

Ternarylogic LLC

Presentations and programs HELP Page

Getting Started













One of the most frustrating aspects of getting started is, well.....getting started.

Everything will appear to be "obvious" later, in hindsight, after you worked with the programs for a while.

However, if you are not familiar with programs like Freemat or Matlab®, the barriers to start may appear to be too high. They are really not. Please take some time to work or struggle through the following steps to get started.

The downloaded file

After placing an order and payment you are provided with a link to download a zipped file: Ternarylogic_Package.zip which contains the presentation and Program files. A listing of the files is provided below.

Name
 Help Pages for TernaryLogic Presentations and Programs.pdf
 LablansMVLTutorialISMVL2015.pdf
 Operator Notation and Matlab.pdf
 PLEASE READ THIS.pdf
 portfolioNov282014.pdf
 Program_Listing_Ternarylogic.pdf
 Ternarylogic LLC License Agreement.pdf
 TernaryPrograms.exe
 TernaryPrograms.zip
 The_Logic_of_More_the_book.pdf
 The_Logic_of_More_The_Presentations.pdf
 The_Logic_of_More_TOC_presentations.pdf

Extract the files, for instance to the directory TernarylogicPackageUnzip. Fill in the directory you wish to extract to.

The file Ternarylogic_Package.zip is password protected. Please use the password 'I accept the License Agreement' (without quotation marks) to unlock.

Matlab and Freemat

Matlab and Freemat are very simple, but powerful programming languages. However, they have their own idiosyncrasies. They do things their own way and they won't budge. If an error message is generated on screen, it means you most likely do something that is not allowed. Even if that is not readily apparent.

The purpose of this introduction is to get the Ternarylogic programs unpacked and installed in a directory; start Freemat; set the proper path; load a stored variable; run a script; and run a program requiring inputs. Once you have reached that stage, you can comfortably start working through the Non-binary programs and presentations.

Freemat

The steps are illustrated with Freemat. They equally apply to Matlab.

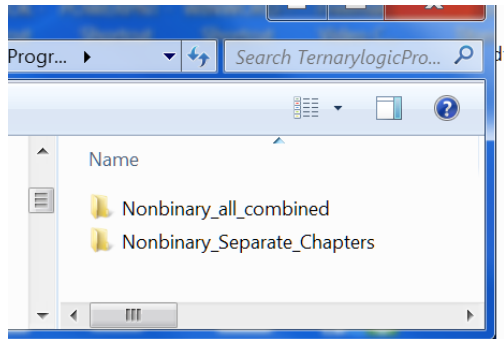
I. Install the Ternarylogic Lectures program files

The files that accompany the lectures are packed in zip files, which can be unpacked and extracted to a preferred directory. For the purpose of the explanation all files are unpacked to the directory /TernaryPrograms. You can of course designate your own directory.

The programs are packed in two ways: (1) as program files stored in individual directories, ranging from Lec1 to Lec24. (2) all files are stored in a single directory.

The following explanation is focused on unpacking or extracting in accordance with (1) to individual directories.

Among the files that you downloaded is also the zip file TernaryPrograms.zip which contains the Matlab/Freemat files.



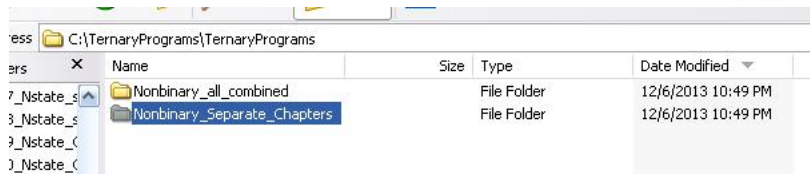
The zipped files that you downloaded can be unpack with your own unzipping facility. You can find an open source unzipping program 7-Zip at www.7-zip.org.

Also included is the self-extracting file TernaryPrograms.exe, which has the same files as TernaryPrograms.zip. Clicking on TernaryPrograms.exe will open the self-extracting facility:

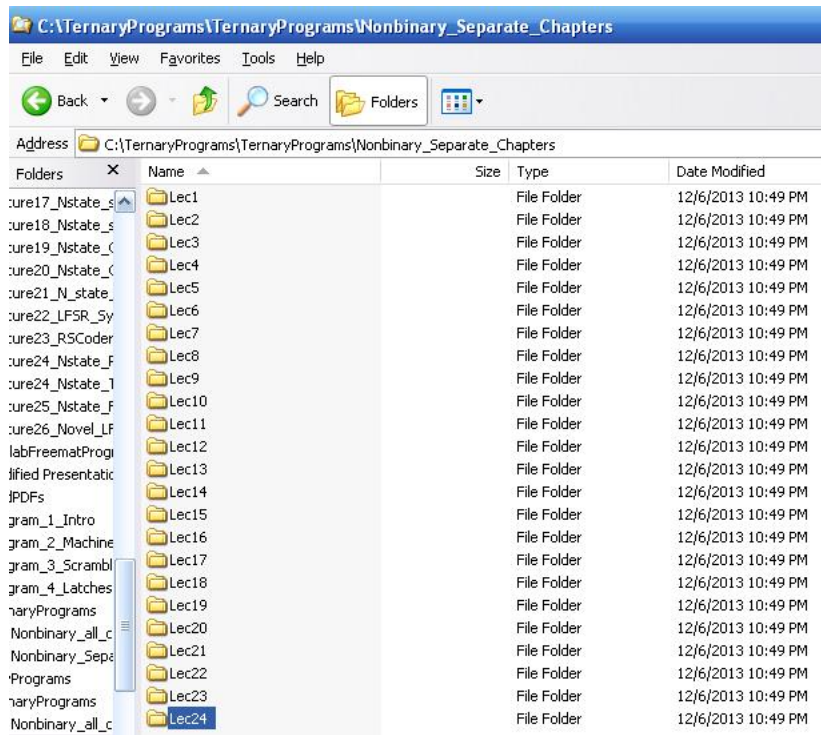


Enter the preferred directory in the "Extract to" window, or find your preferred directory by activating the ... button. Once the directory is selected or entered, activate the "Extract" button.

After unpacking, you will find the files in the designated directory.



We will focus on the directory Nonbinary_Separate_Chapters which has the sub-directories Lec1 ... Lec24.



We will focus on the sub-directory Lec24, with programs and data related to Lecture 24 in the PDF document.

Address		C:\TernaryPrograms\TernaryPrograms\Nonbinary_Separate_Chapters\Lec24			
Folders	Name	Size	Type	Date Modified	
Nonbinary_Sep	cfcdas.m	1 KB	MATLAB M-file	10/10/2013 8:54 PM	
Lec1	cfcsim.m	1 KB	MATLAB M-file	11/29/2013 12:14 PM	
Lec2	cfcsimm.m	2 KB	MATLAB M-file	11/29/2013 12:13 PM	
Lec3	cfcsimp.m	1 KB	MATLAB M-file	10/9/2013 8:19 PM	
Lec4	dec27.m	1 KB	MATLAB M-file	11/30/2013 6:17 PM	
Lec5	dec44.m	1 KB	MATLAB M-file	12/3/2013 8:04 PM	
Lec6	detransp.m	1 KB	MATLAB M-file	11/29/2013 3:10 PM	
Lec7	dscamdetransp.m	1 KB	MATLAB M-file	11/30/2013 11:36 PM	
Lec8	dscramtrp.m	1 KB	MATLAB M-file	12/1/2013 5:18 PM	
Lec9	m27.mat	1 KB	MATLAB MAT-file	12/1/2013 2:13 PM	
Lec10	mesc.mat	1 KB	MATLAB MAT-file	12/5/2013 10:03 PM	
Lec11	msd28.m	1 KB	MATLAB M-file	12/3/2013 8:06 PM	
Lec12	sc2t.m	1 KB	MATLAB M-file	11/30/2013 6:29 PM	
Lec13	sc27.mat	15 KB	MATLAB MAT-file	12/1/2013 2:12 PM	
Lec14	scramtransp.m	1 KB	MATLAB M-file	11/30/2013 7:10 PM	
Lec15	scramtranspsep.m	2 KB	MATLAB M-file	12/1/2013 5:03 PM	
Lec16	scrdescr28.m	1 KB	MATLAB M-file	12/3/2013 8:10 PM	
Lec17	sd28.mat	15 KB	MATLAB MAT-file	12/1/2013 2:14 PM	
Lec18	seq4.m	1 KB	MATLAB M-file	10/8/2013 8:42 PM	
Lec19	seq24s.m	1 KB	MATLAB M-file	2/3/2013 4:47 PM	
Lec20	seq42vec.m	1 KB	MATLAB M-file	12/3/2013 8:12 PM	
Lec21	seq44ssin.m	2 KB	MATLAB M-file	12/1/2013 4:43 PM	
Lec22	seq128.m	2 KB	MATLAB M-file	12/3/2013 8:16 PM	
Lec23	transp.m	1 KB	MATLAB M-file	11/29/2013 3:07 PM	

The directory has two types of extensions. The extension '.m' indicates an executable file. The extension '.mat' indicates a data file.

II. Freemat

We now move to installing and slightly configuring Freemat.

If you have not done so, download and install Freemat. If you have Matlab, work from Matlab. Then no installation of Freemat is required.

Freemat: sourceforge.net/#download

Home

FreeMat is a free environment for rapid engineering and scientific commercial systems such as MATLAB from Mathworks, and FreeMat is available under the GPL license.

Mac OS X

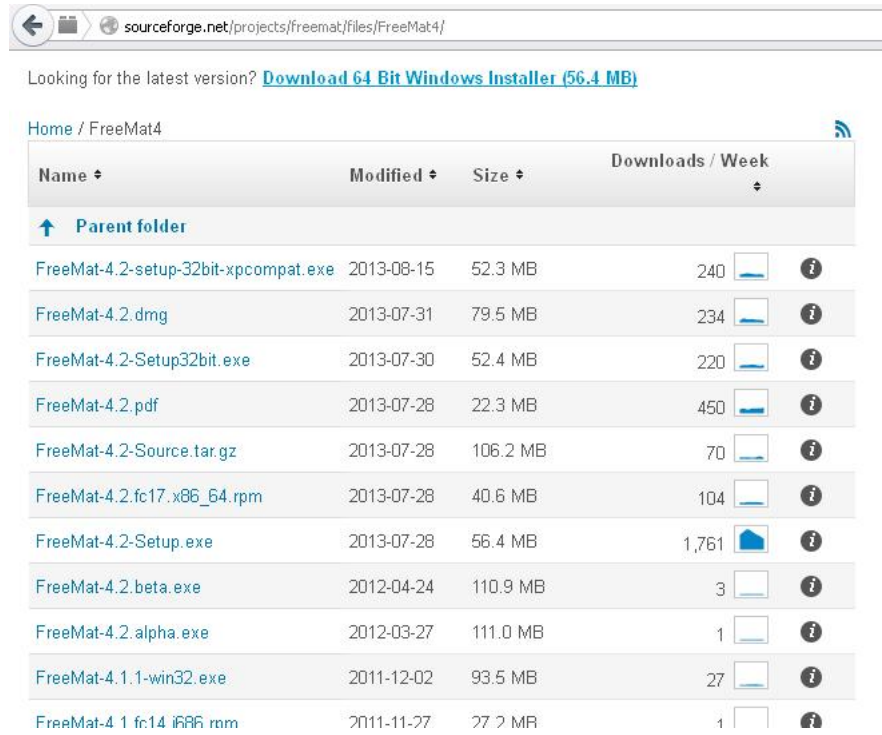
Linux

Windows

Go to the downloads page.

Select the preferred install.

The website offers the possibility to make a determination which version to install. I first installed a win version that did not work, as my computer has an older version of Win XP.



Looking for the latest version? [Download 64 Bit Windows Installer \(56.4 MB\)](#)

Home / FreeMat4

Name	Modified	Size	Downloads / Week
Parent folder			
FreeMat-4.2-setup-32bit-xpcompat.exe	2013-08-15	52.3 MB	240
FreeMat-4.2.dmg	2013-07-31	79.5 MB	234
FreeMat-4.2-Setup32bit.exe	2013-07-30	52.4 MB	220
FreeMat-4.2.pdf	2013-07-28	22.3 MB	450
FreeMat-4.2-Source.tar.gz	2013-07-28	106.2 MB	70
FreeMat-4.2.fc17.x86_64.rpm	2013-07-28	40.6 MB	104
FreeMat-4.2-Setup.exe	2013-07-28	56.4 MB	1,761
FreeMat-4.2.beta.exe	2012-04-24	110.9 MB	3
FreeMat-4.2.alpha.exe	2012-03-27	111.0 MB	1
FreeMat-4.1.1-win32.exe	2011-12-02	93.5 MB	27
FreeMat-4.1.fc14.i686.rpm	2011-11-27	27.2 MB	1

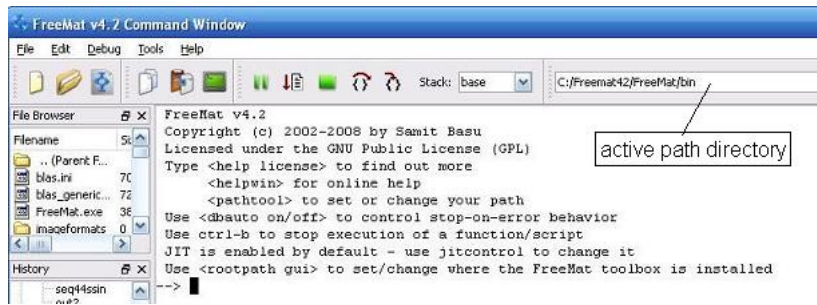
I had to uninstall the first downloaded and installed version and had to replace it with the xpcompat.exe version. Please check directly after installation that you can open and run Freemat.

You can download the install files to a preferred directory, for instance to Freemat42. Double click on the install file and install Freemat, preferably into Freemat42. The executable Freemat.exe is then installed in /Freemat42/Freemat/bin.

Freemat can be started by clicking on the short-cut (if created). If there is no shortcut, you have to go either to the Start Menu in Windows or to the directory where Freemat is installed. (e.g.. .../Freemat42/Freemat/bin/FreeMat.exe) and double-click on FreeMat.exe.

There may be some latency (actually both in starting Freemat or Matlab), so be patient.

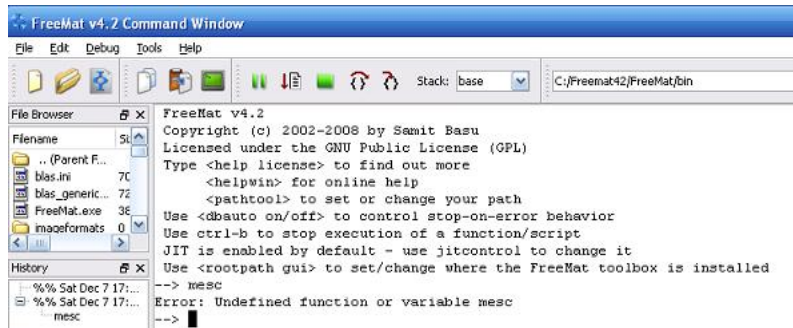
You will see the following after Freemat is started. This is called the Freemat Command Window.



The upper right window displays the "active path window". This means that without any further instructions Freemat (and equally Matlab) only knows or executes programs that are located in the active path.

Furthermore, Freemat does not know any data that has not been 'loaded' into the program as being active data.

The following instruction 'mesc' asks Freemat to list a variable 'mesc' or to execute a program 'mesc'. However, the variable/program 'mesc' is not in /Freemat42/Freemat/bin and an error message will appear.



Remember, we have extracted all programs related to the lectures into the directory /TernaryPrograms/ with relevant subdirectories. To be able to run these programs, Freemat (and Matlab) needs to be instructed to make the directory /TernaryPrograms/ **with its subdirectories** to be in the active path.

You can do this in several ways:

(A) with the 'pathtool' instruction.

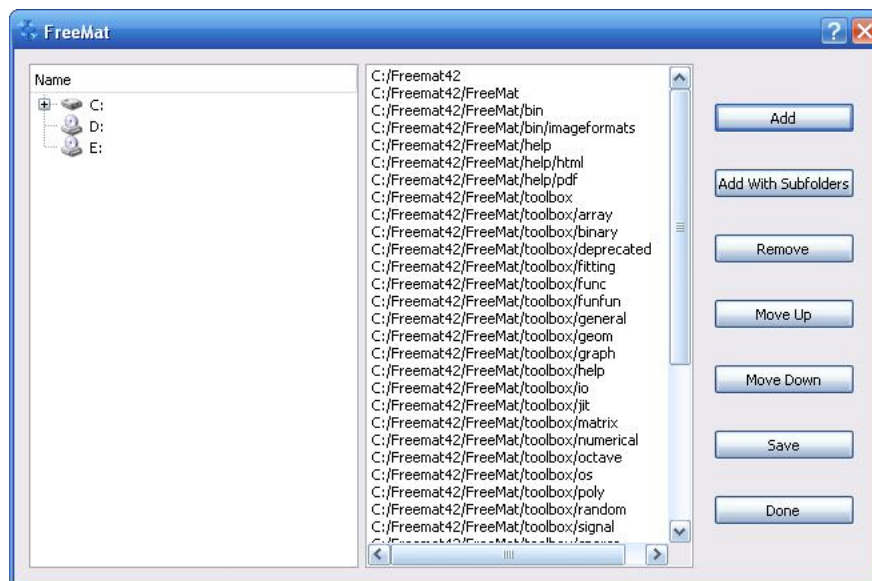
Type "pathtool" in the Command window

```

FreeMat v4.2
Copyright (c) 2002-2008 by Samit Basu
Licensed under the GNU Public License (GPL)
Type <help license> to find out more
  <helpwin> for online help
  <pathtool> to set or change your path
Use <dbauto on/off> to control stop-on-error behavior
Use ctrl-b to stop execution of a function/script
JIT is enabled by default - use jitcontrol to change it
Use <rootpath gui> to set/change where the FreeMat toolbox is installed
--> mesc
Error: Undefined function or variable mesc
--> pathtool

```

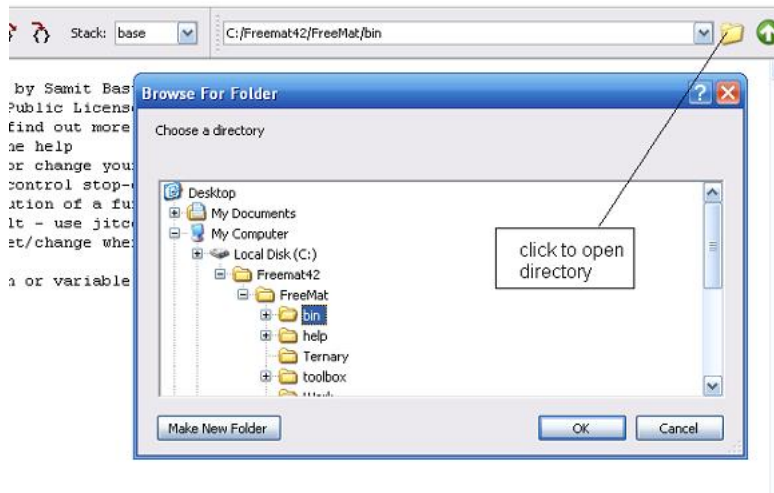
This will bring up the pathtool window



You can search for any of your directories and subdirectories and place it in the path that Freemat will consider. Don't forget to hit 'save' after updating the path.

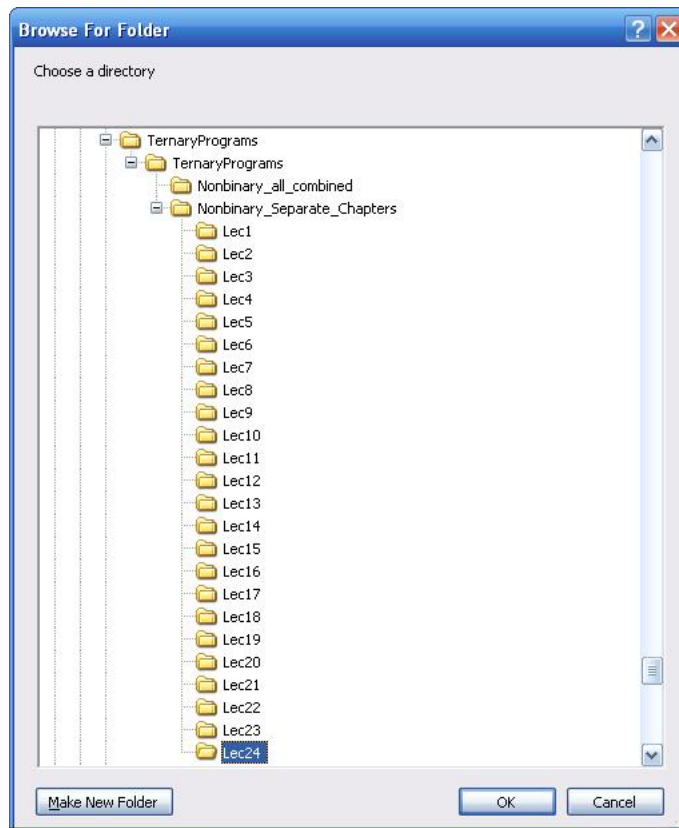
(B) by changing the active window

Click on the Folder icon next right to the folder window



The directory browser will open at the current active directory.

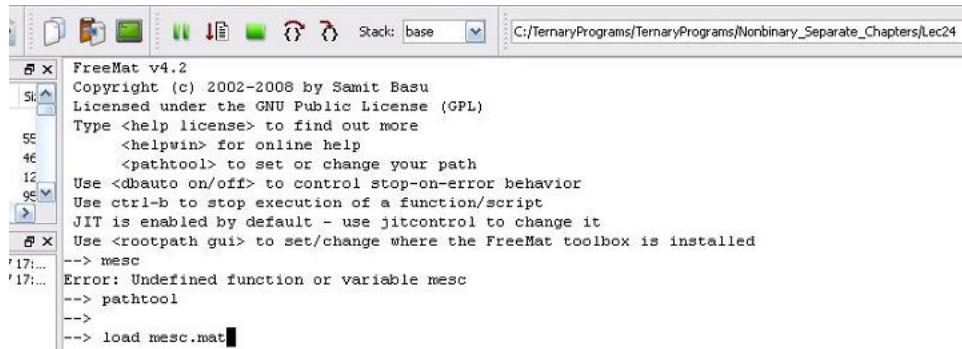
Change to and select
/TernaryPrograms/Ternaryprograms/Nonbinary_Separate_Chapters/Lec24



We have shown earlier that a file mesc.mat is in the active folder.

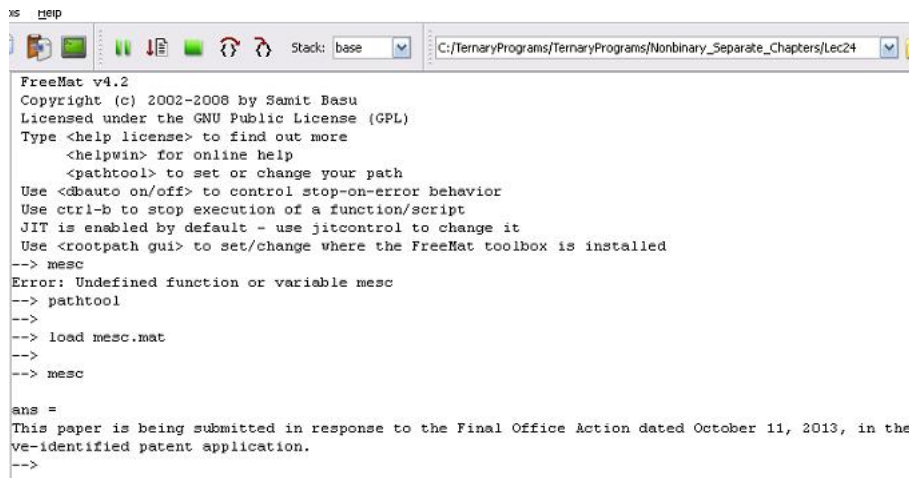
To activate the data, it has to be loaded into Freemat with the instruction 'load mesc.mat'.

Freemat opens the file and assigns the variable 'mesc' to the data.



```
FreeMat v4.2
Copyright (c) 2002-2008 by Samit Basu
Licensed under the GNU Public License (GPL)
Type <help license> to find out more
  <helpwin> for online help
  <pathtool> to set or change your path
Use <dbauto on/off> to control stop-on-error behavior
Use ctrl-b to stop execution of a function/script
JIT is enabled by default - use jitcontrol to change it
Use <rootpath gui> to set/change where the FreeMat toolbox is installed
--> mesc
Error: Undefined function or variable mesc
--> pathtool
-->
--> load mesc.mat
```

Freemat lists the content of 'mesc' by typing 'mesc' which is a text string.



```
FreeMat v4.2
Copyright (c) 2002-2008 by Samit Basu
Licensed under the GNU Public License (GPL)
Type <help license> to find out more
  <helpwin> for online help
  <pathtool> to set or change your path
Use <dbauto on/off> to control stop-on-error behavior
Use ctrl-b to stop execution of a function/script
JIT is enabled by default - use jitcontrol to change it
Use <rootpath gui> to set/change where the FreeMat toolbox is installed
--> mesc
Error: Undefined function or variable mesc
--> pathtool
-->
--> load mesc.mat
-->
--> mesc

ans =
This paper is being submitted in response to the Final Office Action dated October 11, 2013, in the
ve-identified patent application.
-->
```

Mesc is a text that is to be encrypted by transposition and 128-state scrambling and can be recovered by descrambling and reverse transposition.

Matlab

The above steps also apply to Matlab.

It is noted that Matlab is marketed by The Mathworks, Inc. and a valid license to install and operate Matlab is required. Such a license is not and cannot provided herewith and should be obtained separately.

If you are a student, you may be eligible to obtain a relatively inexpensive Student Edition of Matlab.

You are ready to proceed with lecture 1.

When you have reached this point, it seems that everything is installed and is working properly and you can proceed to Lecture 1 with the related programs.

Freemat/Matlab scripts and functions

This is not intended to be a tutorial on either Matlab or Freemat. The purpose is to get you to a stage where you can execute the programs.

One aspect that is important in Matlab/Freemat is the distinction between scripts and functions.

Scripts apply the data and variables that are in the active "variable" space, which is represented by the Variables window in Freemat and the 'Workspace' window in Matlab.

You can define variables 'aa=4' and 'bb=8' in the command window. You can have a script 'addaabb' that generates a result 'cc=aa+bb'.

The variables aa, bb and cc are persistent in the active workspace and can be acted upon by other scripts. All intermediate variables generated by such scripts are in the active workspace.

A function requires an input variable to generate an output. All variables generated during execution are local and are not available after execution.

Here are some examples based on Lecture 24. A page listing several procedures is shown below.

Running the encryption/decryption programs

To encrypt:

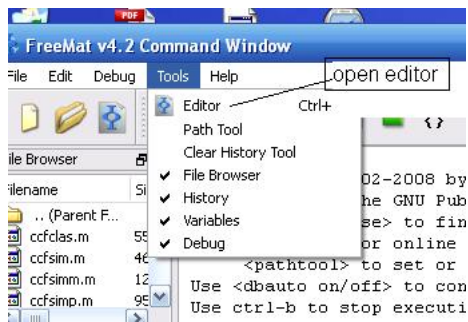
1. run seq44ssin.m to generate 255 4-state symbol PN sequence 'out2'
2. run zz=seq42vec(out2) to generate the transposition sequence 'zz'
3. enter message 'mesc'
for instance mesc = 'this is a message', or perform:
load mesc.mat; type mesc to confirm
4. open 128-state functions by:
load sc27.mat
load m27.mat
load sd28.mat
5. run mesenc = scamtranspsep(zz,mesc,sc27,m27) to encrypt

You can create your own message. Remember that message size should be less than 255 symbols.

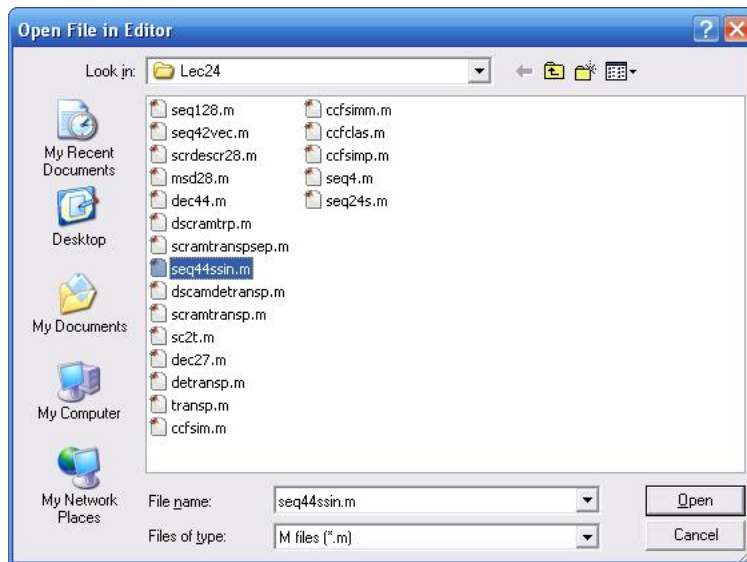
To decrypt:

1. run dest=dscramtrp(zz,mesenc,sc27,m27,sd28)

One can view the listing of scripts or functions by applying the Freemat editor.



Open can open the listing in the Editor.



A partial listing of seq44ssin.m is:

```

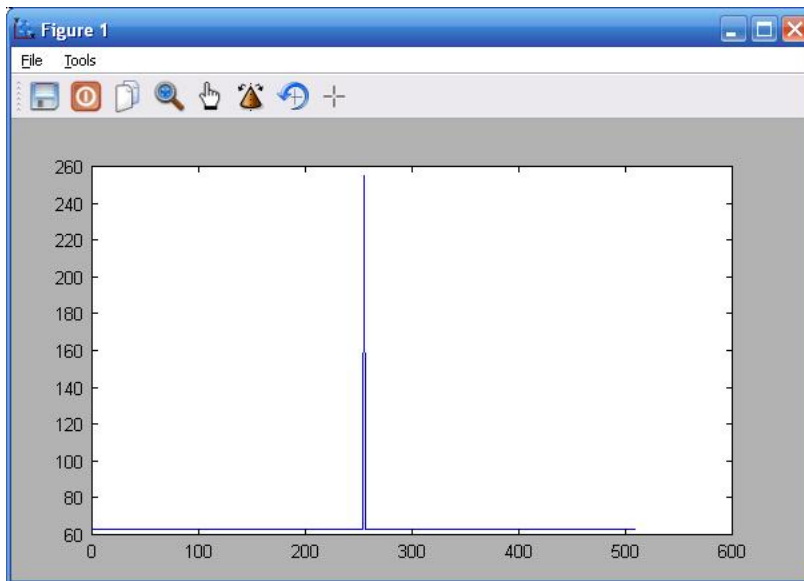
seq44ssin.m*
1      % © Copyright 2013 Ternarylogic LLC. All Rights Reserved
2      % title: seq44ssin.m
3      % generate a single PN 4-state sequence of 255 symbols
4      sc4=[0 1 2 3;1 0 3 2;2 3 0 1;3 2 1 0]+1; % 4-state function
5      sc4d=[0 1 1 1;1 0 1 1;1 1 0 1;1 1 1 0]+1; % 4-state function
6      m4=[0 0 0 0;0 1 2 3;0 2 3 1;0 3 1 2]+1; % inverters
7      tel=1;
8      %res4=zeros(150,6);
9      for i0=3:3
10         for i1=1:1
11             for i2=2:2
12                 for i3=2:2
13                     shifts=[1 3 2 4];
14                     inv3=m4(i3,:);
15                     inv2=m4(i2,:);
16                     inv1=m4(i1,:);
17                     inv0=m4(i0,:);
18         for k=1:255
19             in=shifts(4);
20             in3=inv3(in);

```

The execution of 'seq44ssin.m' is shown below. Just type 'seq44ssin' in the Command window.

```
--> mesc
ans =
This paper is being submitted in response to the Fi
ve-identified patent application.
-->
--> 'generate a sequence out2 of 4-state symbols'
ans =
generate a sequence out2 of 4-state symbols
-->
--> seq44ssin
```

One output is a correlation graph:



Another result is a generated 4-state sequence 'out2' which is not listed. The listing on the screen takes place by typing the name of the variable 'out2' in the Command Window.

```

--> out2
ans =
Columns 1 to 34
2 4 2 3 2 3 2 1 4 1 2 2 2 1 4 2 1 2 1 4 4 3 2 3 3 4 4 4 2 4 4 2 1
Columns 35 to 68
1 2 4 3 2 2 2 4 1 2 4 4 1 2 1 1 1 3 3 1 3 2 1 2 3 2 4 1 1 3 4 2 3 1
Columns 69 to 102
4 2 2 1 1 3 1 3 3 4 2 2 4 4 3 4 1 3 2 3 4 3 4 3 1 2 1 3 3 3 1 2 3 1
Columns 103 to 136
3 1 2 2 4 3 4 4 2 2 2 2 3 2 2 3 1 1 3 2 4 3 3 3 2 1 3 2 2 1 3 1 1 1
Columns 137 to 170
4 4 1 4 3 1 3 4 3 2 1 1 4 2 3 4 1 2 3 3 1 1 4 1 4 4 2 3 3 2 2 4 2 1

```

The following screens show the execution of the consecutive commands and their results.

The next statement shows the generation of a transposition sequence 'zz' by a function 'seq42vec' which requires an input 'out2'.

```

--> zz=seq42vec(out2)
zz =
Columns 1 to 20
118 217 102 153 100 147 76 49 197 21 84 83 77 5
Columns 21 to 40
230 154 107 175 191 255 253 247 223 125 244 208 65
Columns 41 to 60

```

The following statement shows loading of variables 'sc27', 'm27' and 'sd28'.

```

--> load sc27.mat
--> load m27.mat
--> load sd28.mat
-->
-->
\

```

The following statement shows the execution of encryption by a function which requires inputs: 'zz', 'sc27', 'm27' and 'mesc'.

The output is an encrypted message 'mesenc'.

```

-->
--> mesenc=scramtranspsep (zz,mesenc,sc27,m27)

ans =
message with added characters

ans =
                EEEEEEEEEEEEEEEEEEEEESSSSSSSSSSSSSS
esponse to the Final Office Action dated October 11, 2013,
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee.....

ans =
scrambled message in characters

ans =
?PFPQADXX*ZZy&TTQASyQ~(3E{DVEO[-})Q2QoQ2pfF Q~QPPZC~xQ+Z
tQ:9Q7|Q5QinQ-7%eepQ+OuiPQ~tdQQPbzx(zyR9o~Q<vH"Qn rCh241Q
p QeIQwCwbQcRTEc3Cw4QQ Tz#lRvQl8Q QSTc3n...cbQQT Q3FSQ?

```

The decryption of the encrypted message 'mesenc':

```

-->
--> dest=dsclamtrp (zz,mesenc,sc27,m27,sd28)

ans =

Columns 1 to 20
    35    45    25    40    76    99    19    43   116   125     3    80    49   103    60    79    84    58    80

Columns 21 to 40
   103    79    45    53    50    68    51   106    45    65   123   111   110    90    65     4   122    43   100

Columns 41 to 60

```

Don't worry about the functionality of the functions. The above steps are only shown as examples of procedural aspects of Freemath and Matlab.

The actual functionality is explained in the lectures.

End of Getting Started

If you found errors or you are still unsure, send an e-mail to admin@ternarylogic.com.

MatLab® is a trademarked product that is owned and marketed by The MathWorks, Inc.®.