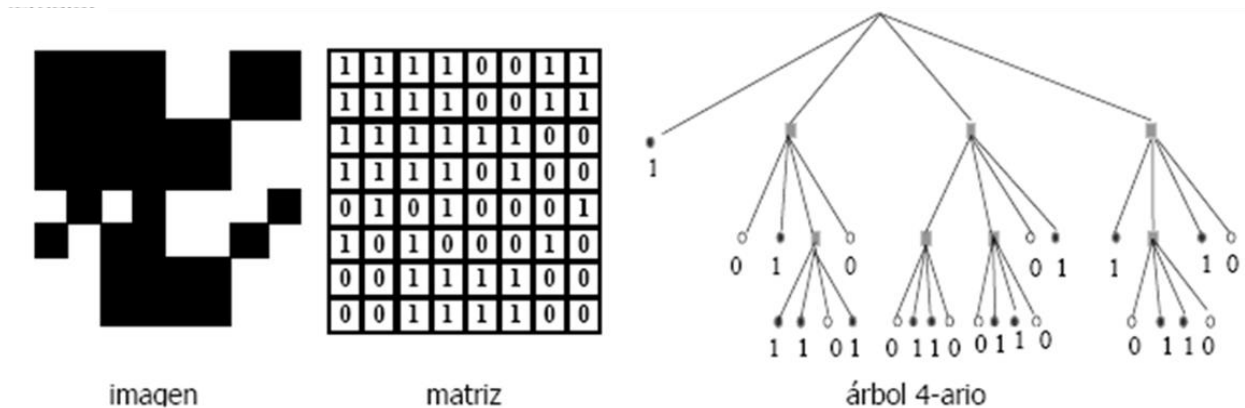
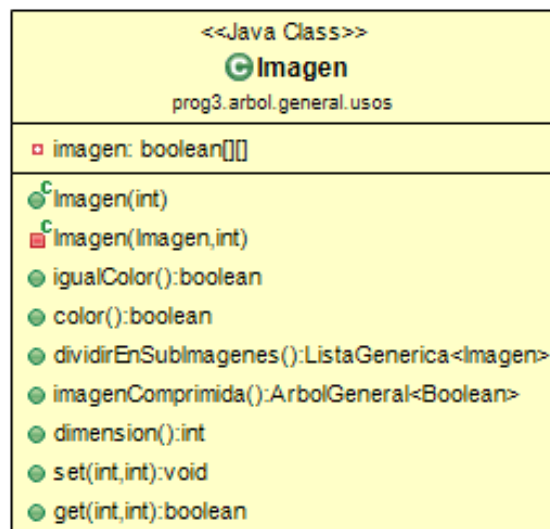


Árbol 4-ario. Una forma de comprimir una imagen (representada como una matriz de puntos) es transformarla a un árbol 4-ario. El algoritmo es el siguiente:

- Si toda la matriz tiene un mismo color, se debe definir un nodo con ese color.
- En caso contrario, se divide la matriz en cuatro partes, se define un nodo con 4 hijos, y cada hijo es la conversión de cada una de las partes de la matriz.



Implemente la clase Imagen de acuerdo al siguiente diagrama:



- Realice un método de instancia llamado **imagenComprimida()** que devuelva la representación de la imagen en su árbol correspondiente (es decir, la matriz de booleanos se representará como árbol general).

NOTA: Teniendo en cuenta que el color blanco lo representamos con el valor falso y el negro con verdadero, implemente los métodos *igualColor()*, *color()* y *dividirEnSubimagenes()* en la clase Imagen.

- El método *igualColor()* devuelve true si la matriz que representa la imagen posee todas sus celdas del mismo color.

- El método *color()* devuelve true si toda la matriz que representa la imagen posee el valor true (negro) y false si toda la matriz posee el valor false (blanco). Este método sólo será invocado cuando la Imagen tiene "igual color".
- El método *dividirEnSubimagenes()* devuelve una Lista con 4 Imágenes.
- Método *imagenComprimida()*: Para comprimir la imagen, debe verificar si la misma posee igual color, en ese caso debe crear y devolver un Árbol General con un único nodo. En cambio, si la imagen no posee el mismo color, debe *dividirEnSubImagenes* y enviar el mensaje *imagenComprimida()* a cada una de las nuevas Imágenes.