

## BME 51:186 — Homework 5

**DUE: November 3, 2004 at the start of class**  
**No late homework accepted!**

In this assignment you will create and test image processing functions for connected components labeling. You can find images for testing in the `homework5` directory. Keep in mind we might choose to test your algorithm on different images, so be sure to test things carefully.

For each of the parts described below, handle common error conditions such as no image loaded, wrong image type loaded, and bad input parameter value by detecting these conditions and displaying an error dialog.

Because these problems require that you devise an algorithm to solve a problem, part of your grade will be based on how well you comment the source code that describes the algorithm. If we can't understand your source code and comments, you will not receive full credit, even if the algorithm works.

**How to turn in your assignment:** All problems must be completed using the MILP framework discussed in class. To submit the homework, use the same procedure you used for homework two, including the HTML page describing your results.

1. Write a function that takes a 2D binary image (zero and 255 pixel values only) and converts it into either a 4-connected or 8-connected grayscale labeled image using the equivalence table algorithm. The connectivity value should be entered through a dialog box. Assign component graylevels so the largest object is labeled as 255, the next largest is 254, etc. Make your function run as operation 01 in the MILP framework. **NOTE: YOU MUST HANDLE RUNTIME ERROR CONDITIONS PROPERLY, INCLUDING THE CONDITIONS OF TEMPORARILY AND PERMANENTLY RUNNING OUT OF LABELS.**
2. Once we have connected components labeling working, other useful operations are possible. Devise an algorithm that takes a 2D binary image containing a single object that might contain interior holes (cavities) and returns the object with all of the holes filled in. Hint: you can do this by combining connected components labeling with a set complement operation. Make your function run as operation 02 in the MILP framework.
3. Connected components labeling can also be used to set up for object size filtering. Write a function that takes two size parameters `low` and `high`, a connectivity value (4 or 8) from a dialog box (or dialog boxes) and a binary input image, and removes all 2D connected components from the image that have an area less than `low` or greater than `high`. After applying your function, the image should still be a binary image. Make your function run as operation 03 in the MILP framework.