ISE 108 Midterm 2 — Additional Practice Problems

1. Text Manipulation

A *cut-swap* is defined as follows: a specific substring is identified in a larger string. The original string is replaced by a new string that contains all the characters that originally followed the substring (in their original order), followed by all the characters that originally preceded the substring (again, in their original order); the substring is not present in the new string. For example, suppose you are given the string "catdogmouse" and the substring "dog". After correctly executing a cut-swap, the result will be "mousecat".

The `CutSwap` sketch prompts the user to enter a source string, followed by a substring to remove. It then executes a cut-swap and displays the final result in the sketch window. Complete the `cutSwap()` function so that the sketch works correctly. You may assume that the substring appears ***exactly*** once in the source string.

2. Image Manipulation

The `RecolorPixels` program creates and displays a blank 333 by 333 pixel image. Complete the `mangle()` function, which changes the contents of the image's `pixels[]` array as follows:

a. If the pixel position is even (i.e., index 0, 2, 4, etc.), replace it with a pure black pixel.

b. If the pixel's position is evenly divisible by 3, replace it with a pure red pixel.

c. Otherwise, replace the pixel with a pure blue pixel.

   If (a) and (b) conflict (for example, with an index like 6), (a) should take priority (in other words, use a single `if-else` chain to do the pixel replacement). Remember to use the syntax `color(r, g, b)` to create a pixel!

3. Arrays

Complete the "Interleave" program, which takes two arrays (of equal length) and returns a single large array that contains the contents of the source arrays, in alternating order. For example, if the input arrays are {1, 3, 5, 7, 9} and {2, 4, 6, 8, 10}, the resulting interleaved array should be {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}.