**CSCI 323-33 K-mean Image (C++)**

Han Wen Loh

Due date: Nov. 13, 2018

Algorithm steps:

Step 0: inFile 🡨 Open the input file argv[]

numRows, numCols, minVal, maxVal🡨 get from inFile.

imgAry 🡨 Dynamically allocate a 2-D arrays, size numRows X numCols.

outFile\_1, outFile\_2, outFile\_3 🡨 open from argv[], argv[], argv[]

K 🡨 ask user from console

Kcentroids[K] 🡨 Dynamically allocate centroids struct, size of K+1

Step 1: kmean 🡨 Kmean

Step 2: kmean->extractPts

Step 3:kmean->loadPointSet(inFile1)

Step 4: kmean->kMeanClustering(inFile2)

Step 5: kmean->writePtSet(outFile3);

Step 6: close all input