Media Computation: Fun with Images!
Suppose we want to **brighten** a picture by **20%**?

- We want to increase the RGB intensity!
- Increase all three color components: R, G, and B
- We can represent the percentage as a decimal fraction
  - new red = old red \( \times (1 + 0.20) \)
  - new green = old green \( \times (1 + 0.20) \)
  - new blue = old blue \( \times (1 + 0.20) \)
What if we want a different percentage?

- To change the brightness by different levels?
- Let’s use a *parameter*!
invert()

Create an invert() method that inverts the red, green, and blue values. For example:

- if the red value is 200, change it to 55
- if the blue value is 20, change it to 235
- if the green value is 150, change it to 105
Create a sepia() method using these:

\[
p.setRed(0.393 \times p.getRed() + 0.769 \times p.getGreen() + 0.189 \times p.getBlue());
\]

\[
p.setGreen(0.349 \times p.getRed() + 0.686 \times p.getGreen() + 0.168 \times p.getBlue());
\]

\[
p.setBlue(0.272 \times p.getRed() + 0.534 \times p.getGreen() + 0.131 \times p.getBlue());
\]
Create a new method called `multicolored()`

How can we apply different operations to different pixels?
 Processing pixels using (x, y) coordinates

```java
for (int x = 0; x < this.getWidth(); x++){
    for (int y = 0; y < this.getHeight(); y++){
        // process pixel at (x, y)
    }
}
```
How do we know when to setGreen() vs. setBlue() vs. setRed()?
Activities

- Call multiple transforms on the same input image
- Create new transform classes based on RGB values
- Make an image rainbowed?