Gavin McRoy (gmcroy)

Question #1
(.0 - 1.0)
Green (.46, .98, .30)
White (.98, .98, .98)
Black (.10, .09, .07)
Orange(.93, .64, .28)
Pink (.93, .44, .42)
Blue (.47, .98, .98)
(Hexadecimal)
Green #76FA4C
White #F9FBF8
Black #1C1813
Orange #ECA247
Pink   #ED716C
Blue   #78F9F8

Question #2

```
/*Declare a struct called pixels with 3 unsigned char values
(r,g,b)*/
struct pixel {
   unsigned char r, g, b;
};
/* Declare a pointer to a pointer of type pixmap */
pixel **pixmap;

/* Set up the width and height variables */
unsigned int W, H;

int main() {
   /* Allocate memory for our pixmap */
   pixmap = new pixel *[H];
   pixmap[0] = new pixel[W * H];

   /* Sets up the array indexes, such that [2][3] points to the
correct part of the image */
   for (int i = 1; i < H; i++) {
       pixmap[i] = pixmap[i - 1] + W;
   }

   /* Invert the images colors */
   for (int row = 0; row < H; row++) {
       for (int col = 0; col < W; col++) {
```

```
            pixmap[row][col].r = 255 - pixmap[row][col].r;
            pixmap[row][col].g = 255 - pixmap[row][col].g;
            pixmap[row][col].b = 255 - pixmap[row][col].b;
        }
    }
}
```

Question #3

```
pixmap1 = new pixel[W1 * H2];
pixmap2 = new pixel[W1 * H2];

      for (int row = 0; row < H; row++)
        for (int col = 0; col < W; col++) {
            pixmap2[row*H + col].r = pixmap1[(H*(H-row)) + col].r;
            pixmap2[row*H + col].g = pixmap1[(H*(H-row)) + col].g;
            pixmap2[row*H + col].b = pixmap1[H*(H-row) + col].b;
        }
```