H02A5a - Computer Vision
Assignment 2: segmentation

1. Intro
Counting cells in histological images is a tedious yet important task for various laboratory examinations: e.g. counting the number of white blood cells.

We want to automate this task using computer vision techniques.

2. Cell counting in histological images (2h)
On Toledo, a skeleton for the Python code and a reference image are provided. We are interested in counting the number of red cells.

The outline of the approach is:

1. Make use of Canny edge detection (provided in OpenCV) to detect the edges of the cells.
2. Use a Hough transform to detect circles (provided in OpenCV). You'll notice that this function combines the Canny operator and the Hough transform in one function. Make sure you understand why this is done. You'll notice that not every found circle corresponds to a cell of interest – this is normal.
3. Calculate for each found circle a feature vector (the average color in the circle). Color can be represented in RGB-values or HSV-values. Which one is more suitable? hsv is more reasonable=>hue, sat, due
   Tip: OpenCV imread() returns 3D numpy array (y,x,channel). There are three channels: B, G, R.
4. Manually determine a lower- and upper-boundary for each feature, and remove the circles that have features outside that interval.
5. Submit your solution (just the python file!) on Toledo using the Assignments-tool

Some remarks:

- You'll notice that the final result is not perfect: there are both undetected cells (false negatives) as falsely found cells (false positives). This is somewhat typical for computer vision: even problems that seem to be fairly simple are very hard to do perfectly. To achieve better results, typically more features are used and more variation on the shape is allowed.
- Even with only 3 features, finding the optimal thresholds is a difficult task. Additionally, we don't know how well these values perform on another image. Now you see why machine learning is used so frequently in computer vision. Supervised learning techniques (e.g. support vector machines) allow finding the optimal parameters, even with hundreds of features (and takes into account the correlation of the features!).