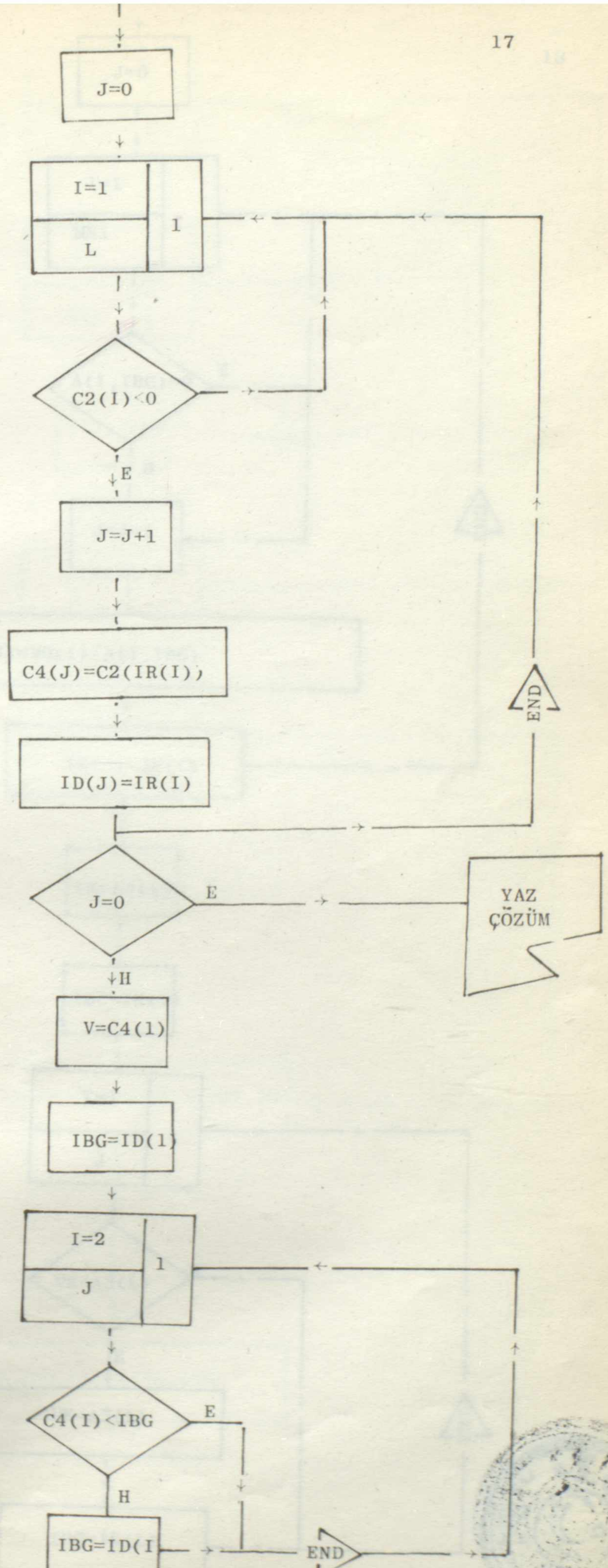

BAŞLA
OKU
M,N
MN=M*N
MN1=M+N-1
OKU
C(J),J=1,N, A1(I),I=1,M, B1(I),I=1,N
I=1
M
1
J=1
N
1
E(I,J)=0
END
I=1
M
1
γ
J=1
N
1
β


X=0
A1(I)≤B1(J)
E
E(I,J)=A1(I)
B1(J)=B1(J)-A1(I)
H
A1(I)=A1(I)-B1(J)
E(I,J)=B1(J)
X=X+1
L=1
U=J+M
K=(I-1)*N+J
BO(U,X)=1
CO(X)=C(K)
VO(X)=K
β
END
A(L,K)=1
γ
END
A(U,K)=1
K=1
MN1
1
I=1
MN1
1
J=1
MN1
1
p

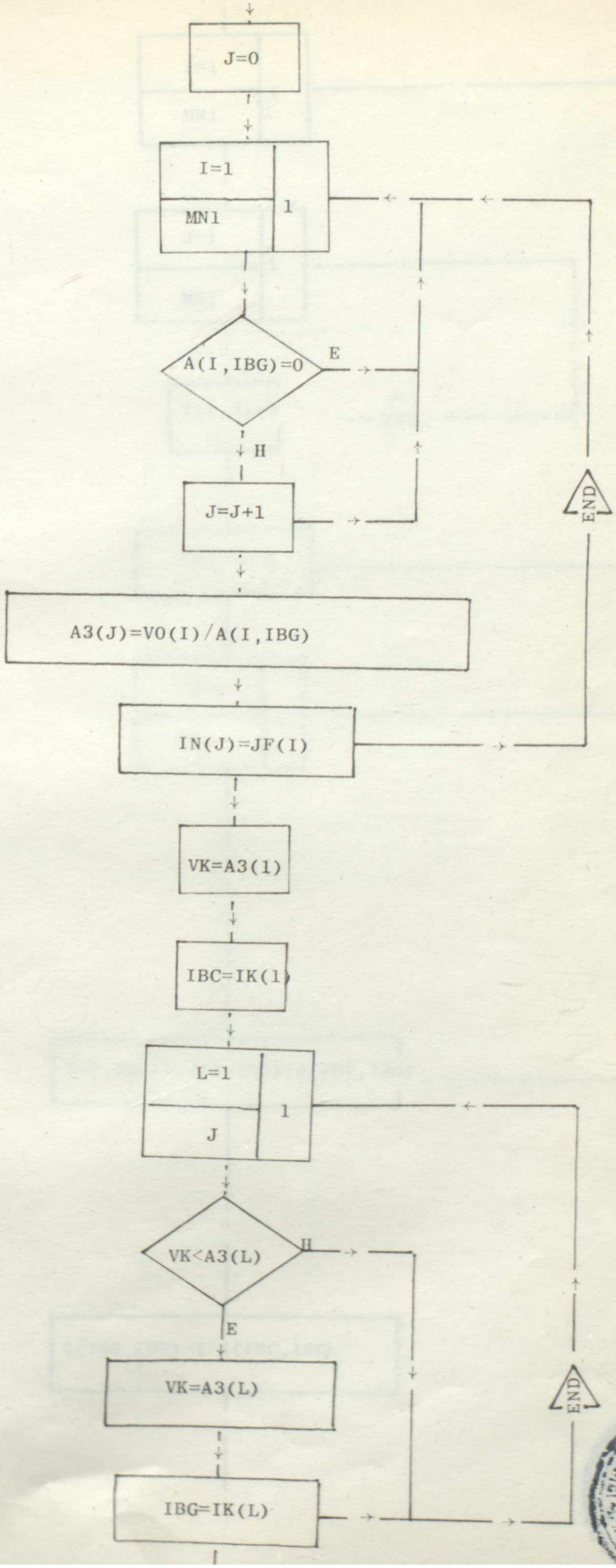

I=K
E
A
H
J=K
E
4
H
p
q
4
A
END
END
BO(I,J)=BO(I,J)-BO(I,K)*BO(K,J)/BO(K,K)
I=1
MN1
1
I=K
E
H
END
BO(I,K)=-BO(I,K)/BO(K,K)
BO(K,I)=-BO(K,I)/BO(K,K)
I=1
MN1
1
J=1
MN1
1
END
END
BO(I,J)=-BO(I,J)

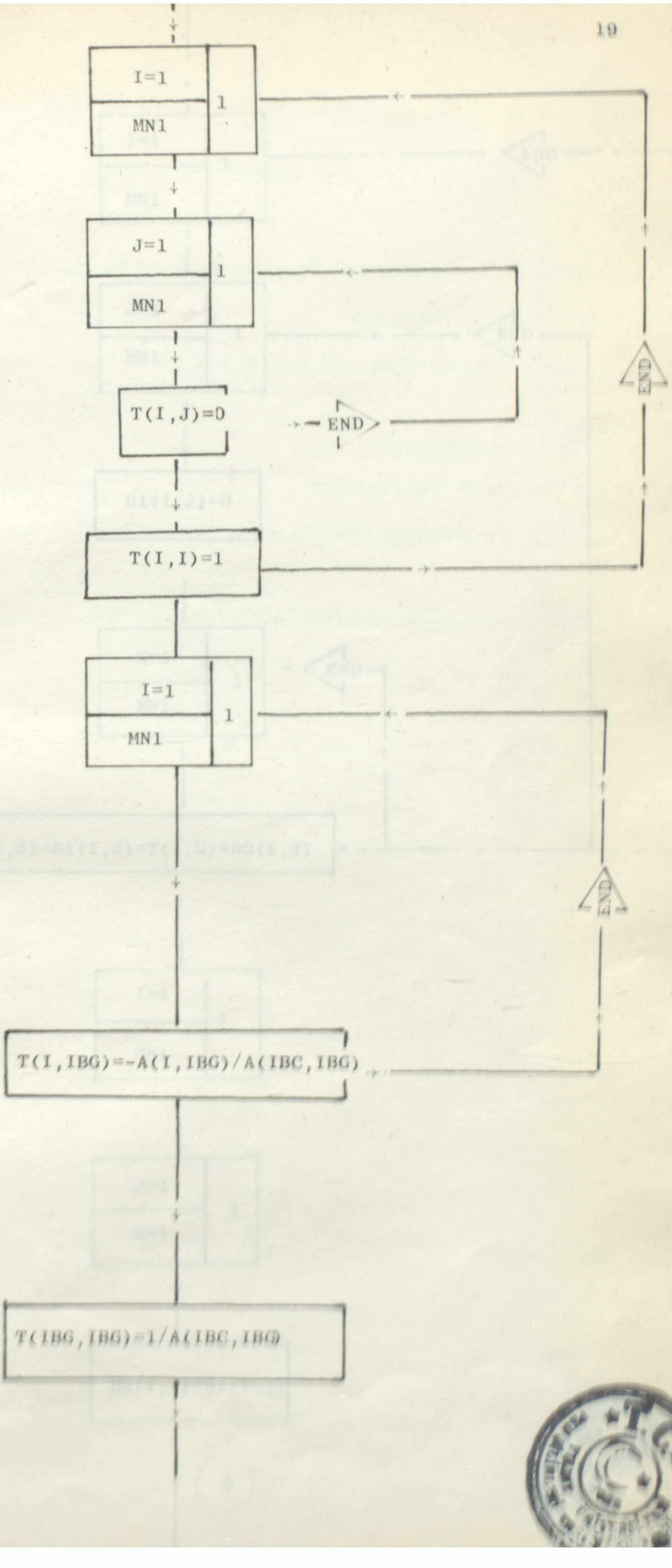


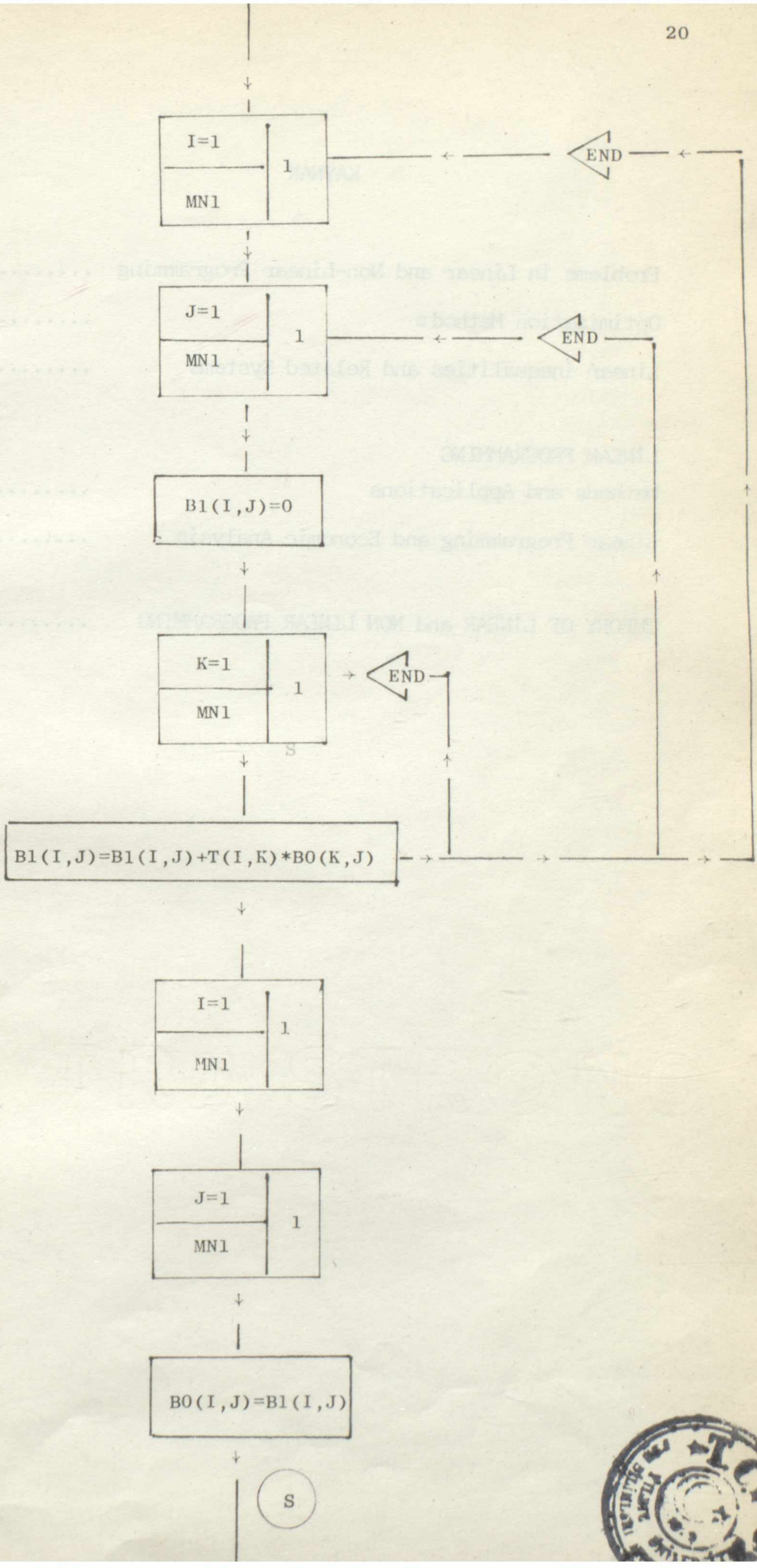
S
I=1
MN1
1
S(I)=0
J=1
MN1
1
END
END
S(I)=S(I)-C0(J)*B0(J,I)
L=0
I=1
MN
1
IR(L)
J=1
MN1
1
I=JF(J)
E
H
L=L+1
C2(I)=0
K=1
MN1
1
END
END
END
C2(I)=C2(I)+A(K,I)*S(K)



J=0
I=1
L
1
C2(I)<0
E
J=J+1
C4(J)=C2(IR(I))
ID(J)=IR(I)
END
J=0
E
YAZ
ÇÖZÜM
H
V=C4(1)
IBG=ID(1)
I=2
J
1
C4(I)<IBG
E
H
IBG=ID(I
END


J=0
I=1
MN1
1
A(I,IBG)=0
E
H
J=J+1
END
A3(J)=VO(I)/A(I,IBG)
IN(J)=JF(I)
VK=A3(1)
IBC=IK(1)
L=1
J
1
VK<A3(L)
H
E
VK=A3(L)
END
IBG=IK(L)


I=1
1
MN1
J=1
1
MN1
T(I,J)=0
END
END
T(I,I)=1
END
I=1
1
MN1
T(I,IBG)=-A(I,IBG)/A(IBC,IBG)
END
T(IBG,IBG)=1/A(IBC,IBG)



I=1
MN1
1
END
J=1
MN1
1
END
B1(I,J)=0
K=1
MN1
1
END
S
B1(I,J)=B1(I,J)+T(I,K)*B0(K,J)
I=1
MN1
1
J=1
MN1
1
B0(I,J)=B1(I,J)
S

KAYNAK

Problems in Linear and Non-Linear Programming .... S.Vajda

Optimization Methods .... K.V.Mital

Linear inequalities and Related Systems .... Kuhn,H.W.,and A.W.Tucker

LİNEAR PROGRAMMING
Methods and Applications .... SAUL I.GASS

Linear Programming and Economic Analysis .... DORFMAN SAMUELSON and SOLOW

THEORY OF LİNEAR and NON LINEAR PROGRAMMING .... S.VAJDA