****

**IFBA – Instituto Federal de Educação, Ciência e Tecnologia da Bahia**

**Departamento de Ciência da Computação**

**Graduação Tecnológica em Análise e Desenvolvimento de Sistemas**

INF008 – Programação Orientada a Objetos

**Prof.:** Frederico Jorge Ribeiro Barboza

**Avaliação 2021.1**

**Instruções (leia com atenção):**

* O trabalho deve ser desenvolvido em equipes de três pessoas.
* A implementação deve ser realizada de forma incremental, à medida que a especificação for sendo liberada
* Sugere-se o uso de uma ferramenta de trabalho colaborativo e versionamento como o git.
* Os commits (ou equivalentes) poderão serem utilizados para avaliar a participação dos membros da equipe

As cores RGB modelam um sistema de cores aditivas em que o Vermelho (**R**ed), o Verde (**G**reen) e o Azul (**B**lue) são combinados de várias formas de modo a reproduzir um largo espectro cromático. Uma cor RGB é representada por uma 3-upla, onde cada elemento pode assumir um valor de 0 a 255, representando a quantidade de vermelho, verde e azul que compõem a cor.

Em cores RGB, a luminosidade é calculada através da expressão de luminosidade = (R\*0.3 + G\*0.59 + B \*0.11), que deve ser truncada para um valor inteiro.

Escreva

1. uma classe que represente uma cor RGB
2. a representação estática da classe criada
3. os métodos que leem e alteram os valores das componentes RGB da cor
4. um método que leia a luminosidade da cor
5. construtores sobrecarregados que permitam a criação de uma cor
   1. um construtor de cópia, que crie uma cor idêntica (mesmo valor de R, G e B)
   2. um construtor sem parâmetros, que crie a cor preta (<R=0, G=0, B=0>)
   3. um construtor que receba três valores como parâmetros e crie uma considerando estes valores como os valores de R, G e B, respectivamente
6. um método que verifique se duas cores são iguais. As cores são consideradas iguais se seus valores de R, G e B são idênticos entre si (o R de uma cor igual o R da outra e assim por diante)
7. altere as propriedades Red, Green e Blue e Luminosidade para serem exclusivamente leitura
8. um método que gere e retorne uma nova cor RGB equivalente ao cinza da cor RGB em questão. Essa cor possui os valores de R, G e B iguais entre si e iguais ao valor da luminosidade da cor original. Por exemplo, para a cor #2596BE, a cor gerada deverá ser #797979.
9. um método que gere uma representação String da cor como hexadecimal. A representação é composta pelo caracter # seguida de dois dígitos representando o valor de R, G e B em hexadecimal. Por exemplo, para a cor RGB, onde R=37, G=150, B=190, o método deve retornar: #2596BE
10. um método clarear que receba um valor e modifique a tonalidade da cor em questão a tornando mais clara. Para isso, o método deve modificar os valores RGB, para um novo valor **p** percentual maior, onde **p** deve ser indicado como parâmetro. Por exemplo, para a cor #2596BE, se o parâmetro de clarear for 0.1 (10%), a cor deve se tornar #3BA1C5.
11. um método escurecer que receba um valor e modifique a tonalidade da cor em questão a tornando mais escura. Para isso, o método deve modificar os valores RGB, para um novo valor **p** percentual menor, onde **p** deve ser indicado como parâmetro. Por exemplo, para a cor #2596BE, se o parâmetro de escurecer for 0.1 (10%), a cor deve se tornar #2187AB.
12. Um método que retorne uma nova instância de CorRGB, igual a cor que recebeu a mensagem
13. Atributos de classe que permitam reduzir o número de instâncias em uso de cores comuns
    1. PRETA #000000
    2. BRANCA #FFFFFF
    3. RED #FF0000
    4. GREEN #00FF00
    5. BLUE #0000FF

Escreva uma classe que represente uma Imagem como um mapa bidimensional de cores RGB.

1. a representação estática da classe criada
2. um construtor que crie uma Imagem. O tamanho da Imagem será passado como parâmetro no construtor. A imagem criada deve ter todos os píxels ajustados para o BRANCO
3. um método que modifique o pixel de uma imagem dada a posição e o pixel
4. sobrecarregue este método para modifique o pixel de uma imagem dada a posição e os valores de RGB do pixel.
5. método que verifique que duas imagens são iguais
6. método que crie uma nova imagem com o equivalente em tons de cinza. Essa imagem deve ter os valores de cada pixel da cor original substituido pelo seu equivalente em cor de cinza.
7. método que verifique se uma imagem é um fragmento da outra