CS1112 Discussion (Lab) Exercise 2

When you have completed the lab, show this sheet and any associated programs to your lab instructor, who will record that you have completed the lab. If you do not finish this exercise during the lab, show the instructor what you have done at the end of the lab and be sure to complete it in the next few days.

If you have any questions, **ask** your lab instructor or a consultant immediately! They are in the lab to help you learn the material.

CMS: Course Management System

You will use CMS to submit homework, find your scores, and view grading comments. Log on to CMS now. If CS1112 is *not* listed as one of your courses, ask your section instructor to add your information into CMS. *Your homework must be submitted through CMS.*

1 Minimum of a quadratic

Download and review `Eg1_2` in *Insight*.
http://www.cs.cornell.edu/courses/cs1112/ → *Insight* Software, Errata, and More

[M1.2.5 from *Insight*] Modify `Eg1_2` to first check that `L` is less than or equal to `R`. Switch the values of `L` and `R` if necessary before computing the minimum of the quadratic.

[M1.2.6 from *Insight*] Reorder the three branches of the conditional statement to first check whether the critical point is inside the interval.

2 Triangle

[M1.2.4 from *Insight*] The three interior angles of any triangle add up to 180°. Complete the program fragment below to print scalene, isoceles, or equilateral given three angles.

```
% Assume a, b, and c are positive integers that sum to 180

if (_____________________________________)
   disp('Scalene triangle')
else
   if (_____________________________________)
      disp('Equilateral triangle')
   else
      disp('Isoceles triangle')
   end
end
```

3 When do 3 random sticks make a triangle?

Complete the following script so that it prints "Yes" if it is possible to form a triangle (that is not degenerate) with three sticks having length a, b, and c. The script should print "no" if it is impossible.

```
clc       % What does this command do?
a = rand  % a random positive number between 0 and 1
b = rand
c = rand
% Add an appropriate if-elseif-else after this comment...
```

4 Which quadrant?

Write two different programs to determine in which quadrant a user-input value of A degrees belongs. Assume that the user may enter any non-negative number. For example, 725° is the same, and should be treated, as 5°. (Hint: the function `rem` that you saw last week might be useful.) To avoid ambiguity, we use the following convention:

$$\text{Quadrant is } \begin{cases} 1 & \text{if } 0 \le A < 90 \\ 2 & \text{if } 90 \le A < 180 \\ 3 & \text{if } 180 \le A < 270 \\ 4 & \text{if } 270 \le A < 360 \end{cases}$$

Print the result. In the first script use four *separate* `if` statements (4 separate `if-end` constructs—no `else` or `elseif`) and call the program `angle1.m`. In the second script, use a *single* `if-elseif-...-else-end` construction for the evaluation and call it `angle2.m`. Pay close attention to the differences among the three programs—are the boolean expressions as concise as they can be?

5 Challenge question*

*Topic for next lecture—not required in this section (but think about it if you have time!)

Nesting in programming refers to the inclusion, or placement, of one construct inside another. For example, we can *nest* `if`-statements:

```
if [ boolean expression 1 ]
      [ code to execute if expression 1 evaluates to true ]
else
      if [ boolean expression 2 ]
            [ code to execute if expr1 evaluates to false but expr2 evaluates to true ]
      else
            [ code to execute if expr1 evaluates to false and expr2 evaluates to false ]
      end
end
```

In the example above, we nested an `if`-statement inside the `else` branch of the first, or outer, `if`-statement. Use *nested* `if`-statements without `elseif` to solve the quadrant question above. Call the script `angle3.m`.

Please delete your files from the computer before leaving the lab!