

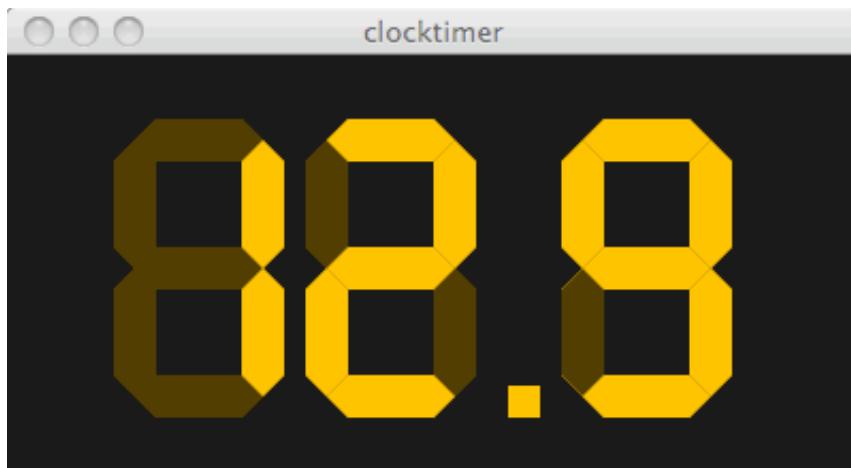


Extra Credit 11: Flexible Segmented Digits

Goal: In this assignment you will practice using functions by rebuilding the timer function from Lecture 11 to have an adjustable size. [Hint: Solve Assignment 11 first to practice using adjustable size.]

Fixed Size to Variable Size

The design for a lighted, segmented digit given in the lecture assumed that the width of each digit would be 60 pixels: A hexagon is made up of rectangle that is 40 pixels long and two triangles of height 10 pixels; these fit together to give an overall width of 60.



So, digits will always be 60 pixels wide. This is the design feature we want to fix.

A New Parameter

Changing the size of the hexagonal segments means that virtually every function except the base functions will need a new parameter, *S*, for size. The parameter *S* gives the *width* of the finished digit in pixels.

Assignment: Rewrite the timer application so that the size of the digit can be controlled by a parameter to the application. (Start with the base functions `hexa()` and `rexa()` from the lecture, but after that it is easiest just to go through the same process of the lecture on your own.) The assignment will involve rewriting all of the functions; however, very few of them will actually require extensive modification.

Wrap-Up

You have practiced programming functions, with the result that you have a timer that has a variable size.

Turn In

Rename your `.pde` file to `<your name>.pde` and submit it to the class drop box.