Vanishing Point Detection

1. **Create a directory named   HW08__LastName__FirstName**
   Do all your work in that directory. Then, when you submit your work, zip up the entire directory (from the directory name on down). This way when it is unzipped after downloading from MyCourses, we will see 38 directories, with unique names that are easy to understand.

   Use the LastName that matches the last name assigned to you in MyCourses, and the first name assigned to you in MyCourses.

2. **Changing your path:**
   When you read in an image, such as the `cameraman.tif` image, Matlab searches your **path**. The path is a list of executable directories. To make life easy for the grader, you are going to put your test images in a standard location, and change your **path** to include that directory.
   a. Put our images in a directory that at the same level as the directory HW08__LastName__FirstName, and name this directory `TEST_IMAGES`.
   b. In your programs, change the path to include this location using the command:
      `addpath( '../TEST_IMAGES' );`
   c. Do not re-submit the test images. The grader will have copies of them.

3. Write one function that runs your entire image chain. The grader is going to run your program given a filename of an image. The image is similar to the ones provided for you to use as a test case. However, the image will be different. There may be a different number of pieces of melon there, but the camera and the lighting will be the same. The orientation of the pieces of melon will be the same – with the flesh of the melon always on the right of the skin.

   a. The function is to be in a file named `HW08_Lastname_Firstname_MAIN( filename )`
      `Where filename is the name of an input image with a vanishing point in it.`
      `This file should be in your ../TEST_IMAGES directory.`

   b. This function will:
      a) Display the input image
      b) Display where the vanishing point is with a red circle, a green plus sign, and a blue cross.

4. Remember to comment your code well to show that you understand it. Write your comments in your own words. Lack of comments will cost points. In addition to your code, submit a PDF write-up describing your image chain. The grader might only look at your code, or might just look at your write-up, or both.

5. **Discussion:**
   We discussed this extensively in class for 40 minutes. The trick to getting the hough( ) detector to work well, is to put in the correct edges.

   Previous versions of the Professor's code just tested for edges at the angles in the range of [-135, -45, +45, and +135]. For each angle a tolerance of +/- 22 degrees was used. This worked fine for most images except the ones which were rotated by 45 degrees.

   In your write-up, we are looking for evidence that you understand the approach to what you are doing.

   Some students have been doing the bare minimum, such as answering questions with "no" or "see code."
   Or they answer only the questions suggested in the homework. The grader is free to assign a minimum grade to such answers.

   Imagine that you are doing this to explain it to next semester's class. Feel free to exceed expectations. The grader is able to assign a grade that exceeds expectations for well written write-ups, and good working code.

   Your code will be tested with a test image that it hasn't seen before. It is not in the set of images that were given to you. However, they were taken under similar circumstances, with the same camera, and lighting.

6. **What to Submit:**

   A PDF write-up, describe your imaging chain.
   This PDF should describe the algorithm(s) you used completely.

   Submit your code.

   All files should start with the prefix "`HW08_Lastname_Firstname_`" and be in one directory named HW08_Lastname_Firstname.

   Zip up the entire directory so that when it is unzipped it is a directory. **Test** your zip file to be sure it unzips to a directory and not into a set of files.

   The grader will run your homework file HW08_Lastname_Firstname_MAIN(filename) on an image that is similar to the images provided for testing, but not one of the test images.

Other hints:
I recommend trying all of the provided images when developing your algorithm(s).