CIS 330: Project #3G
Assigned: May 23rd, 2016
Due May 30th, 2016
(which means submitted by 6am on June 1st, 2016)
Worth 7% of your grade

Please read this entire prompt!

Add 4 new filters:

1) Crop
2) Transpose
3) Invert
4) Checkerboard

Add 1 new source:

1) Constant color

Add 1 new sink:

1) Checksum

Plus: make the two image inputs in Sink be const pointers.

The specifics of the interfaces for filters, source, and sink are specified in the main3G.C file. All of these new filters should work within our existing data flow system (Update/Execute/logging/exceptions/etc).

Place the headers for all of these new modules in "filters.h". (Note that some of them aren't filters … that's OK.)

== Filter 1: Crop ==

Crop takes a range of pixels in width and a range of pixels in height and extracts the square within the range. The range is inclusive, meaning that it should include its endpoints.

NOTE: my Crop code is posted online. You may need to adapt it to work with your own code.

== Filter 2: Transpose ==

Transpose transposes an image. It takes an image of size MxN and makes it be NxM, copying the data so that the first row of the input becomes the first column of the output, etc.

== Filter 3: Invert ==

Invert inverts the colors in an image. This means that that it takes the current value, V, (0<=V<=255) and puts 255-V in its place. (For each color channel)

== Filter 4: checkerboard ==

Takes two input images of identical size and produces an output image of the same size. Pixel (i, j) of the output should be the same as pixel (i, j) from the first input image if ((i/10+j/10)%2 ==0) . Else it should be pixel (i, j) from the second input image. (Note the i/10 and j/10 are doing integer arithmetic … make sure you don't try to do floating point.)

== Source 1: ConstantColor ==

This takes a color and a size. and produces an image of that size with that color

== Sink 1: CheckSum ==

This sink sums up the total value of the red channel, the blue channel and the green channel and outputs it a file. The name of the file is specified as a command line argument.

VERY IMPORTANT: the sum is to be taken modulo 256. So, for each channel (red, green, blue), it will output a number between 0 and 255.

NOTE: if you sum into an unsigned char, then it automatically does summing modulo 256.

Also: the correct output for 3G should be exactly:

"CHECKSUM: 139, 29, 107\n"
139 would be the sum of the red values (% 256)
29 is the sum of the green values (%256)
107 is the sum of the blue values (%256)

== const pointers ==

Your Sink has two pointers two images. Assume they are called input1 and input2. Their declaration in Sink.h likely looks like:

```
Image *input1;
Image *input2;
```

You should change them to be const:

```
const Image *input1;
const Image *input2;
```

Adding the const is the easy part … the issue is dealing with any compilation issues.

== What to turn in ==

The standard! … source code & Makefile in a tarball.