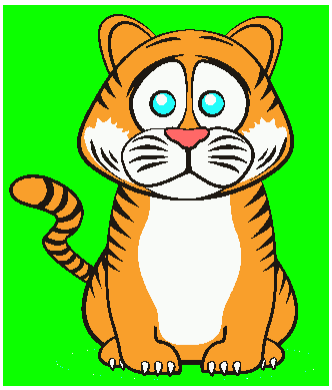
CPSC 4040/6040
Computer Graphics Images

Assigned: Sept. 3, 2014
Due: Sept. 10, 2014 (start of class)

Quiz 1 *(Grading: 0–10 points)*

1. The picture on the right contains (at least) six distinct colors (including the outline and background color). Please identify and give a reasonable RGB specification for each of these colors, assuming that the R, G, and B values are in the range 0.0 to 1.0. Then, give the corresponding six appropriate RGB values, in hexadecimal notation, that you would store in a pixmap that uses three unsigned 8-bit binary numbers to store the color channel values.



2. The code in Listing 1 allocates 2D arrays for storing two pixmaps, `pixmap1` and `pixmap2`. Assume that `pixmap1` receives content from reading some image of size `W`×`H`. Fill in the code for the `for` loop so that `pixmap2` is assigned the pixel values of `pixmap1`, but flipped *vertically*. By this, I mean that the first row of `pixmap2` should be the last row of `pixmap1` and so on.

Listing 1: 2D indexed flip

```
1  struct pixel {
2     unsigned char r, g, b;
3  };
4
5  pixel **pixmap1, **pixmap2;
6  unsigned int W, H;   //assume these are non-zero
7
8  pixmap1 = new pixel*[H];
9  pixmap1[0] = new pixel[W * H];
10 for(int i = 1; i < H; i++)
11    pixmap1[i] = pixmap1[i - 1] + W;
12
13 pixmap2 = new pixel*[H];
14 pixmap2[0] = new pixel[W * H];
15 for(int i = 1; i < H; i++)
16    pixmap2[i] = pixmap2[i - 1] + W;
17
18 //read some file and store in pixmap1
19
20 for(int row = 0; row < H; row++)
21    for(int col = 0; col < W; col++) {
22       pixmap2[row][col].r =
23
24       pixmap2[row][col].g =
25
26       pixmap2[row][col].b =
27
28    }
```

3. Similar to Listing 1, we are allocating two images, and would like to flip them vertically, but in Listing 2 we are instead storing them as 1D arrays of `pixel`'s. Fill in the correct code.

Listing 2: 1D indexed flip

```
1  pixel *pixmap1, *pixmap2;
2  unsigned int W, H;   //assume these are non-zero
3
4  pixmap1 = new pixel[W*H];
5  pixmap2 = new pixel[W*H];
6
7  //read some file and store in pixmap1
8
9  for(int row = 0; row < H; row++)
10    for(int col = 0; col < W; col++) {
11       pixmap2[               ].r =
12
13       pixmap2[               ].g =
14
15       pixmap2[               ].b =
16
17    }
```

4. Assume that you are using a run-length encoding scheme that encodes both runs and non-repeating sequences to compress a greyscale image of 200 × 150 pixels (200 columns, 150 rows). Pixels and run lengths are stored as 8-bit numbers (i.e. with type `char`). Ignoring the size of the image header, answer the following.

   - What is the size (in bytes) of the original uncompressed image?
   - What is the smallest possible size of the compressed image?
   - What is the largest possible size of the compressed image?

5. Different color spaces are more appropriate for different kinds of tasks. What color space would you use if you were writing a program which had to:

   A. Provide an intuitive interface for artists to use?

   B. Provide the ability to print out high quality archival images?

   C. Provide as close as possible replication between how the screen drawns the colors with how the colors are stored? In graphics, this is sometimes called WYSIWYG ("what you see is what you get"), meaning that the appearance image matches the data as closely as possible.

   Give a *separate* answer for A, B, and C and justify your choices. Be sure to think about the devices/media involved as well as the intended purpose.