CS1112 Fall 2015 Project 3 Part A due Monday 10/5 at 11pm

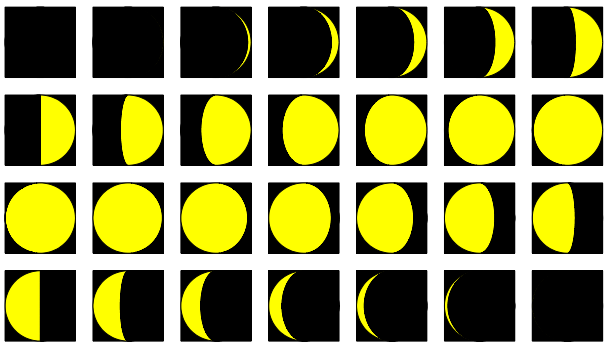
(Part B will appear in a separate document. Both parts have the same submission deadline.)

You must work either on your own or with one partner. If you work with a partner you must first register as a group in CMS and then submit your work as a group. *Adhere to the Code of Academic Integrity.* For a group, "you" below refers to "your group." You may discuss background issues and general strategies with others, but the work that you submit must be your own. In particular, you may discuss general ideas with others but you may not work out the detailed solutions with others. It is not OK for you to see or hear another student's code and it is certainly not OK to copy code from another person or from published/Internet sources. If you feel that you cannot complete the assignment on you own, seek help from the course staff.

Objectives

Completing this project will solidify your understanding of user-defined functions and vectors. Part A focuses on user-defined functions and graphics; Part B additionally involves vectors.

1 Lunar Phases



Complete Problem **P5.3.8** in *Insight* (page 126) but according to the modifications described below. Be sure to read *Insight* §5.3 first—it'll help you with this problem!

1.1 Rectangles without borders

Modify function `DrawRect` (given in *Insight*) so that it draws a colored rectangle *without* a black outline. First study the original function to make sure that you know how it works. To draw a rectangle without a black outline, simply modify the last statement, which calls the built-in function `fill`, to be

```
fill(x,y,c,'line','none')
```

[9/24 Update]: If you use MATLAB verison R2014b or later, use this statement instead:

```
fill(x,y,c,'LineStyle','none')
```

Change the function name (and therefore filename) to be `DrawRectNoLine`.

1.2 `PaintMoon`, modified

Read the original problem statement of **P5.3.8**. You will implement a modified version of function `PaintMoon` that has five parameters instead of four:

```
function PaintMoon(hc,vc,r,T,n)
% Draws on a black background the moon as it appears T days after the new moon.
% T = 0 or 28 corresponds to the new moon.
% T = 14 corresponds to the full moon.
```

```
% The moon is centered at (hc, vc), has radius r, and is painted by
% stacking rectangles of thickness r/n, where n is a positive integer.
% In other words, n is the number of intervals into which the radius is divided.
```

Use the graphics commands `hold on` and `axis equal off` at the beginning of the function and `hold off` at the end. You can use `DrawRectNoLine` and/or the given `DrawRect` function. Depending on your algorithm, you may also use the given `DrawDisk` function.

Test `PaintMoon` to make sure that it works as specified. Be sure that the function is correct before moving on to the next section. If you have included the commands `close all` and `figure` in `PaintMoon` to simplify testing, remove them now—in the next section you need to put multiple moons in *one* figure.

1.3 Phases of the moon

To show the phases of the moon, write a function `LunarCycle` (instead of the script as stated in the original problem statement). Function `LunarCycle` has only one parameter, `n`, which is the number of intervals into which the radius of the moon is divided (for the purpose of painting a moon). Arrange the moons in four rows, as shown in the figure above.

In function `LunarCycle` you will need these graphics commands: `close all`, `figure`, `hold on`, `axis equal off`, and `hold off`. Use the unit radius (radius equals to 1) for the moon.

Submit your files `DrawRectNoLine.m`, `PaintMoon.m`, and `LunarCycle.m` in CMS.

Part **B** will appear in a separate document. Parts A and B have the same due date.