

MATRIZENINVERSION

30.0

(Gleitkomma 24.1)

Funktion: Errechnet die Inverse einer quadratischen Matrix, deren Rang größer als 1 ist.

Eingabe: Die Elemente der Ausgangsmatrix sind zeilenweise in aufeinanderfolgenden Zellen, beginnend in Mo, gespeichert. Die Elemente müssen in der Standardform, die unter 24.1 II beschrieben ist, gegeben sein. Dem R.U.-Befehl muß ein Codewort folgen.

Ausgabe: Die Elemente der Inversen werden in aufeinanderfolgenden Zellen, beginnend in Mo, gespeichert. Sie haben die Standardform von 24.1 II.

Befehlsfolge:

a-1	xE	0000	
a	R	Lo + 14	
a+1	U	Lo	
a+2	n	Mo	(Codewort)
a+3	u.s.w.		

Die Zelle a-1 braucht nur dann den Befehl E 0000 zu enthalten, wenn die vorstehenden Operationen von System 24.1 interpretiert werden.

Codewort: Das Codewort gibt den Rang n der Matrix und die erste Speicherzelle Mo an. (n mit q=15, Mo mit q=29)

Zeit: Die Zeit T für die Inversion ist abhängig von n und ist näherungsweise durch folgende Formel gegeben:

$$T = 1,2 \cdot n^3$$

Speicher-
bedarf:

Die Routine benötigt 160 Speicherplätze für Befehle und Konstanten und 11 Spuren für den Hauptteil von 24.1 Die Matrix belegt $2n^2$ Speicher, beginnend mit Mo. Jedoch werden $2n$ weitere Zellen als Zwischenspeicher benutzt. Der Gesamtbedarf ist somit

$$864 + 2 (n^2 + n)$$

Allgemeines:

Die letzte Zelle des Planes 30.0 muß die Adresse von 24.1 in der Form

xZ tt ss

enthalten. Von 24.1 II werden nur die folgenden Befehle benutzt:

800Bxxxx, 800Dxxxx, 800Mxxxx, 800Hxxxx,
800Cxxxx, 800Axxxx, 800Sxxxx, U0000,
E0000.

30.0

D1-130.0

b0231'y0113'y0114'y0161'y0033'y0034'
y0121'y0122'y0162'y0208'y0209'b0014'
a0109'y0103'b0000'y0036'm0224'h0156'
s0207'h0222'a0106'h0107'n0222'm0221'
h0216'a0036'a0155'y0218'y0141'a0222'
y0037'y0210'y0111'r0000'u0000'800b0219'
800d0000'800h0000'xu0000'xe0000'b0036'
e0206'a0207'y0115'a0216'y0116's0111'
t0113'u0050'b0115'y0123'e0110's0218'
t0121'b0141'u0057'y0129'a0109'y0131'
u0227'y0130's0141't0129'b0222's0223'
h0222't0000'u0033'xz0004'xz0001'
xz0001'xs6363'm0000'xz0001'r0000'u0000'
800m0000'800e0000'xe0000'b0115'u0041''
r0000'u0000'800s0000'xu0000'xe0000'
b0141'y0163'u0137'b0000'c0000'b0000'
c0000'u0158'y0163'u0145'b0123'a0217'
y0200'u0142'b0000's0107'y0201'u0161'
e0110's0111't0149'u0151'b0200'u0138'
b0123's0217'y0211'u0208'xz0002''
b0109'a0131'u0057'r0000'u0000'800m0000'
800a0000'800h0000'xe0000'b0163'a0226'
u0134'xs6363'xz0002'r0000'u0000'800m0000'
800e0000'xe0000'b0156'a0123'u0051',0000011'
'8's0000'40000000'4'10000000'8'40000'
'8's0156'y0132's0108'u0061'00xz0000'.0000000'