

1. 첨부된 영상을 `imread` 함수를 이용하여 흑백으로 읽어들이십시오.

(이때 `float(CV_32F)`로 변환하지 않습니다.)

2. `median filtering`을 수행하기 위한 함수를 만듭니다.

3. 함수의 형식은 `myMedianFilter(InputArray input, OutputArray output, int windowSize)` 로합니다.
이때 `windowSize`는 양의 홀수이어야 합니다.

3.1 먼저 상, 하, 좌, 우의 범위를 계산합니다. $\text{범위} \times 2 + 1 = \text{windowSize}$ 가 되도록 범위를 계산합니다.

3.2 입력 영상과 같은 크기의 결과 `image`를 만들고,

3.3 결과 `image`의 모든 픽셀에 대하여

3.4 입력 영상의 해당 픽셀 및 주변의 범위에 해당하는 픽셀의 밝기 값들을 배열에 모읍니다.

주의: `float`로 변환하지 않고 밝기 값을 읽었으므로 `data type`은 `uchar` 입니다.

여기서, `padding`이 적용되도록 (입력 영상의 범위를 벗어나지 않도록) 적절한 조치가 필요합니다.

(Hint 이때 일반 배열 대신 `std::vector`를 사용하면 편리합니다. `std::vector`의 사용법은 `cppreference`를 참고하십시오.)

3.5 범위내의 픽셀들의 밝기 값들을 정렬합니다.

(Hint. 이때 `std::vector`를 사용한 경우라면 `std::sort`를 사용하면 편리합니다.)

3.6 정렬 된 밝기 중 가운데 값을 찾아서 결과 영상의 해당 픽셀에 대입합니다.

4. 함수의 결과로 얻은 결과 `image`를 `imshow` 합니다.

다양한 `windowSize`를 `test` 하시되, 최종적으로 `windowSize`가 5인 경우로 조정하고 제출합니다.

***** 주의 *****

1. OpenCV 내에 `medianFilter`가 존재합니다. 하지만, 이번 과제는 해당 함수를 직접 구현하는 것입니다.

(다만, 비교 용도로 사용하시는 것은 괜찮습니다.)