1.) Chan-Vese Segmentation

For an input grayscale image $f$, the Chan-Vese model segments the image with a contour $\Gamma$ by minimizing the energy

$$
\min_{\Gamma} E_{CV}[\Gamma|f] = L(\Gamma) + \lambda_{in} \int_{inside \, \Gamma} (f - c_{in})^2 \, d\tilde{x} + \lambda_{out} \int_{outside \, \Gamma} (f - c_{out})^2 \, d\tilde{x}
$$

where the average gray values are given by

$$
c_{in} = \frac{\int_{inside \, \Gamma} f \, d\tilde{x}}{\int_{inside \, \Gamma} d\tilde{x}} \quad \quad c_{out} = \frac{\int_{outside \, \Gamma} f \, d\tilde{x}}{\int_{outside \, \Gamma} d\tilde{x}}
$$

To track the contour $\Gamma$, we examine the zero level curve of a level set function $\phi$. We will assume the level set function satisfies $\phi > 0$ inside $\Gamma$ and $\phi < 0$ outside $\Gamma$. Rewriting the energy in terms of $\phi$, calculating the first variation of energy, and applying steepest descent means we should evolve the PDE:

$$
\frac{\partial \phi}{\partial t} = \delta(\phi) \left[ \frac{\phi_{xx}\phi_{yy} - 2\phi_x\phi_y\phi_{xy} + \phi_{yy}\phi_{xx}}{(\phi_x^2 + \phi_y^2)^{3/2}} - \lambda_{in}(f - c_{in})^2 + \lambda_{out}(f - c_{out})^2 \right]
$$

It would help to smooth the Dirac delta by

$$
\delta_\epsilon(t) = \frac{\epsilon}{\pi(\epsilon^2 + t^2)}
$$

Your goal is to code up Chan-Vese segmentation using level sets. We generally assume the parameters $\lambda_{in} = \lambda_{out}$. For my version, I used the parameters $\lambda_{in} = \lambda_{out} = 0.1$, $\epsilon = 0.1$, $\Delta t = 0.2$, and stopping time $T = 10$. But you should definitely experiment with different values of $\lambda$. You could initialize your level set function $\phi$ as a big box around the perimeter of the image with:

$$
[m,n] = \text{size}(f); \quad \phi(1:m,1:n)=-1; \quad \phi(2:m-1,2:n-1)=1;
$$

You should update the average values $c_{in}$ and $c_{out}$ at the start of each iteration. You can determine which pixels are inside $\Gamma$ by finding where $\phi > 0$. In Matlab, you can create a 0-1 mask indicating these pixels by: $\text{I} = [\phi > 0]$;

Note the first term of the PDE inside the brackets is the same as the smoothing term in TV denoising. It may help to modify your TV denoising code from Lab 2. Or you can use my version of the TV code online at: www.math.ucla.edu/~wittman/Fields

Try drawing the contour every iteration. To draw a contour $\phi$ on top of the image $f$, use the Matlab commands:

```matlab
imagesc(f);  colormap gray;
hold on;  contour(phi, [0,0], 'r');  hold off;  drawnow;
```
2.) Level Set Initialization
Choose a grayscale image that has at least 2 interesting objects to segment. Experiment with different initializations for the level set function $\varphi$. See if you can "trick" the algorithm into segmenting just one object by starting the contour in different parts of the image.

Note the Matlab `contour` function is a little temperamental, so try not to go too crazy with your initialization. Smooth level set functions tend to work better. Try creating a smooth "checkerboard" pattern of positive and negative values as an initialization for $\varphi$.

3.) Extension to Color Images
Extending the Chan-Vese model to color images is fun and fairly easy. We simply replace the parentheses in the 2 fidelity terms with the L2-norm:

$$
\min_f E_{CV}[\Gamma|f] = L(\Gamma) + \lambda_{in} \int_{\text{inside } \Gamma} ||f - c_{in}||^2 d\vec{x} + \lambda_{out} \int_{\text{outside } \Gamma} ||f - c_{out}||^2 d\vec{x}.
$$

You should not have to alter the derivatives on $\varphi$ because the input image $f$ becomes 3-dimensional, but the level set function $\varphi$ is still 2-dimensional. The norm is taken across the 3rd dimension (color). Note your average colors $c_{in}$ and $c_{out}$ should now be 3-dimensional vectors indicating RGB values. One way you could implement this squared L2-norm in Matlab is:

```matlab
lambda * ( (f(:,:,1)-c(1)).^2 + (f(:,:,2)-c(2)).^2 + (f(:,:,3)-c(3)).^2)
```

It may be harder to see the red contour on a color image, so choose a color test image with little red or change the color of your contour.

4.) A Mysterious Term
Modify the PDE for Chan-Vese segmentation by adding one more term:

$$
\frac{\partial \varphi}{\partial t} = \delta(\varphi) \left[ \varphi_{xx} \varphi_y^2 - 2\varphi_x \varphi_y \varphi_{xy} + \varphi_{yy} \varphi_x^2 \right] - \lambda_{in} (f - c_{in})^2 + \lambda_{out} (f - c_{out})^2 + \beta
$$

Try experimenting with different values of $\beta$, including positive, negative and zero. Can you explain how the value of $\beta$ affects the segmentation result?

Try to reason through the variational approach backwards. What does the $\beta$ term become in the original energy? What is the physical meaning behind this term? Can you think of situations where this extra term would be useful?