1. (14 pts total) For the following function

```java
float calcGray(int r, int g, int b) {
    float gray = 0.3*r + 0.59*g + 0.11*b;
    return gray/255.;
}
```

a. (2 pts) What is the return type? ______________float___
b. (2 pts) How many parameters are there? ______3_____  
c. (2 pts) What is the type of each parameter? ______int ____
d. (2 pts) What are the names of each of the parameters? ___r, g, and b

e. (6 pts) What needs to go into the two println statements:

```java
void setup() {
    int myRed   = random(255);
    int myBlue  = random(255);
    int myGreen = random(255);
    println( "For red = " + myRed + ", green = " + myGreen + ", and blue = " + myBlue);
    println("The gray scale value is " + calcGray(myRed, myGreen, myBlue));
}
```

to print the output below (note, the choice of numbers written below may be different since we don’t know ahead of time what random numbers will be chosen).

For red = 241, green = 100, and blue = 16
The gray scale value is .5218

You should make use of the calcGray() function.

2. (3 pts each, 18 pts total) Complex Number Representation:
   a. Place the following in standard form a + b i.
      i. 3 3 4 7 3  
          _______7 - 3 i_______
      ii. -6  2 16  
          _________2_______
   b. What is the standard form for the complex numbers whose values in polar coordinates are
      i. (r, θ) = (√2 , 45°)  ___1 + i________
      ii. (r, θ) = (3, 270°)  ___- 3 i_________
   c. What is the polar coordinate representation (r, θ) for the following complex numbers
      i. −3  (r, θ) =_____3, 180°____
      ii. −2 i  (r, θ) =_____2, 270°____
3. (4 pts each, 24 pts total) Complex numbers: Given \( z_1 = -8 + 2i \) and \( z_2 = (1 - 3i) \). Calculate the following, placing the result in standard form
   a. \( z_1 + z_2 = \) __________ - 7 - i
   b. \( z_1 - 2z_2 = \) __________ -10 + 8i
   c. \( z_1 z_2 = \) __________ - 2 + 26i
   d. \( \bar{z}_1 + z_1 = \) __________ -16
   e. \( \bar{z}_1 z_1 = \) __________ 68
   f. Length of \( z_1 = |z_1| = \sqrt{68} \)

4. (8 pts total) Class syntax:
   a. (3 pts) How would you create a new Complex object with real component equal to 4.2 and imaginary component equal to 5.8?

   ```java
   Complex c = new Complex(4.2, 5.8);
   ```

   b. (5 pts) In Processing, suppose you have created complex numbers \( c_1, c_2, \) and \( c_3 \). How do you compute (i.e. what is the syntax for) computing

   \[ c_3 = c_1/c_2 + c_1*c_2 \]

   ```java
   c3 = Complex.cAdd( Complex.cDiv(c1,c2) , Complex.cMult(c1,c2) ) ;
   ```
5. (10 pts) Functions and Classes: Write a Processing function called `zSquared` that takes a Complex number as a parameter and returns a float which is equal to the square of the Complex number's real part plus the square of its imaginary part. That is, if \( z = a + b \, i \), then the function will return \( a^2 + b^2 \) (of course, you need to put this all in Processing syntax).

```java
float zSquared(Complex c) {
    float sq = c.real*c.real + c.imag*c.imag;
    return sq;
}
```

6. (4 pts each, 8 pts total) Rescaling: Given a region of the complex plane where the real and imaginary components range as shown in the figure below:

![Diagram showing the range of real and imaginary values](image)

a. How does one use the map function to determine the pixel location of the complex number \( z = -0.25 + 0.43 \, i \).

```java
int pixeli = map(-.25 , -1.5, 2.5, 0, width);
int pixelj = map(.43, -2.0, 1.0, 0, height);
```

b. How does one use the map function to determine the complex number corresponding to the pixel \((i,j)\)

```java
int real = map( i, 0, width , -1.5, 2.5);
int imag = map( j, 0, height, -2.0, 1.0);
```

7. (8 pts) The following for-loop computes the sum of the numbers from 10 to 100. Write a while-loop that does the same thing.

```java
int sum = 0;
for (int i =10; i <= 100; i++) {
    sum = sum + i;
}
```

```java
int sum = 0;
int i = 10;
while (i <=100) {
    sum = sum + i;
    i++;
}
```
8. (10 pts) Recursion: Write a **recursive** function called `multNums` that will multiply the numbers from 1 to `n`, for some value of `n`. It could be called from the `setup()` function as follows:

```java
void setup() {
    int n = random(20);
    int product = multNums(n);
    println("The product is " + product);
}

int multNums(int n) {
    if (n <= 1) {
        return 1;
    } else {
        return multNums(n-1) * n;
    }
}
```