Lecture 1

- Current directory & Path Setting
- Programs
  - Scripts
  - Functions

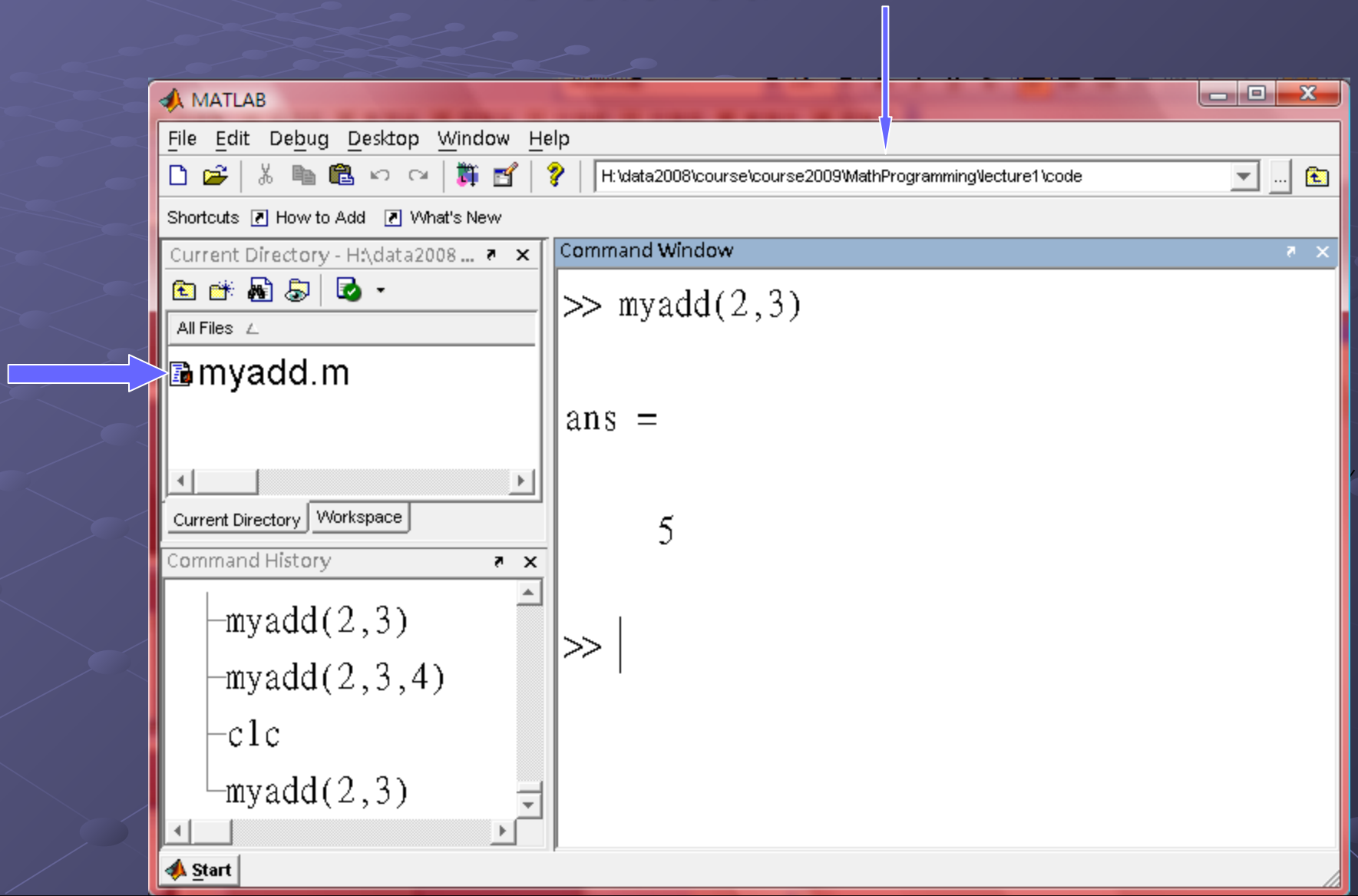
Function myadd

- myadd.m
- Experiment
  - Download myadd.m and store it to some directory (named ‘code’)
  - Set current directory to where myadd.m is stored

```
>> myadd(2,3)

ans =

    5
```

```
>> myadd(2,3,4)

ans =

    9
```

Set current directory to where myadd.m is stored



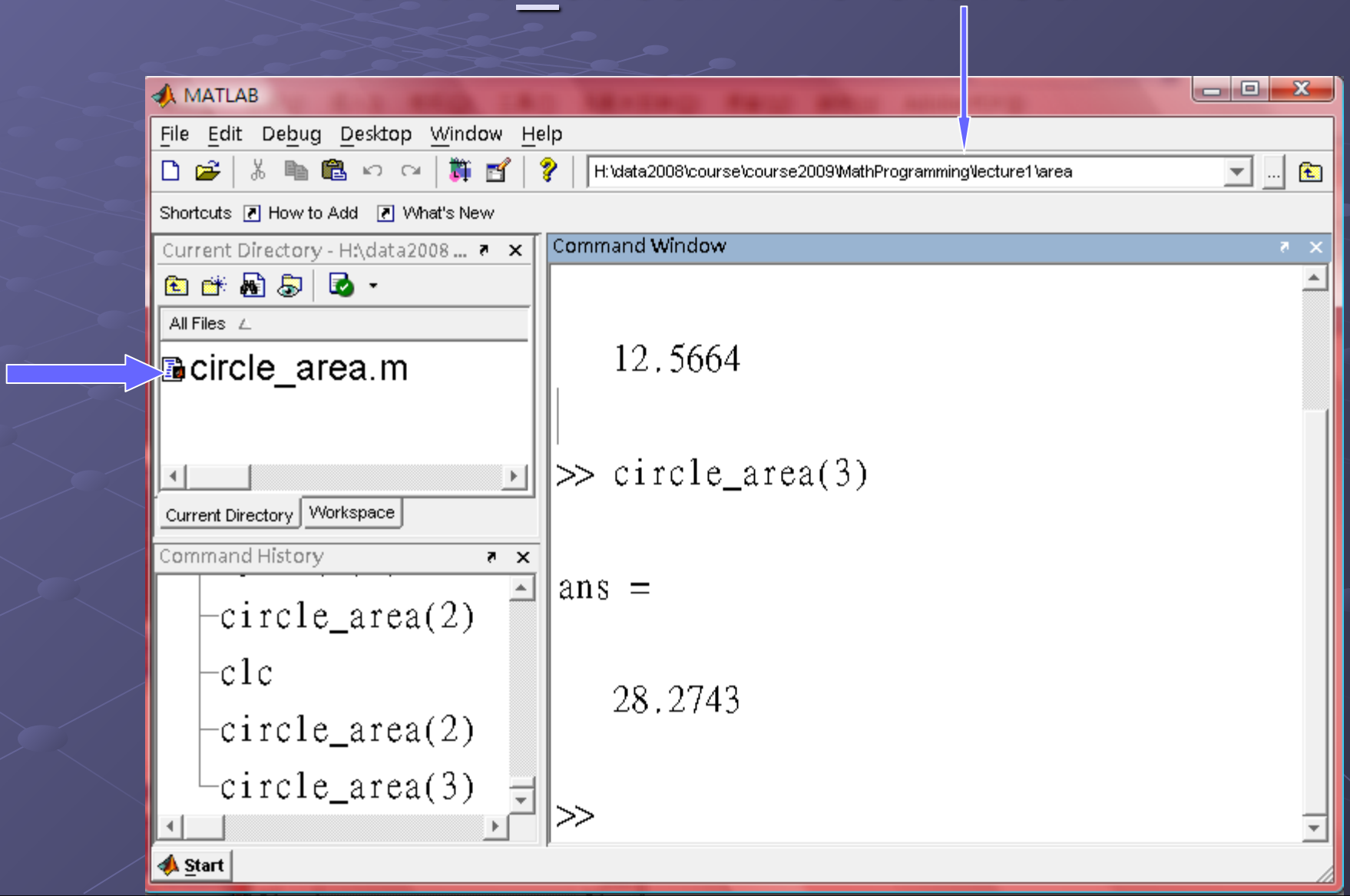
Function circle_area

- circle_area.m
- Experiment (1)
  - Download circle_area.m and store it in some directory (named ‘area’)
  - Set current directory to where circle_area is stored

```
>> circle_area(2)

ans =

   12.5664
```

```
>> circle_area(3)

ans =

   28.2743
```

Set current directory to where circle_area.m is stored



Undefined functions

- Experiment (2)
  - Set current directory to ‘code’

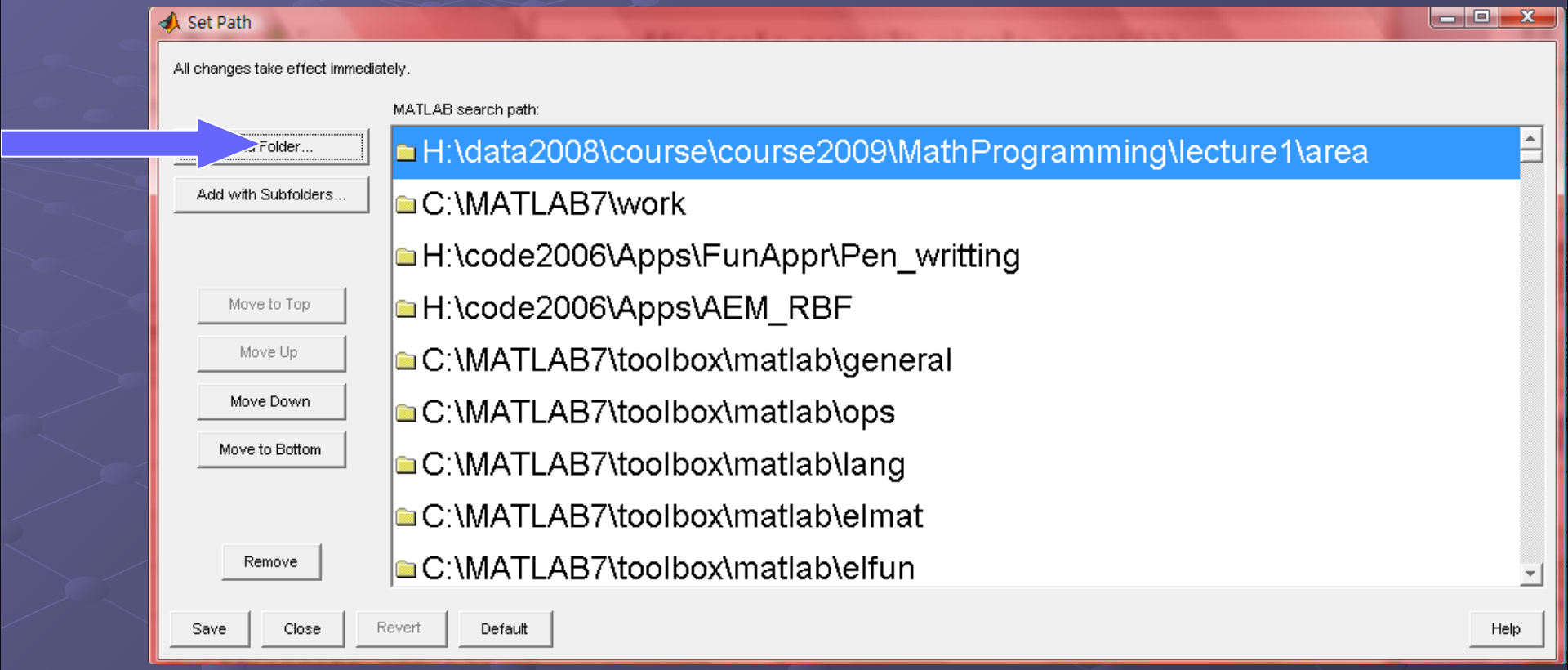
```
>> myadd(circle_area(2),circle_area(3))
??? Undefined command/function 'circle_area'.
```

- Unable to reach directory ‘area’
- MATLAB engine found no function that is named as ‘circle_area’

Add folds to path

- Access user-created MATLAB functions
- Add directory ‘area’ to path
  - Click ‘file’
  - Select ‘set path’
  - Press ‘add fold’
  - Select directory that stores ‘circle_area.m’

Add Folder


Set Path
All changes take effect immediately.
MATLAB search path:
Folder...
Add with Subfolders...
Move to Top
Move Up
Move Down
Move to Bottom
Remove
H:\data2008\course\course2009\MathProgramming\lecture1\area
C:\MATLAB7\work
H:\code2006\Apps\FunAppr\Pen_writting
H:\code2006\Apps\AEM_RBF
C:\MATLAB7\toolbox\matlab\general
C:\MATLAB7\toolbox\matlab\ops
C:\MATLAB7\toolbox\matlab\lang
C:\MATLAB7\toolbox\matlab\elmat
C:\MATLAB7\toolbox\matlab\elfun
Save
Close
Revert
Default
Help

Experiment (3)

- Set current directory to ‘code’
- Add ‘area’ to path
- Execute

```
>> myadd(circle_area(2),circle_area(3))

ans =

   40.8407
```

- Now both ‘myadd’ and ‘circle_area’ are well defined and reachable

Matlab functions

- Built-in functions
- User-created functions

Built-in functions

- Examples
  - Matrix generation : rand, zeros, ones, eye
  - Matrix manipulation : eig, size, repmat, reshape
  - Algebraic functions : sin, cos, tan, tanh, exp, log…
- I/O
  - load, plot, input, display, imread, image

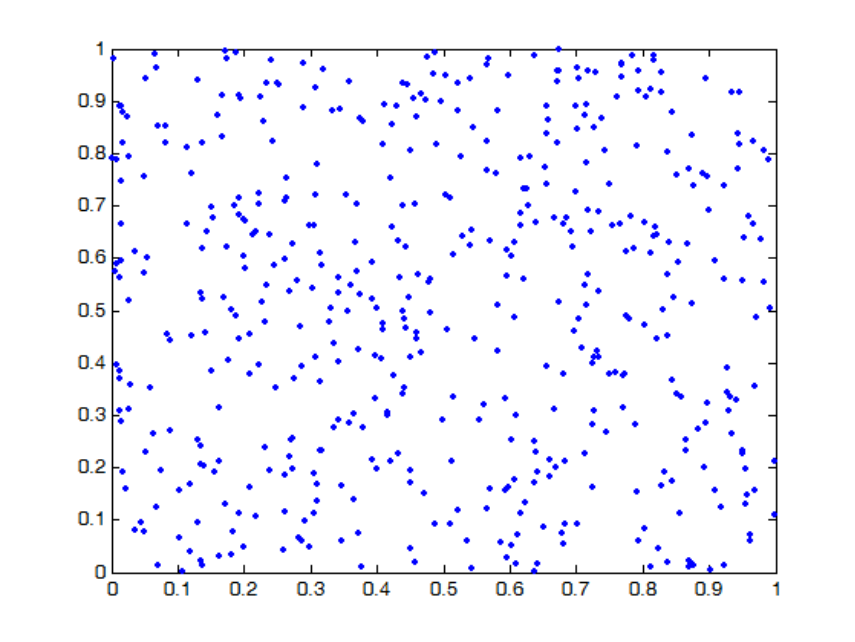
Matrix generation

- rand
  - generate a matrix whose elements are sampled from random variables
- zeros
  - generate a matrix whose elements are all zeros
- ones
  - generate a matrix whose elements are all ones
- eye
  - generate an identity matrix

rand

- rand(m,n)
  - Create an mxn matrix with elements sampled from a uniform distribution

```
>> m=2;n=500;
>> x=rand(m,n);
>> plot(x(1,:),x(2,:),'.');
>>
```



zeros

- zeros(m,n)
  - Create an mxn matrix with zero elements

```
>> m=2;n=5;
>> x=zeros(m,n)

x =

     0     0     0     0     0
     0     0     0     0     0
```

ones

- ones(m,n)
  - Create an mxn matrix with all elements equaling ones

```
>> m=2;n=5;
>> x=ones(m,n)

x =

     1     1     1     1     1
     1     1     1     1     1
```

eye

- eye(m)
  - Create an mxm identity matrix

```
>> m=3;
>> A=eye(m)

A =

     1     0     0
     0     1     0
     0     0     1
```

User-defined functions

- User-defined functions
  - New a file
  - Define a function and store it to some directory with some function name
- MATLAB toolboxes
  - Functions created by third parties
    - Signal process
    - Statistics
    - Bioinformatics
    - Neural networks
    - Image process

Access of Matlab functions

- Function calls drive Matlab engine to search functions
- Searching directories
  - Current directory
  - Then directories added in path
    - Top down