The task is to implement a utility to perform simple image manipulation. The utility should be able to resize, rotate, or flip an image or perform a combination of the aforementioned operations. The first command line argument of your programme will be the input image filename, and the second argument should be the name of the output file. The rest of the command line arguments will consist of a sequence of ‘enlarge’, ‘shrink’, ‘rotate’, ‘flip’ keywords. These will define the operations your utility is supposed to perform in the order they are provided on the command line.

The **‘enlarge’** command will resize the image to twice its original size, **‘shrink’** will scale down the image to half of its original size (or half the size minus 0.5 in case the size is odd). The **‘rotate’** command will rotate the image counterclockwise by 90° and **‘flip’** will flip the image along its horizontal axis. You will be provided with a header file and source code capable of creating, loading and saving a bitmap image in bmp format.

Example: “**./main a.bmp b.bmp rotate rotate flip shrink**” should rotate the image by 180˚ and then flip it along the horizontal axis and finally resizing it to half the original size. This sequence of operations will cause the resulting **b.bmp** to be half the size of **a.bmp** and horizontally mirrored.
You will be provided with a static library **libsimage** and a corresponding header **simage.h**, which contains functions for image loading and saving. You will be provided with an example image as well.

The C99 documentation and lecture slides are available to you on the course wiki.

You have to zip and upload your solution to the brute system.

Suggested, but optional walkthrough:

1) Download and unzip the assignment files (e.g. by "**`wget https://cw.fel.cvut.cz/wiki/_media/courses/be5b99cpl/lectures/test_cpl_2025-02-13.zip`**"). Check their structure and create a compilable `main.c`, which reports if the command line arguments are not in the expected format. **(1 point)**

2) Create the functions for the flip, enlarge, shrink and rotate functionalities (let them just print what they should do) and make sure that they are called according to the command line arguments. **(1 point)**

3) Write a Makefile that compiles the main program and links it with the **simage** library. Make sure that functions have comments regarding their arguments and what the function does. Correct any compiler warnings/errors when compiling with "`-Wall -Werror -std=c99 -pedantic`" flags. **(1 point)**

4) Modify your main.c so that it can load and save an image using the provided 'simage.h' and 'simage.cpp' codes. Check the saved image. **(1 point)**

5) Implement the horizontal 'flip' functionality. If the flip is called twice, the image should remain the same. **(2 points)**

6) Implement the 'enlarge' functionality. The image should be enlarged twice with each call. Do not worry about pixel interpolation. **(1 point)**

7) Implement the 'shrink' functionality. The image should be resized with each call. Do not worry about aliasing effects or interpolation. **(1 point)**

8) Implement the 'rotate' functionality. The image should be rotated with each call. **(2 points)**