In this project, we will write the program code for an image processing algorithm. The adaptive two-pass rank filter by Xu and Miller (see *Advanced Biomedical Image Analysis* Chapter 5.2.4.) is ideally suited to gain an understanding of the elements of image processing software.

You are free to use the programming language of your choice. If you choose to use the C language, a GTK-based GUI will be available. It is not the purpose of this homework to design a perfect GUI. Rather, variables such as file names or the user-selectable parameters $a$ and $b$ may be "hard-wired" into the code. You are also free to program variations of the algorithm if you believe that your variation is more advantageous, but you need to provide a reason.

The idea behind the adaptive two-pass rank filter is to reverse potential over-correction by the conventional median filter. Therefore, it is reasonable to program a median filter in the first step. You need a duplicate of the original image $I$ (or an empty spaceholder of the same size). Write a function that loops over every pixel of a 2D grayscale image and performs these steps:

- Collect all 9 values of the 3x3 neighborhood of the pixel
- Sort these 9 values ascendingly
- Select the median value from the list of 9
- place the median value into the corresponding pixel of the image copy

After the loop is complete, the new image, let’s call it $F$, is the output of the non-adaptive median filter. Make sure that your program handles boundary pixels correctly. You cannot access pixels at negative coordinates or pixels beyond the highest row or column. Assuming a $M$ by $N$ image with the
indices $x$ and $y$ running from 0 to $M - 1$ and 0 to $N - 1$ \footnote{Note that Octave and Matlab use indices from 1 to N rather than 0 to N-1}, respectively, choose one suitable method. Options are

- Zero-padding: Any out-of-range pixels are assumed to be zero.
- Flat continuation: Out-of-range pixels are assumed to be identical to the last in-range pixel. For example, $I(x, -k) = I(x, 0)$ for $k > 0$.
- Tiling: The image is assumed to continue periodically, i.e., $I(x, -k) = I(x, N - k)$ for $k > 0$.
- Specifically for the median filter, the central value may be repeated for each out-of-bounds pixel (preferred method)

It may be a good idea to have a function that returns the image value at coordinates $(x, y)$ even for $x, y < 0$, for $x \geq M$ or for $y \geq N$. Test your median filter thoroughly, for example, by comparing its output with the output from another program.

The next step is to program the second stage of the adaptive two-pass rank filter. In the second stage, only pixels that have been changed by the median filter are examined, and the filter action possibly reversed. For this purpose, create another image matrix $D$ that contains binary values, namely, $D(x, y) = 1$ when $I(x, y) \neq G(x, y)$ and $D(x, y) = 0$ otherwise. Duplicate $D$ into a new image matrix $E$ to hold the pixels where the filter action is to be reversed. Next, create the row vectors $w$ and $e$ with $M$ elements. Calculate $w$ as the the row sums of $D$ normalized by $N$. This allows you to obtain the two thresholds $\eta$ and $\kappa$ for each row, and to compute the number of filtered pixels $K$ to be reversed. Calculate the row vector $e$ as the pixel-by-pixel difference $(I - F)^2$. Identify the $K$ smallest values in $e$ (see the book for definition of $K$). In each row, set all elements of $E$ to zero if the corresponding element in $e$ is among the $K$ smallest. Note that you can only include those elements of $e$ in the set of $K$ smallest elements if the corresponding value of $D$ is nonzero, that is, zero entries of $e$ need to be ignored. At this time, $E$ contains values of 1 only at pixels where the median filter did not overcorrect. Therefore, in the last loop over the images, copy the values of $F(x, y)$ into $I(x, y)$ if, and only if, $E(x, y) = 1$. The original image matrix $I$ is the output of the entire filter.
Test your filter. Use test images, such as the Shepp-Logan phantom with shot noise, a regular image (e.g., CT or MRI slice) spoiled with artificial shot noise, and a regular noisy image without noise addition. Choose different levels of noise. Compare your filter to the conventional median filter and the CWMF.