

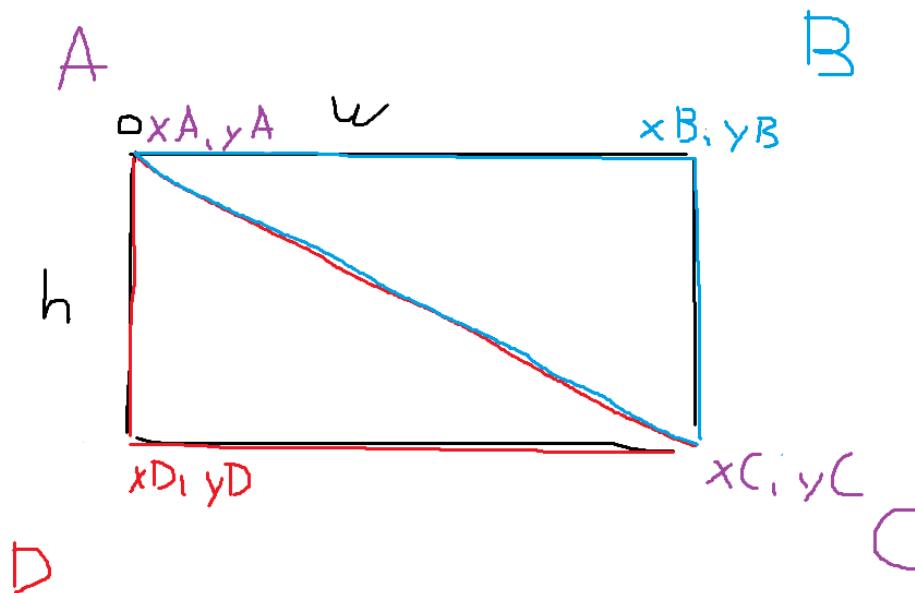
Image Editor Documentation

IMPORTANT!

The github repo already has a picture provided by me for the editor to work... in the given case that the user desires to try the editor with a different image will have to do the following steps:

1. delete the image called "edit.png" inside the /src folder
2. rename the new image to "edit.png"
3. enjoy 😊

Rectangle logic:



General information:

This program allows editing an image (edit.png) through three operations: rotation of a selected region, color inversion in a selected region, and cropping of a selected region. The modified image is saved as editOutput.png in the /src folder, and successive edits accumulate on that file.

CLASS - image

Purpose: Represents the working image, storing its width, height, and the BufferedImage object. After construction, it tries to load editOutput.png first (if it exists), otherwise edit.png.

Attributes:

	TYPE	DESCRIPTION
w	int	Image width
h	int	Image height
image	BufferedImage	Object containing the image pixels

Constructor:

	PARAMETERS	DESCRIPTION
image()	—	Tries to load the image from src/editOutput.png or src/edit.png. Throws IOException if it fails.

Methods:

	PARAMETER	RETURN	DESCRIPTION
setImage	BufferedImage img	void	Assigns the internal image
getImage	—	BufferedImage	Returns the internal image
setWidth	int w	void	Sets the width
setHeight	int h	void	Sets the height
getWidth	—	int	Returns the width
getHeight	—	int	Returns the height

CLASS - points

Purpose: Stores a pair of coordinates (x, y) and provides accessor methods.

Attributes:

	TYPE	DESCRIPTION
x	int	X coordinate
y	int	Y coordinate

Constructor:

	PARAMETER	DESCRIPTION
points	int x, int y	Constructs a point with the given coordinates.

Methods:

	PARAMETERS	RETURN	DESCRIPTION
setPoints	int x, int y	void	Assigns new coordinates
getx	—	int	Assigns new coordinates
setx	int x	void	Modifies the X coordinate

gety	—	int	Returns the Y coordinate
sety	int y	void	Modifies the Y coordinate

CLASS - lambda

Purpose: Contains a utility method to determine if a point lies inside a triangle using barycentric coordinates.

Constructor:

	PARAMETERS	DESCRIPTION
lambda()	—	—

Methods:

	PARAMETERS	RETURNS	DESCRIPTION
isInside	int px, int py, points p1, points p2, points p3	boolean	Returns true if the point (px, py) is inside the triangle formed by p1,p2,p3.

CLASS - InvertColors

Purpose: Inverts the colors of all pixels inside a rectangular region selected by the user

Attributes:

	TYPE	DESCRIPTION
A, B, C, D	points	The four vertices of the selection rectangle
tempX, tempY, tempX2, tempY2	int	Temporary variables to capture user input
sc	Scanner	Scanner to read from console

Constructor:

	PARAMETERS	DESCRIPTION
invertColors()	—	Calls assignPoints() to request coordinates

Methods:

	PARAMETERS	RETURN	DESCRIPTION
assignPoints	—	—	Asks the user for coordinates of corners A (top-left) and C (bottom-right) and builds the four points
paintImg	image editImg	—	Iterates over the entire image, inverts pixels inside the rectangle (split into two triangles) and saves the result to src/editOutput.png

CLASS - rotate

Purpose: Rotates the content of a rectangular region selected by the user by an angle of 90°, 180°, or 270° (counter-clockwise).

Attributes:

	TYPE	DESCRIPTION
A, B, C, D	points	The four vertices of the selection rectangle
tempX, tempY, tempX2, tempY2	int	Temporary variables to capture user input
sc	Scanner	Scanner to read from console

Constructor:

	PARAMETERS	DESCRIPTION
rotate()	—	Calls assignPoints() to request coordinates

Methods:

	PARAMETERS	RETURN	DESCRIPTION
assignPoints	—	—	Asks the user for coordinates of corners A (top-left) and C (bottom-right) and builds the four points
rotateImg	image editImg	—	Asks for the angle, creates a copy of the image, clears the original region (paints it black), and applies an inverse rotation to paint the rotated rectangle. Saves to src/editOutput.png

CLASS - crop

Purpose: Crops the image to the rectangular area selected by the user and updates the image object with the new cropped image.

Attributes:

	TYPE	DESCRIPTION
A, B, C, D	points	The four vertices of the selection rectangle
tempX, tempY, tempX2, tempY2	int	Temporary variables to capture user input
sc	Scanner	Scanner to read from console

Constructor:

	PARAMETERS	DESCRIPTION
crop()	—	Calls assignPoints() to request coordinates

Methods:

	PARAMETERS	RETURN	DESCRIPTION
assignPoints	—	—	Asks the user for coordinates of corners A (top-left) and C (bottom-right) and builds the four points
cropImg	image editImg	—	Obtains the subimage defined by the rectangle, updates the image object with it, and saves to src/editOutput.png

CLASS - Main

Purpose: Entry point of the program. Displays an interactive menu and executes the selected operation.

Attributes:

	TYPE	DESCRIPTION
sc	Scanner	Scanner to read input

Methods:

	PARAMETERS	RETURN	DESCRIPTION
main	—	—	Creates the image object, shows the menu, and based on the chosen option instantiates and runs the corresponding class (rotate, invertColors, crop) or exits

Relationships Between Classes:

- *Main* instantiates *image* and, depending on the option, creates objects of *rotate*, *invertColors* or *crop*.
- *rotate*, *invertColors* and *crop* receive the *image* object to modify it.
- *rotate*, *invertColors* and *crop* use *points* to represent coordinates.
- *rotate* and *invertColors* use *lambda* to determine whether a pixel lies inside the selected area (split into two triangles).