**CSC IP (381): Project #7 <Edge Detectors>(C++)**

**Student name: Jian Chen**

**Due Date: 03/29/2017**

**Submit Date: 03/29/2017**

**Algorithm Steps for the implementation for this project:**

step 0: open the image and read the image header

dynamically allocate mirrorFramedAry and all the edge arrays

step 1: loadImage // load input file to imgAry

step 2: mirrowFramed (mirrorFramedAry)

step 3: process the mirrorFramedAry, from left to right and top to bottom

begin at (1, 1) // process all pixels!!!

SobelVertical(i,j) <-- abs( convolute (i,j, maskVertical))

SobelHorizontal(i,j) <-- abs( convolute (i,j, maskHorizontal))

SobelRightDiag(i,j) <-- abs(convolute (i,j, maskRightDiag))

SobelLeftDiag(i,j) <-- abs (convolute (i,j, maskLeftDiag))

GradiantEdge(i,j) <-- computeGradiant(i,j)

step 4: repeat step 3 until all pixels are processed.

step 5: write the results of the four Sobel edge arrays to argv[2],argv[3], argv[4], argv[5]

write the result of the gradiant edge array to argv[6]

step 6: close all files