- Announcements:
  - Project 3 will be posted this weekend, due Thurs after break
  - Prelim 1 solutions:
    - Check it out if you made minor mistakes
    - Don't read the solution to a particular question if you had significant errors! Instead, try to re-do it and/or come to office hr to work with a staff member to really learn the material.

- Previous Lecture:
  - Easy plots in MATLAB
  - 1-d array of characters—string
- Today's Lecture:
  - 2-d array—matrix
- Reading:
  - CFile: Chapter 9 Sec 9.1, 9.2

2-d array: **matrix**

- An array is a named collection of like data organized into rows and columns
- A 2-d array is a table, called a *matrix*
- Two *indices* identify the position of a value in a matrix, e.g.,

  `mat(r,c)`

  refers to the cell in row r, column c of matrix mat
- Array index starts at 1

Creating a matrix

- Built-in functions: ones, zeros, rand
  - E.g., ones(2,3) gives a 2-by-3 matrix of 1s
- "Build" a matrix using square brackets, [ ]:
  - [x y] puts y to the right of x
  - [x; y] puts y below x
- What are the dimensions of a matrix M?
  - [nr, nc] = size(M) % nr is #of rows, nc is #of columns
  - nr= size(M, 1) % # of rows
  - nc= size(M, 2) % # of columns


```
A= [1  1]
A= [A'  ones(2,1)]
A= [1  1  1  1;  A  A]
```

a. Error in 2nd statement
b. Error in 3rd statement
c. A is 3-by-4 matrix
d. A is 4-by-3 matrix
e. A is vector of length 12

Example: min value in a matrix

```
function val = minInMatrix(M)
% val is the lowest value in matrix M
```

Pattern for traversing a matrix M

```
[nr, nc] = size(M)
for  r= 1:nr
    for  c= 1:nc

        % do something with M(r,c) ...
    end
end
```



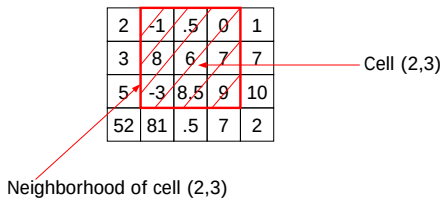
```
% Given an nr-by-nc matrix M
for  r= 1: nr
   for c= 1: nc
      A(c,r)= M(r,c);
   end
end
```

a. A is M with the columns in reverse order
b. A is M with the rows in reverse order
c. A is the transpose of M
d. A and M are the same

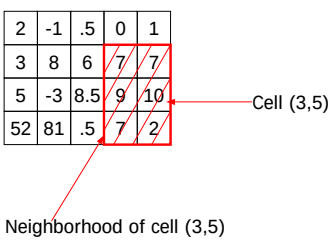

```
% Given an n-by-m matrix A
for  g= 1: n
   for  h= 1: floor(m/2)
      A(g,h)= A(g, m-h+1);
   end
end
```

a. Reflect the right half of A onto the left half
b. Reflect the bottom half of A onto the top half

Local minimum in a neighborhood



Local minimum in a neighborhood



Local minimum in a neighborhood

- Write a function minInNeighborhood
- Input parameters:
  - M: matrix of numeric values
  - loc: location of the middle of the neighborhood
    loc(1), loc(2) are the row, column numbers
- Output parameter: minVal
  The minimum value of the neighborhood

Ask yourself leading questions!

- Can you find the min of a (sub)matrix?
  - Yes! Our function minInMatrix(A)
- Given the indices r, c (representing cell M(r,c)), is it easy to define the neighborhood?
  - Yes, for the general case the neighborhood is M(r-1:r+1, c-1:c+1)
  - But the "border cases" add complexity
- Can we get rid of the border cases?

Local minimum in a neighborhood

M

2	-1	.5	0	1
3	8	6	7	7
5	-3	8.5	9	10
52	81	.5	7	2

Cell (3,5)

Want to be able to use the general case, `M(r-1:r+1,c-1:c+1)`

Local minimum in a neighborhood

M

2	-1	.5	0	1
3	8	6	7	7
5	-3	8.5	9	10
52	81	.5	7	2

Want to be able to use the general case, m(r-1:r+1,c-1:c+1)