

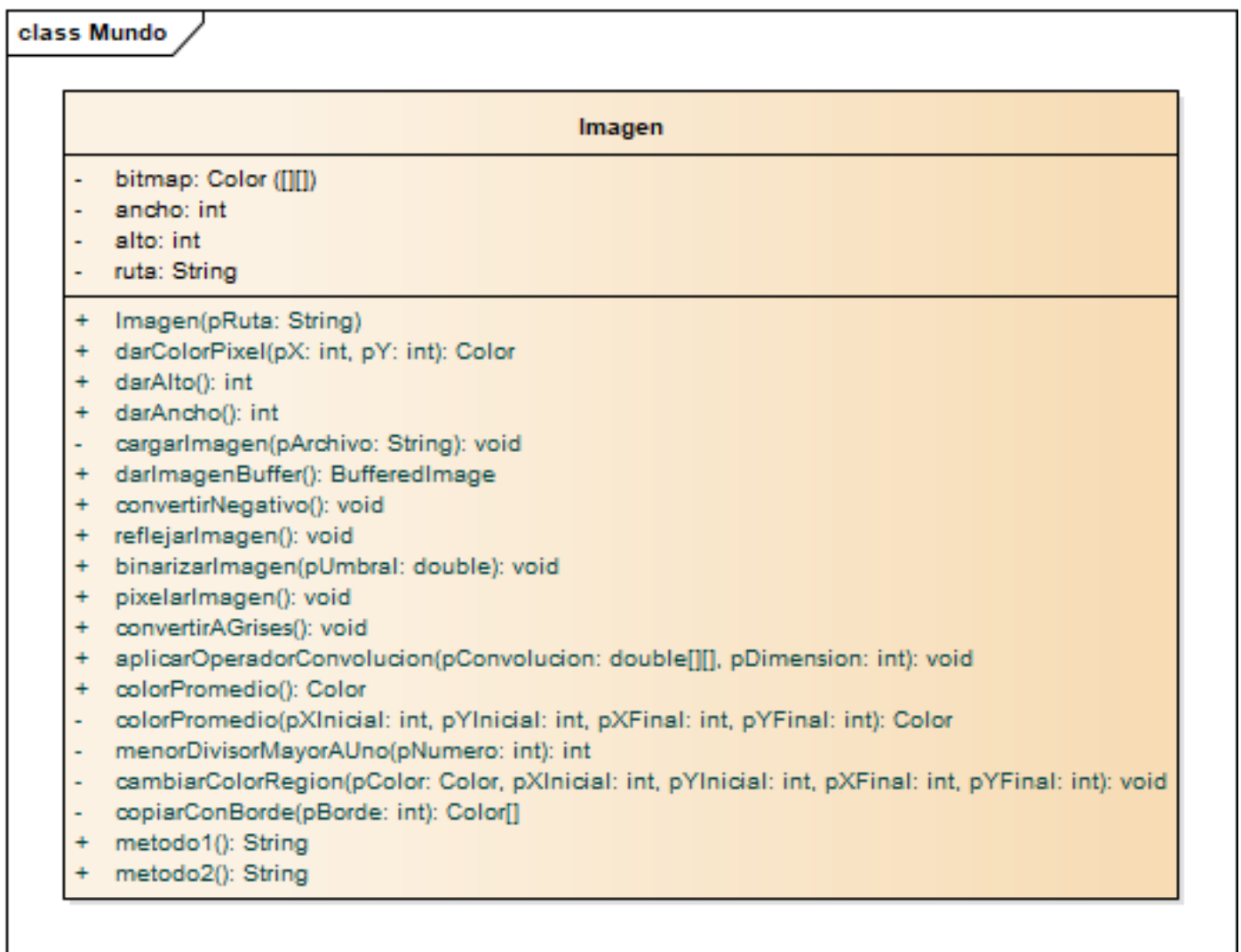
Visor de Imágenes

Hoja de trabajo

Nombre:

Código:

Considere el ejercicio del visor de imágenes, cuyo modelo conceptual se muestra a continuación (puede utilizar cualquiera de los elementos definidos en dicho modelo para completar las actividades solicitadas en este taller):



1. Complete el siguiente método que devuelve el número de píxeles en la imagen cuyo color es igual al dado como parámetro.

```
public class Imagen
{
    //-----
    // Métodos
    //-----

    public int cuantosPixelesColor( Color pColorBuscado )
    {

    }
}
```

2. Complete el siguiente método que indica cuál es la tendencia de color de la imagen. Un píxel tiene un color de tendencia roja, si su índice es mayor que los otros dos. Lo mismo sucede con los demás colores. Este método retorna 0 si la imagen no tiene ninguna tendencia, 1 si la tendencia es roja, 2 si la tendencia es verde y 3 si la tendencia es azul.

```
public class Imagen
{
    //-----
    // Métodos
    //-----

    public int calcularTendencia( )
    {

    }
}
```

3. Complete el siguiente método que implementa un algoritmo de filtrado que calcula un nuevo valor para cada píxel de la imagen. Este valor es calculado como el promedio de los 8 vecinos del píxel en la imagen original, sobre cada uno de los componentes RGB. No se incluyen los bordes de la imagen. Debe retornar una matriz con una copia de la imagen filtrada.

```
public class Imagen
{
    //-----
    // Métodos
    //-----

    public Color[][] imagenFiltrada( )
    {

    }
}
```

4. En algunos contextos, en lugar del color exacto de cada píxel, sólo nos interesa distinguir el fondo de la imagen (en blanco) de otros elementos que puedan aparecer. Complete el siguiente método que modifica la matriz de píxeles así: si la suma de los tres componentes RGB de un píxel es menor que 100, lo debe reemplazar por el color blanco (0,0,0). En caso contrario lo reemplaza por el color negro (255,255,255).

```
public class Imagen
{
    //-----
    // Métodos
    //-----

    public void binarizar( )
    {

    }
}
```

5. Complete el siguiente método que es capaz de rotar una imagen 90 grados a la derecha.

```
public class Imagen
{
    //-----
    // Métodos
    //-----

    public void rotar90AlaDerecha( )
    {

    }
}
```