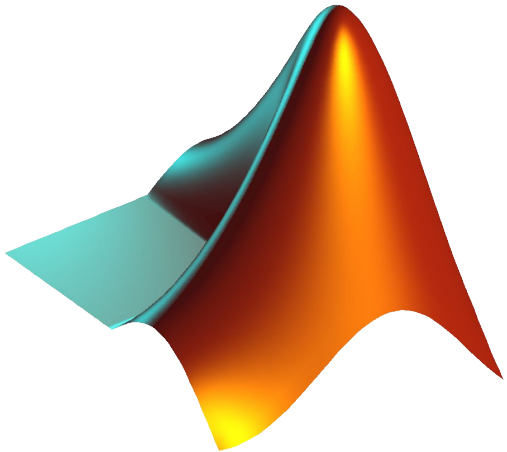
CS 1112 Introduction to Computing Using MATLAB



Instructor: Dominic Diaz

Website: https://www.cs.cornell.edu/courses/cs1112/2022fa/

Today: loops, nested loops, and graphics

Agenda and announcements

- Last time
  - For loops and while loops
- This time
  - More loops and graphics
- Announcements
  - Project 2 posted and due 9/19!
  - If you filled out partner matching survey, partner suggestions have been posted!
  - Do not ask project clarification questions over email
  - Do not ask to get your exercises checked off over email. You must go to consultant hours or TA office hours

For loops versus while loops

Do something n times (or a fixed number of times)

```
for [var] = [start]:[step]:[end]
    [code executed multiple times]
end
```

Do something until a condition stops being true

```
while [continueCriteria]
    [code executed multiple times]
end
```

```
n = 10;
for k = 1:1:n
    disp(k);
end
```

These two codes do the same thing!

```
k = 1; n = 10;
while k <= n
    disp(k);
    k = k + 1;
end
```

Question:

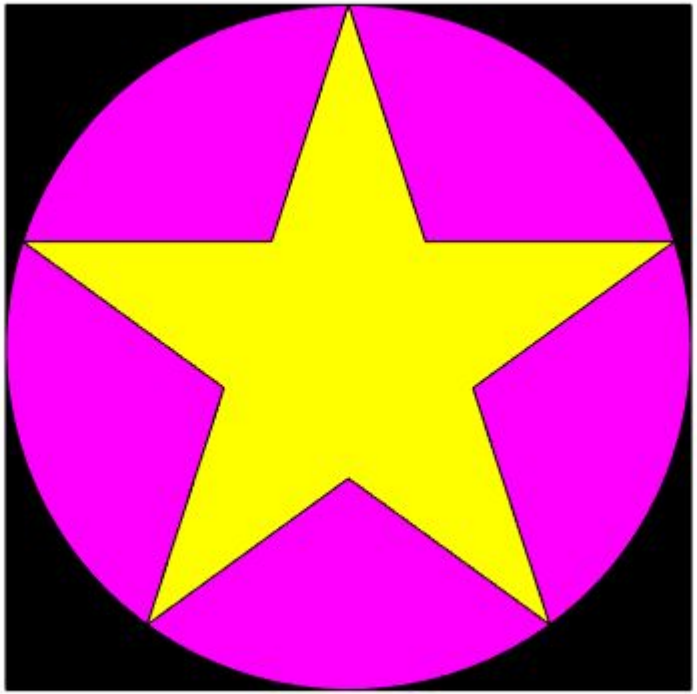
- When do I use `for` loops and `while` loops?
  - Use loops when you need you need some task executed multiple times
- How do I know if I need to use a `for` loop or a `while` loop?
  - Use a while loop when you need some code executed multiple times until some condition is (or is not met)!
  - Use a for loop when you know how many times the loop should run!
- Important features of iteration:
  - Need a starting point
  - Need to know when to stop
  - Need to keep track of (and measure) progress

Review loops/conditionals using graphics function

Draw a black square;

Then draw a magenta disk;

Then draw a yellow star.



We can do this using DrawRect, DrawDisk, and DrawStar (user-defined functions for drawing rectangles, disks, and stars).

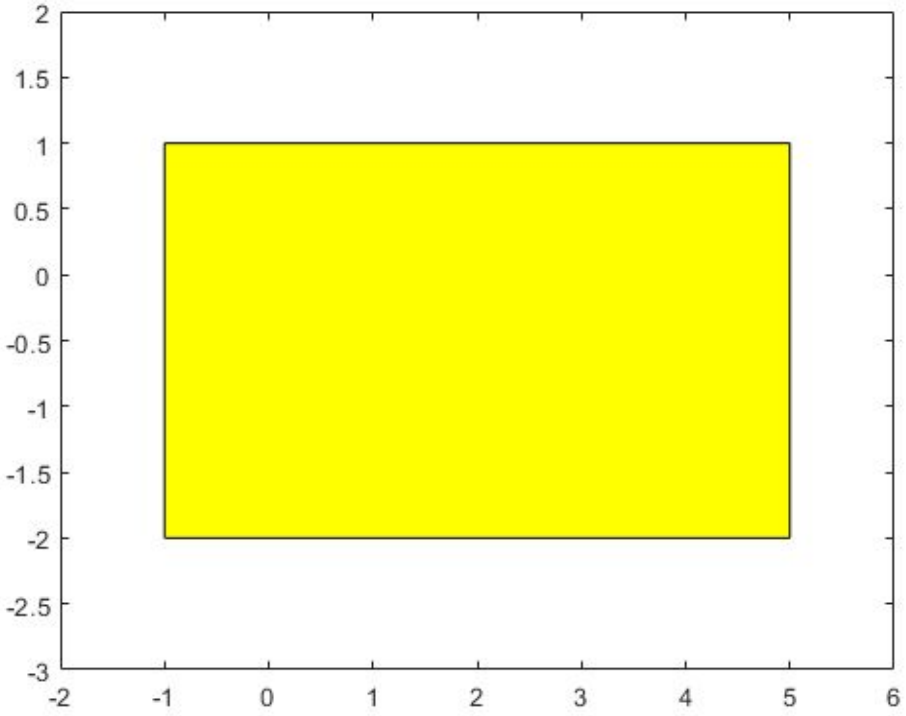
```
DrawRect(-1, -2, 6, 3, 'y')
```

x and y coordinates of lower left corner.

width

height

color

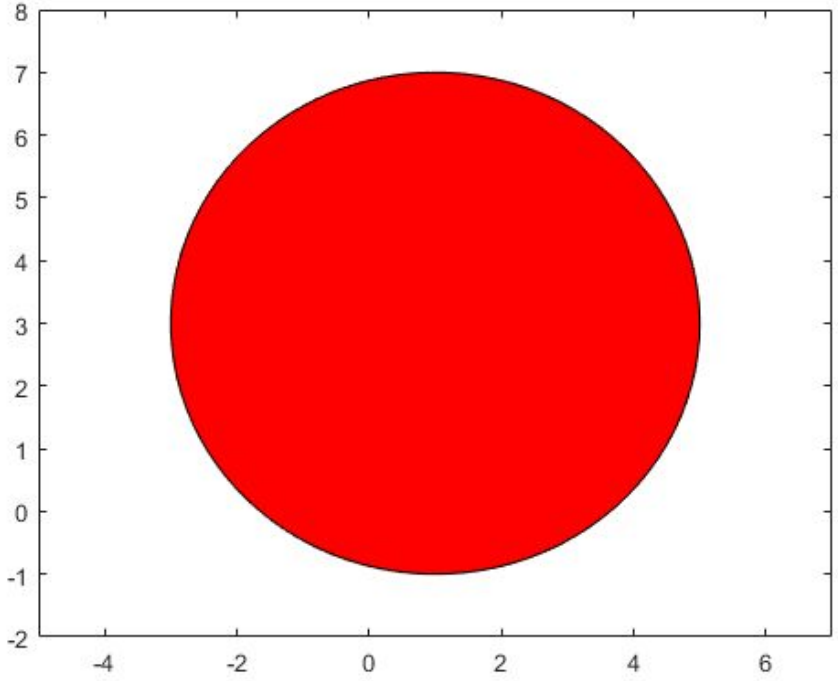


```
DrawDisk(1, 3, 4, 'r')
```

x and y coordinates of the center.

radius

color

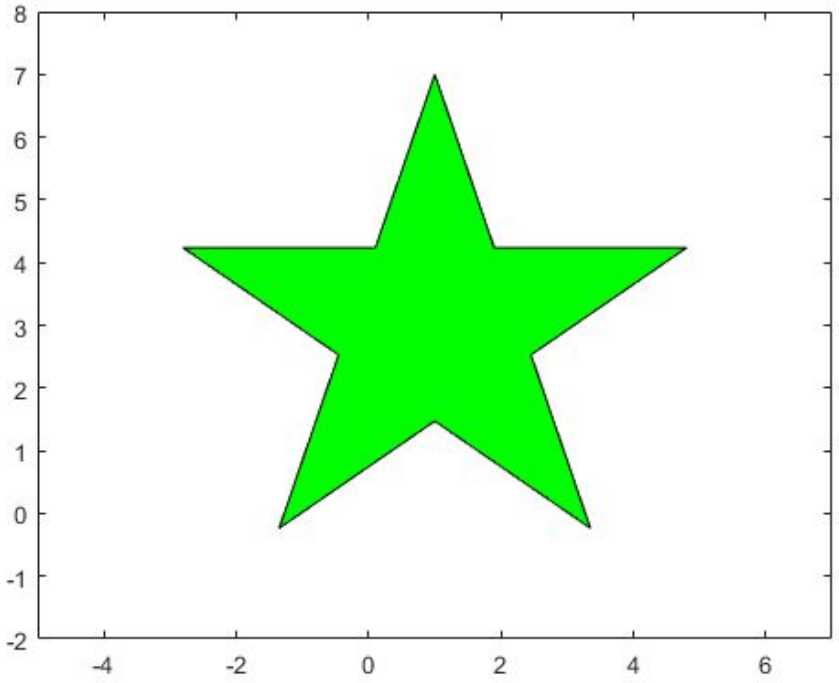


```
DrawStar(1, 3, 4, 'g')
```

x and y coordinates of the center.

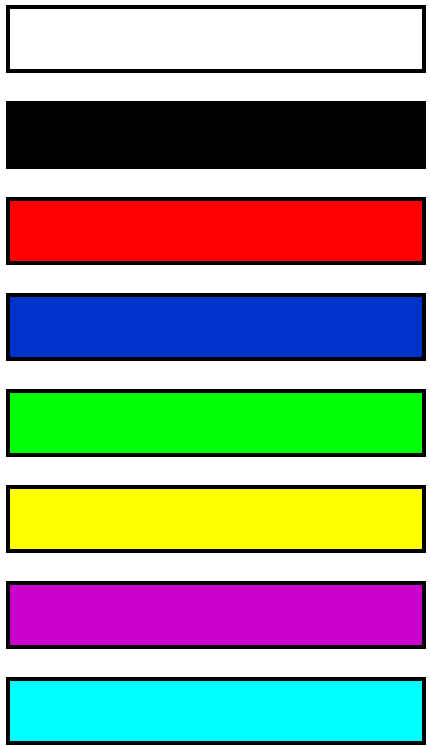
"radius"

color



Color options

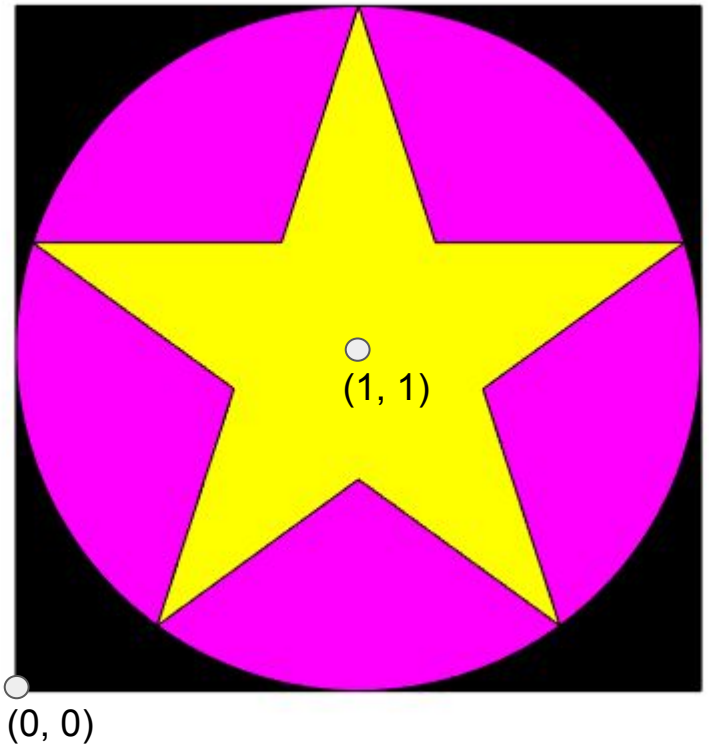
White	'w'
Black	'k'
Red	'r'
Blue	'b'
Green	'g'
Yellow	'y'
Magenta	'm'
Cyan	'c'



Code to draw a star nested inside a circle nested inside a square

To draw the shapes, we need positions and sizes of the shapes!

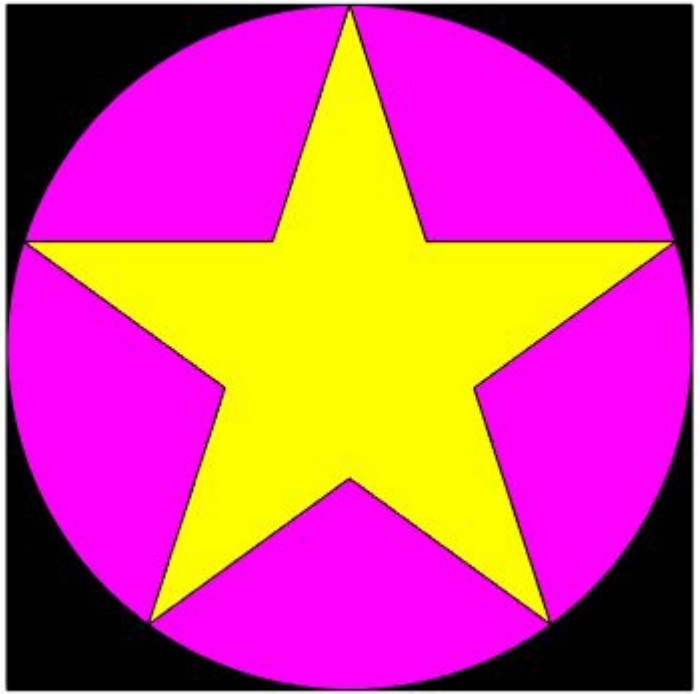
```
DrawRect(0, 0, 2, 2, 'k')
DrawDisk(1, 1, 1, 'm')
DrawStar(1, 1, 1, 'y')
```



Order matters: draw the background one first and the frontmost one last.

Code to draw a star nested inside a circle nested inside a square

```
% Draw square, circle, and star
close all
figure
axis equal off
hold on

DrawRect(0, 0, 2, 2, 'k')
DrawDisk(1, 1, 1, 'm')
DrawStar(1, 1, 1, 'y')

hold off
```



Code to draw a star nested inside a circle nested inside a square

```
% Draw square, circle, and star
close all
figure
axis equal off
hold on

DrawRect(0, 0, 2, 2, 'k')
DrawDisk(1, 1, 1, 'm')
DrawStar(1, 1, 1, 'y')

hold off
```

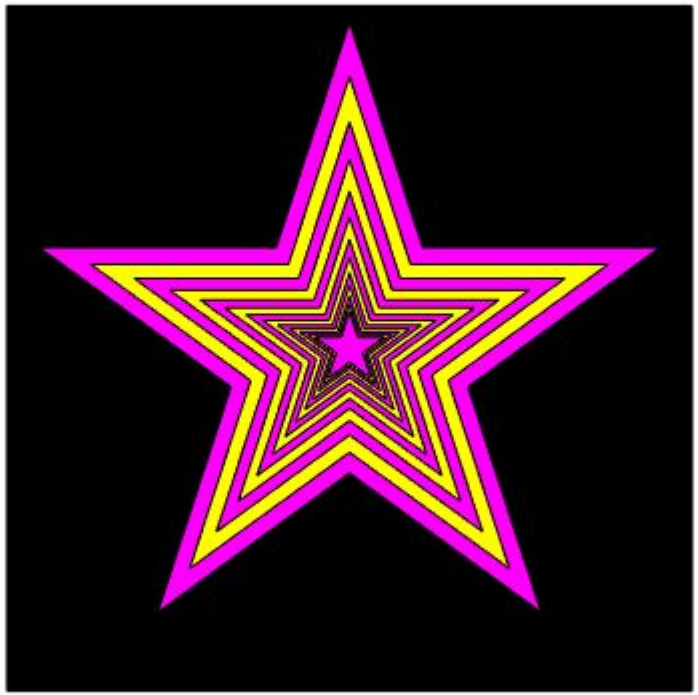
You will need these settings for project 2, problem 3.

- close all: closes all figures
- figure: creates a new figure to draw in
- axis equal off: turns off x and y axes
- hold on: Tells the computer that you want to plot multiple things on the current figure
- hold off: indicates that you are done drawing on the current figure

A general graphics framework

```
% Draw something or multiple things
close all
figure
axis equal off    % only use this line when you don’t want to show axes
hold on

[code fragment to draw objects (like rectangle, disk, star)]

hold off
```

Example: Nested Stars



First draw a square black background then draw nested stars until the star “radius” is too small. We can write a loop that first draws the outermost star and continues to draw smaller stars until the star “radius” is too small.

To do:

Draw a black square (center at (0,0))

Draw a sequence of stars

- Stars alternate in color and get smaller
- Stop when r is small

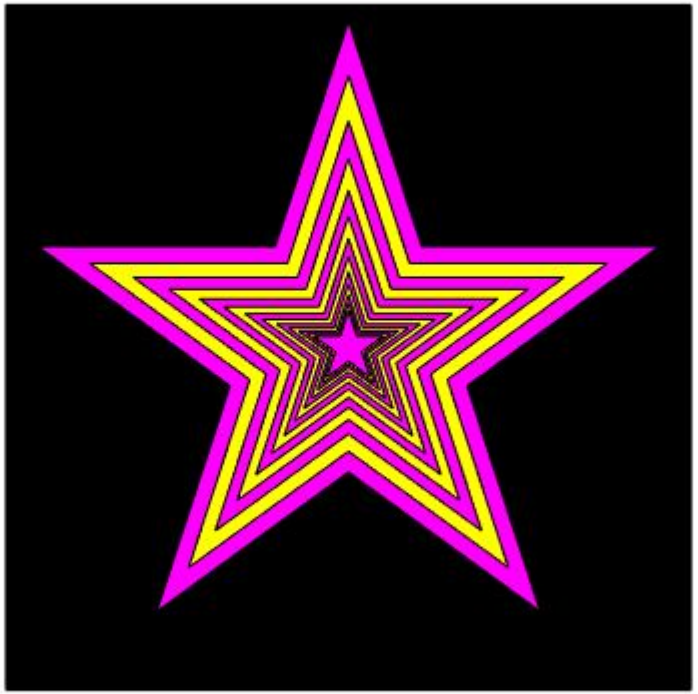
```
% Draw nested stars on a black background
close all
figure
axis equal off
hold on









while r >= 0.1









end
hold off
```

```
% Draw nested stars on a black background
close all
figure
axis equal off
hold on


% Draw the background black square



while r >= 0.1


    % Draw a star with radius r




end
hold off
```

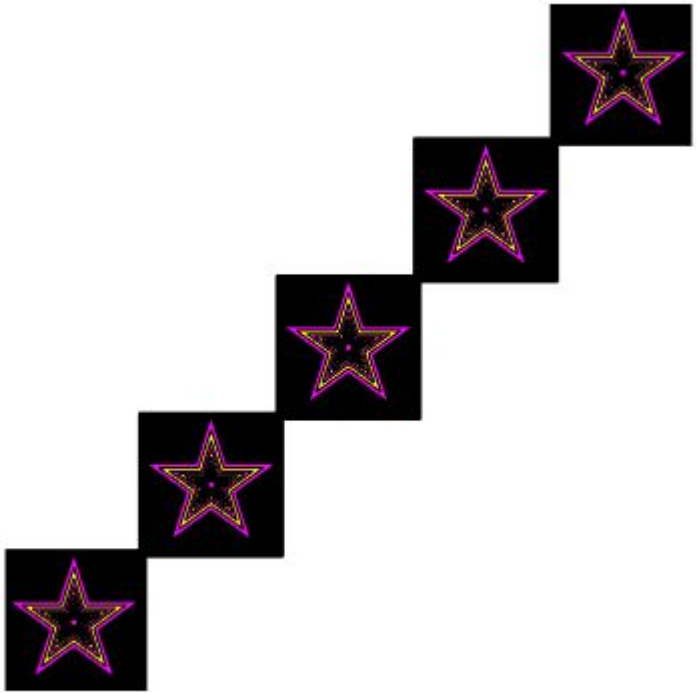
```
% Draw nested stars on a black background
close all
figure
axis equal off
hold on
x = 0; y = 0;   % figure centered at (0,0)
r = 1;          % radius of outermost star
s = 2*r;  % side length of square
DrawRect(x-s/2,y-s/2,s,s,'k')
% Draw nested stars, smallest r at least .1
k = 1;
while r >= 0.1



    % Draw a star with radius r




end
hold off
```

Parametrization: will be doing this in project 2, problem 3.

```
% Draw nested stars on a black background
close all
figure
axis equal off
hold on
x = 0; y = 0;   % figure centered at (0,0)
r = 1;          % radius of outermost star
s = 2*r;  % side length of square
DrawRect(x-s/2,y-s/2,s,s,'k')
% Draw nested stars, smallest r at least .1
k = 1;
while r >= 0.1
     % Draw a star with radius r
     if rem(k,2)==1  % odd k
          DrawStar(x,y,r,'m')  % magenta
     else
          DrawStar(x,y,r,'y')  % yellow
     end
     r = r/1.2;  % Reduce r
     k = k + 1;
end
hold off
```



How do we draw 5 sets of nested stars?



We know how to draw one set of nested stars. But how can we draw 5 sets of nested stars?

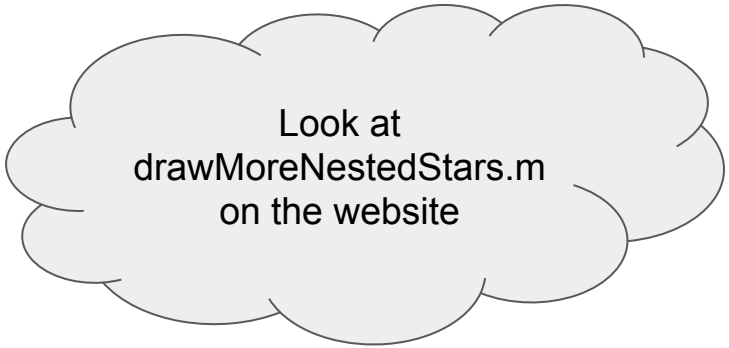
Answer: use nested loops!

I used a `while` loop to draw 1 set of nested stars. What do I use to draw 5 sets of nested stars?

Answer: a `for` loop! (Because we know how many times we need the loop to evaluate.)

Code to draw 5 sets of nested stars

```
for c = 0:2:8
    x = c; y = c;  % figure centered at (x,y)
    r = 1;         % radius of outermost star
    s = 2*r + .1;  % side length of square
    DrawRect(x-s/2,y-s/2,s,s,'k')
    % Draw nested stars, smallest r at least .1
    k = 1;
    while r >= 0.1
        % Draw a star with radius r
        if rem(k,2)==1  % odd k
            DrawStar(x,y,r,'m')  % magenta
        else
            DrawStar(x,y,r,'y')  % yellow
        end
        r = r/1.2;  % Reduce r
        k = k + 1;
    end
end
```



Previous prelim question

```
z = 0;
for i = 1:10
    for j = 1:50
        if i == j
            z = z + 1;
        end
    end
end
```

What is the value stored in z after the above code is executed?