# free42 Random Math Stuff

# Mitch Richling

# 2021-03-19

Author: Mitch Richling Updated: 2021-05-18 21:09:45

Copyright 2021 Mitch Richling. All rights reserved.

# Contents

1	Metadata	1
2	Introduction	1
3	NLA: Linear Algebra 3.1 Menu 3.2 Notes on individual programs 3.2.1 MXCPLY: Characteristic polynomial 3.2.2 EQCPLY: Characteristic polynomial as an MVAR program 3.3 Code for Menu 3.4 Functions	
4	POLY: A collection of polynomial tools  4.1 Menu 4.2 Notes for individual programs 4.2.1 PEVAL, PEVL1, & PEVL2: Evaluating Polynomials 4.2.2 PWRP: Wrapping a polynomial matrix in an MVAR function 4.2.3 PRIST & PRNXT: Finding the roots of a polynomial 4.2.4 PLGRR: Search for a polynomial root  4.3 Code for Menu 4.4 Local functions	5 6
5	VEC3: 3D Real Vector Application 5.1 Menu 5.2 Code for Menu 5.3 Local functions	15
6	STATR: Statistics Registers 6.1 Menu	
7	EOF	20

# 1 Metadata

The home for this HTML file is: https://richmit.github.io/hp42/math.html
A PDF version of this file may be found here: https://richmit.github.io/hp42/math.pdf
Files related to this document may be found on github: https://github.com/richmit/hp42
Directory contents:

src - The org-mode file that generated this HTML document
 src\_42s - Ready to convert source listings for 42s code in this document
 docs - This html document and associated PDF

bin - Importable RAW program files

### 2 Introduction

This org-mode file collects together a handful of mathematical stuff I find useful. Note that in the past I had a collection of simple mathematical functions in this file. That stuff has moved to sfun.org.

Menu	Lab		Inputs	Output
IDEN	MXIDN	Create an nxn identity matrix	X: N	X: identity matrix
DIAG	MXDIAG	Create diagonal matrix with given elements	X: VEC	X: diagonal matrix
TR	MXTR	Compute the trace of a matrix	X: M	X: trace
CPLY	MXCPLY	Compute matrix Characteristic polynomial	X: M	X: polynomial
EDIT	LBL $96$			
MAT	LBL 98	Store/Recall Current CPOLY matrix		
X	LBL $97$	Store/Recall current value of "X"		
EVCP	EQCPLY	Evaluate Characteristic polynomial		

# 3 NLA: Linear Algebra

#### 3.1 Menu

#### 3.2 Notes on individual programs

#### 3.2.1 MXCPLY: Characteristic polynomial

MXCPLY uses the Faddeev-LeVerrier algorithm to compute the characteristic polynomial of a matrix. The polynomial is a matrix of coefficients suitable for use by the polynomials tools found later on this page.

One can find the eigenvalues of a matrix by using PRIST & PRNXT to solve the characteristic polynomial.

#### 3.2.2 EQCPLY: Characteristic polynomial as an MVAR program

The EQCPLY function is an MVAR function that directly computes values of the characteristic polynomial. It is horribly inefficient, but it can be used by the built in solver to find real eigenvalues or to plot the characteristic polynomial (See: pgmforfun.org).

## 3.3 Code for Menu

(MJR-generate-42-menu-code "NLA" 0 tbl 0 1 'stay 'up 'auto #'MJR-custom-gen-lab #'MJR-custom-gen-sub)

```
(NLA)
@@@@ DSC: Auto-generated menu program
LBL "NLA"
LBL 01
               @@@@ Page 1 of menu NLA
CLMENU
"IDEN"
KEY 1 XEQ "MXIDN"
"DIAG"
KEY 2 XEQ "MXDIAG"
"TR"
KEY 3 XEQ "MXTR"
"CPLY"
KEY 4 XEQ "MXCPLY"
"EDIT"
KEY 6 XEQ 96
KEY 7 GTO 02
KEY 8 GTO 02
KEY 9 GTO 00
MENU
STOP
GTO 01
LBL 02
               0000 Page 2 of menu NLA
CLMENU
"MAT"
KEY 1 XEQ 98
"X"
KEY 2 XEQ 97
"EVCP"
KEY 6 XEQ "EQCPLY"
KEY 7 GTO 01
KEY 8 GTO 01
KEY 9 GTO 00
MENU
STOP
GTO 02
LBL 00 @@@@ Application Exit
EXITALL
RTN
0000 Free labels start at: 3
```

#### 3.4 Functions

```
(MXCPLY)
@@@@ DSC: Compute matrix Characteristic polynomial
0000 IN: X: Matrix
@@@@ OUT: X: Characteristic polynomial
@@@@ LBL: 28
@@@@ FAQ: Uses INDEX
@@@@ UPD: 2021-04-27
@@@@ TC: [[1,2,3][4,5,6][7,8,10]] \Rightarrow [1, -16, -12, 3]
LBL "MXCPLY"
FUNC 11
              @@## REQ:free42>=2.5.24
              @@## REQ:free42>=3.0
L4STK
LSTO "_A"
DIM?
XEQ "MXIDN"
LSTO "_M"
LSTO "_I"
R↓
1
+
1
X<>Y
NEWMAT
LSTO "_P"
INDEX "_P"
R↓
-1
ST0EL
.T+
+/-
LSTO "_CTR"
0
              0000 p_{n-1}
LBL 28
RCL "_A"
              @@@@ A
                                 p_{n-1}
RCL "_M"
              0000 M
                                         p_{n-1}
                                  Α
RCL "_I"
              0000 I
                                  М
                                          Α
                                                   p_{n-1}
                                                   p_{n-1}
RCL× ST T
              0000 I*p_{n-1}
                                 M
                                          Α
              0000 M-I*p_{n-1}
                                 Α
                                          p_{n-1}
              0000 A*(M-I*p_{n-1}) p_{n-1}
LSTO "_M"
XEQ "MXTR"
              0000 A*(M-I*p_{n-1}) p_{n-1}
RCL "_CTR"
              0000 p_n
                                 p_{n-1}
STOEL
ISG "_CTR"
NOP
J+
FC? 77
GTO 28
RCL "_P"
+/-
RTN
(MXTR)
0000 DSC: Compute matrix trace (sum of the diagonal elements)
0000 IN: X: Matrix
@@@@ OUT: X: trace
@@@@ FAQ: Dosen't use INDEX
@@@@ UPD: 2021-04-27
0000 TC: [[1,2,3][4,5,6][7,8,10]] \Rightarrow 16
LBL "MXTR"
              @@## REQ:free42>=2.5.24
FUNC 11
L4STK
              @@## REQ:free42>=3.0
LSTO "_M"
              @@@@ M
                             -- M is an nxn matrix
DIM?
              0000 n n M
1
              0000 1 n n M
              0000 1+n n M
DIM "_M"
              0000 1+n n M
                              -- M is now an nx(n+1) matrix with original diag in first column
1
              0000 1 1+n n M
              0000 1 1 1+n n M
1
NEWMAT
              0000 P 1+n n M -- P is a 1x1 zero matrix
STGN
              0000 P 1+n n M -- P is a 1x1 identity matrix
LSTO "_P"
R↓
              0000 1+n n M
                              -- P is a 1x1 matrix e_1
              0000 1 1+n n -- P is a 1x1 matrix e_1
1
```

```
X<>Y
                                           0000 1+n 1 n -- P is a 1x1 matrix e_1
DIM "_P"
                                                                                         -- P is now an 1x(1+n) e_1 row matrix
                                        0000 1+n 1
RCL "_P"
                                      0000 P 1+n 1
RCL "_M"
                                      0000 M P 1+n
                                       0000 X P 1+n -- X is now an (n+1)xn matrix with original diag in first row 0000 X 1+n 1 -- X is now an 1xn matrix with original diag in first row 0000 X 1+n 1 -- X is now a 1x1 matrix with the sum of the diag
TRANS
RSIIM
DET
                                        0000 TR
                                                                                          -- DET of a 1x1 matrix is matrix element
RTN
occorrections of the contraction of the contracti
                                                                                                                                                                                                                                                                                            (MXIDN)
0000 DSC: Create an XxX identity matrix
0000 IN: X: Size of matrix to make
0000 OUT: X: Identity matrix of size X
0000 FAQ: Dosen't use INDEX
@@@@ UPD: 2021-04-27
0000 REF: https://forum.swissmicros.com/viewtopic.php?f=19&t=2958
@@@@ FAQ: This code is longer, but easier to understand -- for me anyhow.
I.BI. "MXTDN"
FUNC 11
                                           @@## REQ:free42>=2.5.24
L4STK
                                          @@## REQ:free42>=3.0
1
NEWMAT
                                          0000 X is an nx1 zero matrix
                                           0000 X is now a constant matrix filled with 1s
STGN
XEQ "MXDIAG"
RTN
occorrections of the contraction of the contracti
                                                                                                                                                                                                                                                                                            (MXDIAG)
0000 DSC: Create diagonal matrix with given elements
0000 IN: X: matrix
0000 IN: X: diagonal matrix
@@@@ FAQ: Dosen't use INDEX
0000 FAQ: Uses all elements of X -- even if it is not 1xn or nx1
@@@@ UPD: 2021-04-27
0000 REF: https://forum.swissmicros.com/viewtopic.php?f=19&t=2958
\tt @QQQQ\ FAQ: This code is longer, but easier to understand -- for me anyhow.
LBL "MXDIAG"
                                       @@## REQ:free42>=2.5.24
FUNC 11
                                       @@## REQ:free42>=3.0
L4STK
LSTO "_M"
                                         @@@@ D
                                        0000 n m
DIM?
                                         @@@@ N
1
                                         0000 1 N
X=Y?
GTO 23
                                         0000 non 1x1 case
RCL+ ST Y
                                         0000 N+1 N
                                         0000 N N+1
X<>Y
DIM "_M"
                                          @@@@ N N+1
                                                                                      -- M is now an (N+1)xN matrix with D on first row
RCL "_M"
                                          0000 M N N+1
TRANS
                                          0000 M N N+1
                                                                                       -- M is now an Nx(N+1) matrix with D on first column
STO "_M"
                                           0000 1 N+1 N
R.J
ENTER
DIM "_M"
                                         0000 1 N+1 N -- M is now an NxN matrix with D on the diagonal
                                          0000 1 N+1 N -- due to the resize reshuffle
LBL 23
                                           000 All done. Return
RCL "_M"
RTN
(EQCPLY)
0000 DSC: Evaluate Chararstic Polynomial of a Matrix: DET(M-X*I)
@@@@ I/O: N/A MVAR program
0000 VAR: CPM a square matrix
0000 VAR: X a real or complex number
@@@@ LAB: 24-25
0000 FAQ: Can be used
0000 FAQ: Dosen't use INDEX
@@@@ UPD: 2021-04-27
LBL "EQCPLY"
MVAR "CPM"
MVAR "X"
RCL "CPM'
RCL "X"
RCL "CPM"
DIM?
```

```
R↓
XEQ "MXIDN"
×
-
DET
RTN
```

#### 

0000 Store/Recall variable "CPM"

LBL 98 FS? 64 RCL "CPM" STO "CPM"

RTN

RTN

0000 Store/Recall variable "X"

LBL 97
FS? 64
RCL "X"
STO "X"

@@@@ Edit matrix

LBL 96
FUNC 11
EDIT
"Enter data; R/S"
|-- to end"
PROMPT
EXITALL
RTN

END

# 4 POLY: A collection of polynomial tools

#### 4.1 Menu

Menu	$_{ m LBL}$	Description	Inputs	Output
NEW	NEWPLY	Create a polynomial of degree X	X: N	X: P
INTRP	PINTRP	Create interpolateing polynomial	Y: XDAT X: YDAT	X: P
EDIT	LBL 78	Edit a polynomial	X: P	X:P
VIEW	VPOLY	View a polynomial	X: P	N/A
SLV2	P2SLV	Solve quadratic polynomial	X: P	Y: root_1 X: root_2
SLV1	P1SLV	Solve linear polynomial	X: P	X: root
R1ST	PR1ST	Find a root	X: P	Z: OPoly Y: DPoly X: root
RNXT	PRNXT	Find next root	Z: OPoly Y: DPoly X: GUESS	Z: OPoly Y: DPoly X: root
VIEW	VPOLY	View the coefficients of a polynomial	X: P	N/A
DFALT	PDEFLT	Deflate polynomial	Y: P X: R	Y: Remainder X: P/(X-R)
EVAL	PEVAL	Evaluate polynomial P @ X	Y: P X: X	X: P(X)
EVAL1	PEVL1	Evaluate polynomial P & P' @ X	Y: P X: X	Y: P(X) X: P'(x)
EVAL2	PEVL2	Evaluate polynomial P, P', & P" @ X	Y: P X: X	Z: P(X) Y: P''(x) Y: P'(x)
LGRR	PLGRR	Root search (Laguerre's Method)	T: P Z: GUESS Y: ACC X: MAXITR	Z: Status y: P_Val X: root
VIEW	VPOLY	View the coefficients of a polynomial	X: P	N/A
POLY	LBL 98	Store/Recall Current Polynomial		
X	LBL $97$	Store/Recall current value of "X"		
EVAL	PWRP	Evaluate wrapped polynomial at X	N/A	X: P("X")

A polynomial is represented as 1xn matrix of coefficients. The first element of the matrix is the coefficient on the highest degree.

# 4.2 Notes for individual programs

# 4.2.1 PEVAL, PEVL1, & PEVL2: Evaluating Polynomials

These functions efficiently evaluate a polynomial (and its first and/or second derivative). They are handy for simply evaluating a polynomial repeatedly; however, they are more tuned for use as subroutines in other programs – ex: solvers. Note that the last page of the main menu provides

a more efficient way to repeatedly evaluate a polynomial.

#### 4.2.2 PWRP: Wrapping a polynomial matrix in an MVAR function

Simply store the polynomial into the global variable "WRPP", and then feed PWRP to things like the built in solver/integrator and similar tools (See: pgmforfun.org).

#### 4.2.3 PRIST & PRNXT: Finding the roots of a polynomial

These two programs provide a way to find all the roots of a polynomial. They work on real or complex polynomials, and finds both real and complex roots.

These functions use the global variable ACC to specify how close to zero the polynomial must be to accept a root. If ACC is not set, then 1e-15 is used.

The first function, PR1ST, is used to find an initial root of a polynomial. It only takes a polynomial. It will almost always find a root; however, it is possible for it to fail and return an error. When it fails, I suggest running the function again to see if it will find a root – it uses a random initial guess each time it runs. When it finds a root, it returns the original polynomial, the polynomial with the located root removed (deflated), and a root. This output is precisely what is needed to find more roots.

The second function, PRNXT, finds the next root of the polynomial. It requires three arguments (original polynomial, deflated polynomial, and a guess). This is precisely what the PRIST function returns. PRNXT also returns the original polynomial, the polynomial with the located root removed, and a new root. So you can feed PRNXT the return of PRIST or PRNXT.

A common question: Why is the original polynomial required by PRNXT, and not just the deflated one? A series of polynomial deflations leads to a deflated polynomial with some round off error. So wen PRNXT finds a root of the deflated polynomial, it then uses that root as the initial guess to PLGRR on the original polynomial. This significantly reduces round-off error, and almost always works – it is possible that it may converge to a root we already found. Currently PRNXT dosen't check for this case – that is on my todo list.

In summary, to find all the roots of a polynomial: Put the polynomial on the stack, and press PRIST to get the first root. Then hit PRNXT until you have found all the roots.

Alternately, with a bit more round off error, you can just repeatedly use PR1ST on the deflated polynomial that PR1ST returns.

#### 4.2.4 PLGRR: Search for a polynomial root

This is designed to be used by other programs. It takes a polynomial, a guess, a tolerance, and a maximum number of iterations. If the tolerance is negative, then the function will always preform the maximum number of iterations. This is useful for "refining" a root.

#### 4.3 Code for Menu

"DFALT"

(MJR-generate-42-menu-code "POLY" 0 tbl 0 1 'stay 'up 'auto #'MJR-custom-gen-lab #'MJR-custom-gen-sub)

```
(POLY)
@@@@ DSC: Auto-generated menu program
LBL "POLY"
LBL 01
                0000 Page 1 of menu POLY
CLMENU
"NEW"
KEY 1 XEQ "NEWPLY"
"INTRP"
KEY 2 XEQ "PINTRP"
"EDIT"
KEY 5 XEQ 78
"VIEW"
KEY 6 XEQ "VPOLY"
KEY 7 GTO 04
KEY 8 GTO 02
KEY 9 GTO 00
MENU
STOP
GTO 01
LBL 02
                0000 Page 2 of menu POLY
CLMENU.
"SI.V2"
KEY 1 XEQ "P2SLV"
"SLV1"
KEY 2 XEQ "P1SLV"
"R1ST"
KEY 3 XEQ "PR1ST"
"RNXT"
KEY 4 XEQ "PRNXT"
"VIEW"
KEY 6 XEQ "VPOLY"
KEY 7 GTO 01
KEY 8 GTO 03
KEY 9 GTO 00
MENU
STOP
GTO 02
LBL 03
                0000 Page 3 of menu POLY
CLMENU
```

```
KEY 1 XEQ "PDEFLT"
"EVAL"
KEY 2 XEQ "PEVAL"
"EVAL1"
KEY 3 XEQ "PEVL1"
"EVAL2"
KEY 4 XEQ "PEVL2"
"LGRR"
KEY 5 XEQ "PLGRR"
"VIEW"
KEY 6 XEQ "VPOLY"
KEY 7 GTO 02
KEY 8 GTO 04
KEY 9 GTO 00
MENU
STOP
GTO 03
LBL 04
                 @@@@ Page 4 of menu POLY
CLMENU
"POLY"
KEY 1 XEQ 98
"X"
KEY 2 XEQ 97
"EVAL"
KEY 6 XEQ "PWRP"
KEY 7 GTO 03
KEY 8 GTO 01
KEY 9 GTO 00
MENU
STOP
GTO 04
LBL 00 @@@@ Application Exit
EXITALL
RTN
0000 Free labels start at: 5
4.4 Local functions
\tt QQQQ\ DSC\colon Create\ an\ interpolateing\ polynomial
0000 IN: Y: X data matrix
രരരര
         X: Y data matrix
@@@@ OUT: X: interpolateing polynomial
@@@@ TST: free42_3.0.2.2
@@@@ UPD: 2021-04-28
@@@@ FAQ: X & Y must have the same number of elements, but dimintions may differ.
@@@@ FAQ: Explicitly constructs the vandermonde matrix, and solves the system
@@@@ FAQ: Uses INDEX
0000 TC: xdat:[1, 2, 3, 4] ydat:[1, -1, 1, -1] => [-4/3 10 -68/3 15] = [-1.33.. 10 -22.66.. 15]
@@@@ TC: xdat:[-1, 0, 1, 2] ydat:[-2, 3, -24, -77] => [1, -16, -12, 3]
LBL "PINTRP"
FUNC 21
               @@## REQ:free42>=2.5.24
L4STK
               @@## REQ:free42>=3.0
LSTO "_YDAT"
              @@@@ YDAT XDAT
DIM?
1
DIM "_YDAT"
               0000 1 N XDAT -- YDAT is now an Nx1 matrix
R↓
               @@@@ N XDAT
R↓
               @@@@ XDAT
LSTO "_XDAT"
               @@@@ XDAT
XEQ "MXDIAG"
               @@@@ MUL
                             -- nxn diag matrix
LSTO "_MUL"
               @@@@ MUL
DIM?
               @@@@ N N
R↓
               @@@@ 1 N
1
NEWMAT
               0000 TPL
                             -- TPL is an nx1 zero matrix
                             -- TPL is now an NX1 1 matrix
SIGN
               0000 TPL
LSTO "_TPL"
               @@@@ TPL
DIM?
               0000 1 N
R.↓
               0000 N
ENTER
               @@@@ N N
NEWMAT
               0000 VM
                             -- VM is an NXN zero matrix
LSTO "_VM"
               0000 VM
DIM?
               0000 N N
               0000 N
R.J
```

```
1
              0000 1 N
X<>Y
              0000 N 1
STOIJ
LBL 79
RCL "_TPL"
PUTM
RCL "_MUL"
X<>Y
STO "_TPL"
.T-
FC? 77
GTO 79
RCL "_YDAT"
RCL÷ "_VM"
             0000 YDAT
             @@@@ POLY
RTN
@@@@ DSC: Create a new polynomial of degree X
0000 IN: X: degree
0000 OUT: X: polynomial
@@@@ TST: free42_3.0.2
@@@@ UPD: 2021-04-26
LBL "NEWPLY"
             @@## REQ:free42>=2.5.24
FUNC 11
L4STK
            @@## REQ:free42>=3.0
1
X<>Y
1
+
NEWMAT
RTN
{\tt QQQQQ}\ {\tt DSC}\colon \ {\tt Edit}\ {\tt a}\ {\tt polynomial}\ {\tt in}\ {\tt the}\ {\tt matrix}\ {\tt editor}
@@@@ IN: X: polynomial
@@@@ OUT: X: polynomial
0000 TST: free42_3.0.2
@@@@ UPD: 2021-04-26
LBL 78
FUNC 11
             @@## REQ:free42>=2.5.24
L4STK
             @@## REQ:free42>=3.0
EDIT
"Enter data; R/S"
⊢" to end"
PROMPT
EXTTALL.
RTN
0000 DSC: Solve quadratic polynomial
@@@@ IN: X: Polynomial matrix
@@@@ OUT: Y: root_1
രരരര
        X: root_2
@@@@ TST: free42_3.0.2
@@@@ FAQ: Uses INDEX
@@@@ UPD: 2021-04-26
LBL "P2SLV"
FUNC 12
             @@## REQ:free42>=2.5.24
L4STK
             @@## REQ:free42>=3.0
LSTO "_M"
R.↓
INDEX "_M"
WR.AP
LBL 77
RCLEL
J+
FC? 77
GTO 77
RCL ST Z
              0000 a
                                                                             a
             0000 |a|
ABS
                                                                             а
X=0?
RTNERR 3
R↓
              0000 a
                                                         b
                                      С
                                                                             a
2
              @@@@ 2
```

INDEX "\_VM"

```
0000 c
0000 c/(2a)
RCL× ST T
                                    С
                                                      b
R.↓
                                    b
        0000 b
0000 b/(2a)
0000 -B
RCL÷ ST T
                                  b
                                                      a
                                c/(2a)
c/(2a)
C
X<>Y
RCL÷ ST T
                                                      a
+/-
                                                      a
                                                      C
ENTER
                                   -B
X↑2
           @@@@ B^2
                                  -B
                                                     C
C
                                 -B
                                  -B
                                                     C
                                   -B
                                                     C
                                  √(B^2-2C)
                                                     -B
√(B^2-2C)
-B-√(B^2-2C)
                                                     -B
                                                     √(B^2-2C)
                                                     √(B^2-2C)
RCL+ ST Z
           0000 -B+\(\(\)(B^2-2C)
                                   -B - \sqrt{(B^2 - 2C)}
RTN
@@@@ DSC: Solve linear polynomial
@@@@ IN: X: Polynomial matrix
@@@@ OUT: X: root
@@@@ TST: free42 3.0.2
@@@@ FAQ: Uses INDEX
@@@@ UPD: 2021-04-26
LBL "P1SLV"
FUNC 11
            @@## REQ:free42>=2.5.24
L4STK
            @@## REQ:free42>=3.0
LSTO "_M"
R.↓
INDEX "_M"
WRAP
RCLEL
             0000 a_1
J+
RCLEL
           0000 a_0
+/-
RTN
0000 DSC: Evaluate a polynomial
\tt @@@@ IN: Y: Matrix with polynomial coefficients. DIM of 1xn, nx1, whatever...
0000 X: Value at which polynomial should be evaluated
0000 OUT: X: value of polynomial evaluated at X
0000 LBL: 91
0000 FAQ: Uses INDEX
@@@@ TST: free42_3.0.2
@@@@ UPD: 2021-04-03
LBL "PEVAL"
             @@## REQ:free42>=2.5.24
FUNC 21
L4STK
            @@## REQ:free42>=3.0
X<>Y
             0000 P X
LSTO "_M"
INDEX "_M"
WRAP
Ω
            @@@@ PV P X
LBL 91
          0000 PV*X P X
RCL× ST Z
RCLEL
            0000 Coef PV*X P X
             @@@@ PV=Coef+PV*X P X
J+
FC? 77
GTO 91
RTN
0000 DSC: Evaluate a polynomial and it's first derivative
@@@@ IN: Y: Matrix with polynomial coefficients. DIM of 1xn, nx1, whatever...
0000 X: Value at which polynomial & derivative should be evaluated
@@@@ OUT: Y: value of polynomial evaluated at X
\tt @@@@ X: value of polynomial's derivative evaluated at X
0000 LBL: 91
@@@@ FAQ: Uses INDEX
@@@@ TST: free42_3.0.2
@@@@ UPD: 2021-04-03
LBL "PEVL1"
FUNC 22
             @@## REQ:free42>=2.5.24
```

a

2a

2a

2a

2a

2a

C

-B

-B

```
L4STK
              @@## REQ:free42>=3.0
X<>Y
              @@@@ P
                        Х
LSTO "_M"
INDEX "_M"
R↓
              @@@@ X
WRAP
Ω
              @@@@ PV
                         Х
0
              @@@@ DV
                         PV
                                 Х
LBL 92
RCL× ST Z
             @@@@ DV*X
                         PV
                                 X
RCL+ ST Y
             @@@@ DV*X+PV PV
             X<>Y
RCL× ST Z
            @@@@ PV*X
                         DV*X+PV X
             @@@@ C
                         PV*X
                                 DV*X+PV X
RCLEL
             @@@@ C+PV*X DV*X+PV X
X<>Y
             @@@@ DV*X+PV C+PV*X X
.T+
FC? 77
GTO 92
RTN
0000 DSC: Evaluate a polynomial and it's first two derivatives
@@@@ IN: Y: Matrix with polynomial coefficients. DIM of 1xn, nx1, whatever...
        X: Value at which polynomial & derivative should be evaluated
0000 OUT: Z: value of polynomial evaluated at X
0000
        Y: value of polynomial's first derivative evaluated at X
0000
        X: value of polynomial's second derivative evaluated at X
@@@@ LBL: 91
@@@@ FAQ: Uses INDEX
@@@@ TST: free42_3.0.2
@@@@ UPD: 2021-04-03
I.BI. "PEVI.2"
              @@## REQ:free42>=2.5.24
FUNC 23
L4STK
              @@## REQ:free42>=3.0
LSTO "_X"
             @@@@ X
                        Р
R↓
              @@@@ P
LSTO "_M"
INDEX "_M"
R↓
              0000
WRAP
0
              0000 PV
              0000 DV
                          ΡV
0
0
              @@@@ DDV
                          DV
                                  P۷
LBL 93
RCL× "_X"
            @@@@ DDV*X
                        DV
                                  ΡV
RCL+ ST Y
           @@@@ DDV*X+DV DV
                       DDV*X+DV PV
X<>Y
             @@@@ DV
RCL× "_X"
             @@@@ DV*X
                          DDV*X+DV PV
             @@@@ DV*X+PV DDV*X+DV PV
RCL+ ST Z
X<>Y
             @@@@ DDV*X+DV DV*X+PV PV
RCL ST Z
            0000 PV
                          DDV*X+DV DV*X+PV PV
            @@@@ PV*X
                          DDV*X+DV DV*X+PV PV
RCL× "_X"
RCLEL
             @@@@ C
                          PV*X
                                DDV*X+DV DV*X+PV
             @@@@ C+PV*X DDV*X+DV DV*X+PV
            @@@@ C+PV*X DDV*X+DV DV*X+PV C+PV*X
STO ST T
R↓
             @@@@ DDV*X+DV DV*X+PV C+PV*X
T+
FC? 77
GTO 93
2
0000 DSC: Deflate polynomial
@@@@ IN: Y: Matrix with polynomial coefficients. DIM of 1xn, nx1, whatever...
0000
        X: Root
0000 OUT: Y: Remainder (a number)
0000 X: Deflated polynomial
@@@@ LBL: 96
@@@@ FAQ: Uses INDEX
@@@@ TST: free42_3.0.2
@@@@ UPD: 2021-04-03
LBL "PDEFLT"
FUNC 22
              @@## REQ:free42>=2.5.24
```

```
L4STK
             @@## REQ:free42>=3.0
REAL?
GTO 88
X<>Y
XEQ 89
             @@@@ MAT2C
X<>Y
LBL 88
X<>Y
             0000 P R
LSTO "_M"
INDEX "_M"
WRAP
             0000 R
R↓
+/-
0
             0000 LC R
LBL 96
          0000 LC*R R
0000 C LC*R R
RCL× ST Y
RCLEL
X<>Y
            0000 LC*R C R
            0000 C-LC*R R
STOEL
J+
FC? 77
GTO 96
           0000 REM
RCL "_M"
DIM?
            0000 m n REM
            0000 N REM
×
1
            0000 1 N REM
            @@@@ N-1 REM
            0000 1 N-1 REM
1
X<>Y
             0000 N-1 1 REM
DIM "_M"
R↓
            0000 REM
R↓
RCL "_M"
             0000 P REM
RTN
@@@@ DSC: Make matrix complex
@@@@ NAM: MAT2C 89
@@@@ IN: X: Matrix
0000 OUT: X: Matrix
@@@@ FAQ: Uses INDEX
@@@@ LBL: MAT2C
LBL 89
FUNC 11
L4STK
LSTO "_M"
INDEX "_M"
RCLEL
REAL?
GTO 87
R↓
RTN
LBL 87
R↓
ENTER
DIM?
NEWMAT
COMPLEX
RTN
@@@@ DSC: View elements of polynomial
0000 IN: X: polynomial matrix
0000 OUT: N/A
@@@@ TST: free42_3.0.2
@@@@ FAQ: Uses INDEX
@@@@ UPD: 2021-04-03
LBL "VPOLY"
FUNC 00
LSTO "_M"
INDEX "_M"
WRAP
DIM?
LBL 90
```

```
"X~"
AIP
⊢":"
RCLEL
ARCL ST X
R↓
AVIEW
STOP
1
J+
FC? 77
GTO 90
R.TN
@@@@ DSC: Find a root
@@@@ IN: X: Polynomial
@@@@ OUT: Z: Origional Polynomial
        Y: Deflated Polynomial
0000
         X: Root
@@@@ FAQ: If set, the global variable ACC is used to set accuracy
@@@@ TC: [1, -16, -12, 3] => -0.90574, 0.1982, 16.70749
LBL "PR1ST"
FUNC 13
L4STK
XEQ 81
              @@@@ PLYBAD
RTNERR 5
LSTO "_P"
              @@@@ P
RAN
R.A.N
COMPLEX
              0000 Guss Poly
SF 25
RCL "ACC"
FC?C 25
              0000 Tol Guss Poly
1e-15
              0000 Itr Tol Guss Poly
XEQ "PLGRR"
             @@@@ Root Pval Stat
0\neq? ST Z @0## TODO: Memory leak in free42 < 3.0.3
RTNERR 6
RCL "_P"
              @@@@ Poly Root Pval Stat
RCL "_P"
             @@@@ Poly Poly Root Pval
              0000 Root Poly Poly Root
RCL ST Z
              @@@@ DPly Rem Poly Root
XEQ "PDEFLT"
X<>Y
              0000 Rem DPly Poly Root
              @@@@ DPly Poly Root
R.J
RCL ST Z
              @@@@ Root DPly Poly Root
R.T.N
@@@@ DSC: Find a another/next root
0000 in: Z: Origional Polynomial
0000
        Y: Deflated Polynomial
0000
        X: Guess
0000 OUT: Z: Origional Polynomial
       Y: Deflated Polynomial or 0 if fully deflated
0000 FAQ: If set, the global variable ACC is used to set accuracy
LBL "PRNXT"
FUNC 33
L4STK
RCL ST Z
              @@@@ Poly Gues DPly Poly
LSTO "_P"
R↓
              @@@@ Gues DPly Poly
RCL ST Y
              @@@@ DPLY Gues DPly Poly
LSTO "_DP"
XEQ 81
              @@@@ PLYBAD
RTNERR 5
R↓
              @@@@ Gues DPly Poly
SF 25
RCL "ACC"
FC?C 25
1e-15
              @@@@ Tol Gues DPly Poly
              @@@@ Itr Tol Gues DPly
50
XEO "PLGRR"
              0000 Root Pval Stat
RCL ST Z
              @@@@ Root Pval Stat
                                                 QQ## REQ: free 42 < 3.0.3 QQ## TODO: Delete when DM42 >= free 42 3.0.3
```

```
X≠0?
                                                 @@## REQ:free42<3.0.3 @@## TODO: Delete when DM42 >= free42 3.0.3
0≠? ST Z
                                                 @@## REQ:free42>=3.0.3
RTNERR 6
              0000 Root Pval Stat
                                                @@## REQ:free42<3.0.3 @@## TODO: Delete when DM42 >= free42 3.0.3
RCL " P"
              @@@@ Poly Root Pval Stat
X<>Y
              @@@@ Root Poly Pval Stat
             @@@@ -1 Root Poly Pval
-1
             @@@@ 5 -1 Root Poly
5
XEQ "PLGRR"
             0000 Root Pval Stat
RCL ST X
             @@@@ Root Root Pval Stat
RCL "_DP"
              @@@@ DPly Root Root Pval
X<>Y
             @@@@ Root DPly Root Pval
XEQ "PDEFLT"
             0000 DPly Rem Root Pval
X<>Y
R↓
              @@@@ DPly Root Pval
@@@@ TODO: Should check if |Rem| is near zero. If it is not, then we probably converged to a previously found root and removed from DPI
@@@@ TODO: that we we might have diverged, but that is super unlikely. In fact, both cases are quite unlikely. Still good software sho
            @@@@ Poly DPly Root Pval
X<>Y
              @@@@ DPly Poly Root Pval
RCL ST Z
              @@@@ Root DPly Poly Root
RTN
{\tt QQQQQ} DSC: RTNNO if X is not a polynomial of degree>0
@@@@ NAM: PLYBAD 81
@@@@ IN: X: Polynomial
0000 OUT: N/A
LBL 81
FUNC 00
MAT?
GTO 82
RTNYES
LBL 82
DIM?
2
X>Y?
RTNYES
RTNNO
0000 DSC: Use Laguerre's method to find a polynomial root
0000 IN: T: Polynomial
0000
        Z: Guess
0000
        Y: Tolerance
        X: Maximum Iteration
രരരര
0000 OUT: Z: Reason for exit
        0 = A solution has been found.
രരരര
0000
           3 = Bad guess was used.
0000
        Y: P(X)
രരരര
        X: Root
LBL "PLGRR"
FUNC 43
L4STK
              @@@@ ITR TOL GUESS POLY
LSTO "_I"
R↓
              @@@@ TOL GUESS POLY
LSTO "_T"
R.↓
              @@@@ GUESS POLY
LSTO "_G"
R↓
              @@@@ POLY
LSTO "_P"
DIM?
              @@@@ N
LSTO "_N"
0000 TODO: Check N>1. Another status: 4 = polynomial is constant
              @@@@ N-1
LSTO "_NM1"
R↓
LBL 94
RCL "_P"
RCL "_G"
XEQ "PEVL2"
              0000 P''
                             P'
                                   Ρ
                                            ?
RCL ST Z
                             р,,
                                   P,
ABS
              0000 |P|
                                            Р
                                                @@## REQ:free42>=3.0.3
X<? "_T"
                                                @@## REQ:free42>=3.0.3
```

```
XEQ 76
                                         @@## REQ:free42<3.0.3 @@## TODO: Delete when DM42 >= free42 3.0.3
GTO 95
            0000 P''
                      P' P
R↓
@@@@ TODO: Check P' for /O. Another status: 5 = Iteration failed due to division by zero
RCL÷ ST Y @@@@ P''/P' P' P
R↓ @@@@ P' P ?
                                     ?
                                     P''/P'
           COCCC -N=P/P' ? P''/P' P''/P'
COCCC -N=P/P' ? P''/P' P''/P'
COCCC -N P''/P' P''/P' ?
COCCC -N P''/P' P''/P' ?
COCCC N P''/P' P''/P' ?
÷
X<>Y
          0000 ?
           - /r' P''/P'
0000 P''/P' N
0000 - /r' P''/P'
          @@@@ -N
R↓
           0000 N
+/-
                                     ?
X<>Y
                                     ?
p,,/p,
                                     ?
                         N P''/P'
                                     ?
                             P''/P' ?
1
           @@@@ 1-L*n/(n-1) N P'''/P' P'''/P'
SQRT
RCL× "_NM1"
RCL÷ " N"
RCL "_N"
1/X
1/X
            @@@@ LD P''/P' P''/P'
STO+ "_G"
DSE "_I"
GTO 94
0000 EXIT: Max iter
                          P''/P'
                                   ?
3
          0000 3
                     LD
         0000 POLY 3
0000 G POLY
0000 P 3
RCL "_P"
                            LD
RCL "_G"
                            3
XEQ "PEVAL"
RCL "_G"
         0000 G P
                           3
R.TN
          0000 |P| P''
LBL 95
                           P'
                                   Р
@@@@ EXIT: Found root
R↓ 0000 P''
                            Ρ
R↓
           @@@@ P'
                     Р
R↓
           0000 P
0
X<>Y
            @@@@ P
                     0
RCL "_G"
            0000 Root Pval
R.T.N
RCL "_T" @@## REQ:free42<3.0.3 @@## TODO: Delete when DM42 >= free42 3.0.3
X>Y?
      @@## REQ:free42<3.0.3 @@## TODO: Delete when DM42 >= free42 3.0.3
0000 DSC: Make a polynomial stored in a matrix into a =MVAR= function
0000 IN: X: N/A
0000 OUT: X: N/A
@@@@ GLB: WRPP -- Store a polynomial matrix in this variable
0000 TST: free42_3.0.2
0000 FAQ: Allows one to use SOLVER & INTEG on polynomials
@@@@ USE: PEVAL
@@@@ UPD: 2021-04-03
LBL "PWRP"
MVAR "X"
RCL "WRPP"
RCL "X"
XEQ "PEVAL"
RTN
0000 Store/Recall variable "WRPP"
LBL 98
FS? 64
RCL "WRPP"
STO "WRPP"
RTN
```

```
0000 Store/Recall variable "X"
LBL 97
FS? 64
RCL "X"
STO "X"
RTN
0000 DSC: Is a number very close to zero
@@@@ NAM: ZEROISH 80
LBL 80
FUNC 11
L4STK
ABS
1e-10
X>Y?
RTNYES
RTNNO
```

## 

F.ND

# 5 VEC3: 3D Real Vector Application

This is a simple little application that makes working with 3D vectors less painfull.

#### 5.1 Menu

Menu	Target	
→V	LBL 99	Put stack elements X, Y, & Z into a vector: [Z, Y, X]
$V\rightarrow$	LBL $98$	Vector contents to stack. [A, B, C] => X: C, Y: B, Z: A
DOT		Dot product
CROSS		Cross product
MAG	FNRM	Euculidian magnitude
VVIEW	LBL 96	View a vector one element at a time – press R/S for next element

# 5.2 Code for Menu

 $(\texttt{MJR-generate-42-menu-code} \ \texttt{"VEC3"} \ \texttt{0} \ \texttt{tbl} \ \texttt{0} \ \texttt{1} \ \texttt{'stay} \ \texttt{'up} \ \texttt{'auto} \ \texttt{\#'MJR-custom-gen-lab} \ \texttt{\#'MJR-custom-gen-sub})$ 

```
(VEC3)
@@@@ DSC: Auto-generated menu program
LBL "VEC3"
LBL 01
               @@@@ Page 1 of menu VEC3
CLMENU
"→\"
KEY 1 XEQ 99
"∀→"
KEY 2 XEQ 98
"DOT"
KEY 3 XEQ 02
"CROSS"
KEY 4 XEQ 03
"MAG"
KEY 5 XEQ 04
"VVIEW"
KEY 6 XEQ 96
KEY 9 GTO 00
MENU
STOP
GTO 01
LBL 00 @@@@ Application Exit
EXITALL
RTN
LBL 02
        0000 Action for menu key DOT
DOT
RTN
LBL 03
        0000 Action for menu key CROSS
CROSS
RTN
LBL 04
        0000 Action for menu key MAG
FNRM
RTN
0000 Free labels start at: 5
```

#### 5.3 Local functions

```
0000 DSC: Create a vector from stack contents
@@@@ NAM: →V 99
0000 IN: Z: real number
0000 Y: real number
@@@@ OUT: X: 1x3 matrix
0000 LBL: Used: 51
0000 FAQ: Uses INDEX
@@@@ TST: free42_3.0.2
@@@@ UPD: 2021-04-03
LBL 99
FUNC 31
XEQ 95
LSTO "_M"
R↓
INDEX "_M"
WRAP
J-
LBL 51
STOEL
R↓
J-
FC? 77
GTO 51
RCL "_M"
RTN
0000 DSC: Put vector elements on stack
@@@@ NAM: V→ 98
0000 IN: X: 1x3 matrix V
<code>@@@@</code> <code>OUT:</code> <code>Z:</code> First component of <code>V</code>
0000
        Y: Second component of V
രരരര
        X: Third component of V
@@@@ LBL: Used: 52
@@@@ FAQ: Uses INDEX
@@@@ TST: free42_3.0.2
@@@@ UPD: 2021-04-03
LBL 98
FUNC 13
LSTO "_M"
R↓
INDEX "_M"
WRAP
LBL 52
RCLEL
J+
FC? 77
GTO 52
RTN
0000 DSC: View elements of vector
0000 NAM: VVIEW 96
@@@@ IN: X: 1x3 matrix V
@@@@ OUT: N/A
0000 LBL: Used: 53
0000 FAQ: Uses INDEX
@@@@ TST: free42_3.0.2
@@@@ UPD: 2021-04-03
LBL 96
FUNC 00
LSTO "_M"
INDEX "_M"
WRAP
1
LBL 54
CLA
AIP
⊢":"
RCLEL
ARCL ST X
R↓
```

```
AVIEW
STOP
1
J+
FC? 77
GTO 54
RTN
0000 DSC: Make a 3D vector full of zeros
@@@@ NAM: VVIEW 95
0000 IN: N/A
@@@@ OUT: X: 1x3 Matrix
0000 LBL: Used: 53
@@@@ TST: free42_3.0.2
@@@@ UPD: 2021-04-03
LBL 95
FUNC 01
1
3
NEWMAT
```

END

# STATR: Statistics Registers

#### 6.1 Menu

Menu	Code
$\Sigma x$	FUNC 01; L4STK; $\Sigma$ REG?; 0; +; RCL IND ST X; " $\Sigma$ x="; ARCL ST X; AVIEW
$\Sigma \mathbf{x} \uparrow 2$	FUNC 01; L4STK; $\Sigma$ REG?; 1; +; RCL IND ST X; " $\Sigma$ x^2="; ARCL ST X; AVIEW
$\Sigma \mathbf{y}$	FUNC 01; L4STK; $\Sigma$ REG?; 2; +; RCL IND ST X; " $\Sigma$ y="; ARCL ST X; AVIEW
$\Sigma \mathbf{y} \uparrow 2$	FUNC 01; L4STK; $\Sigma$ REG?; 3; +; RCL IND ST X; " $\Sigma$ y $^2$ ="; ARCL ST X; AVIEW
$\Sigma xy$	FUNC 01; L4STK; $\Sigma$ REG?; 4; +; RCL IND ST X; " $\Sigma$ xy="; ARCL ST X; AVIEW
n	FUNC 01; L4STK; \( \text{SREG?}; 5; +; \) RCL IND ST X; \"n="; ARCL ST X; AVIEW
$\Sigma lnx$	FUNC 01; L4STK; ΣREG?; 6; +; RCL IND ST X; "Σlnx="; ARCL ST X; AVIEW
$\Sigma lnx \uparrow 2$	FUNC 01; L4STK; $\Sigma$ REG?; 7; +; RCL IND ST X; " $\Sigma$ (lnx)^2="; ARCL ST X; AVIEW
$\Sigma$ lny	FUNC 01; L4STK; ΣREG?; 8; +; RCL IND ST X; "Σİny="; ARCL ST X; AVIEW
$\Sigma lny \uparrow 2$	FUNC 01; L4STK; $\Sigma$ REG?; 9; +; RCL IND ST X; " $\Sigma$ (lny)^2="; ARCL ST X; AVIEW
$\Sigma$ lnxlny	FUNC 01; L4STK; ΣREG?; 10; +; RCL IND ST X; "Σlnxlny="; ARCL ST X; AVIEW
$\Sigma$ xlny	FUNC 01; L4STK; \( \text{SREG?}; \) 11; +; RCL IND ST X; "\( \text{Sxlny=}"; \) ARCL ST X; AVIEW
$\Sigma$ ylnx	FUNC 01; L4STK; ΣREG?; 12; +; RCL IND ST X; "Σylnx="; ARCL ST X; AVIEW

# 6.2 Code

KEY 8 GTO 02

```
(MJR-generate-42-menu-code "STATR" 0 tbl 0 nil 'stay 'up 'auto
                          #'MJR-local-only-gen-lab
                           (lambda (atrg target row)
                             (cl-destructuring-bind (menu prog) row
                               (mapconcat #'string-trim-left
                                          (split-string prog ";") "\n"))))
```

(STATR)

```
0000 DSC: Auto-generated menu program
LBL "STATR"
LBL 01
                    @@@@ Page 1 of menu STATR
CLMENU
"\Sigma x"
KEY 1 XEQ 04
"∑x↑2"
KEY 2 XEQ 05
"\Sigmay"
KEY 3 XEQ 06
"\Sigmay\uparrow2"
KEY 4 XEQ 07
"\Sigmaxy"
KEY 5 XEQ 08
KEY 6 XEQ 09
KEY 7 GTO 03
```

```
KEY 9 GTO 00
MENU
STOP
GTO 01
LBL 02
                       @@@@ Page 2 of menu STATR
CLMENU
\texttt{"}\Sigma \texttt{lnx"}
KEY 1 XEQ 10
\texttt{"}\Sigma \texttt{lnx} \texttt{\uparrow} \texttt{2"}
KEY 2 XEQ 11
\texttt{"}\Sigma \texttt{lny"}
KEY 3 XEQ 12
"\Sigmalny\uparrow2"
KEY 4 XEQ 13
"\Sigmalnxlny"
KEY 5 XEQ 14
\texttt{"}\Sigma \texttt{xlny"}
KEY 6 XEQ 15
KEY 7 GTO 01
KEY 8 GTO 03
KEY 9 GTO 00
MENU
STOP
GTO 02
LBL 03
                       @@@@ Page 3 of menu STATR
CLMENU
"\Sigmay\lnx"
KEY 1 XEQ 16
KEY 7 GTO 02
KEY 8 GTO 01
KEY 9 GTO 00
MENU
STOP
GTO 03
LBL 00 @@@@ Application Exit
EXITALL
RTN
LBL 04
             0000 Action for menu key \Sigma \mathbf{x}
FUNC 01
L4STK
\SigmaREG?
0
RCL IND ST X
"∑x="
ARCL ST X
AVIEW
RTN
LBL 05
             0000 Action for menu key \Sigma x \uparrow 2
FUNC 01
L4STK
\SigmaREG?
1
RCL IND ST X
"\sum_x^2="
ARCL ST X
AVIEW
RTN
LBL 06
             0000 Action for menu key \Sigma \mathbf{y}
FUNC 01
L4STK
\SigmaREG?
2
RCL IND ST X
"∑y="
ARCL ST X
AVIEW
RTN
LBL 07
             0000 Action for menu key \Sigma y \uparrow 2
FUNC 01
L4STK
\SigmaREG?
3
```

```
RCL IND ST X
"∑y^2="
ARCL ST X
AVIEW
RTN
LBL 08
            0000 Action for menu key \Sigma xy
FUNC 01
L4STK
\SigmaREG?
4
RCL IND ST X
"∑xy="
ARCL ST X
AVIEW
RTN
LBL 09
            0000 Action for menu key n
FUNC 01
L4STK
\SigmaREG?
5
RCL IND ST X
"n="
ARCL ST X
AVIEW
RTN
LBL 10
           0000 Action for menu key \Sigma \ln x
FUNC 01
L4STK
\Sigma \mathtt{REG?}
6
RCL IND ST X
"\Sigmalnx="
ARCL ST X
AVIEW
RTN
LBL 11
            0000 Action for menu key \Sigma \ln x \uparrow 2
FUNC 01
L4STK
\SigmaREG?
7
RCL IND ST X
"\Sigma(\ln x)^2="
ARCL ST X
AVIEW
RTN
LBL 12
           0000 Action for menu key \Sigmalny
FUNC 01
L4STK
\SigmaREG?
RCL IND ST X
"\Sigmalny="
ARCL ST X
AVIEW
RTN
LBL 13
            0000 Action for menu key \Sigma \ln y \uparrow 2
FUNC 01
L4STK
\SigmaREG?
9
RCL IND ST X
"\Sigma(lny)^2="
ARCL ST X
AVIEW
RTN
            0000 Action for menu key \Sigmalnxlny
LBL 14
FUNC 01
L4STK
\SigmaREG?
10
```

```
RCL IND ST X
\texttt{"}\Sigma \texttt{lnxlny="}
ARCL ST X
AVIEW
RTN
LBL 15
             0000 Action for menu key \Sigma \mathtt{xlny}
FUNC 01
L4STK
\SigmaREG?
11
RCL IND ST X "\Sigmaxlny="
ARCL ST X
AVIEW
RTN
LBL 16
             0000 Action for menu key \Sigma {\rm yln} {\rm x}
FUNC 01
L4STK
\SigmaREG?
12
RCL IND ST X
"\Sigmaylnx="
ARCL ST X
AVIEW
0000 Free labels start at: 17
```

7 EOF