**CSCI 381 – Computer Vision (C++)**

**Program: Project 1.1: computeHistogram**

**Name: Isaac Gordon**

**Due Date:**

**Soft copy: 2/04/2019 Monday before midnight**

**Hard copy: 2/05/2019 Tuesday in class**

step 0: make sure all inputted arguments are valid

step 1: open input and output files

step 2: numRows, numCols, minVal, maxVal <-- get from input file

dynamically allocate the hist array and initialize to 0

step 3: // process the input file from left to right and top to bottom

by putting all value into an array

step 4: // process the array from left to right and top to bottom

p(i,j) <- read from input // you must read one integer at a time

hitogram[p(i,j)]++

step 5: repeat step 4 until the file is empty

step 6: output histogram array to output file // follow the format given

step 7: close input file and output file

**CODE:**

#include <fstream>

#include <iostream>

#include<string>

using namespace std;

void computeHistogram(int \*\*hist, int \*\*data, int rows, int cols){

for(int r = 0; r < rows; r++){

for(int c = 0; c < cols; c++){

int val = data[r][c];

hist[val][1]++;

}

}

}//computeHistogram

bool endsWith(string str, string ex){

int pos = str.find(ex);

if(pos != str.size() - 4) return false;

return true;

}//endsWith

int main(int argc, char \*argv[]){

//set arg error message

string BAD\_ARGS = "Correct arguement format is \"<inputFile> <outputFile>\".\nBoth should end in \'.txt\'.";

//check for correct number of args

if(argc != 3){

cout << "Wrong number of arguements.\n" << BAD\_ARGS << endl;

exit(1);

}//if

//make sure they are all text files

for(int i = 1; i < argc; i++){

if(!endsWith(argv[i], ".txt")){

cout << argv[i] << " is not a .txt file. Try again." << endl;

exit(1);

}//if

}//for

//declare data structs

ifstream inFile1;

ofstream outFile1;

inFile1.open(argv[1]);

outFile1.open(argv[2]);

//if input file could not be opened then exit

if(!inFile1.is\_open()){

cout << "Input File could not be opened. " << endl;

exit(1);

}//if

//if output file could not be opened then exit

if(!outFile1.is\_open()){

cout << "Output File could not be opened. " << endl;

exit(1);

}//if

//get header values

int numRows, numCols, minVal, maxVal;

inFile1 >> numRows;

inFile1 >> numCols;

inFile1 >> minVal;

inFile1 >> maxVal;

int header[] = {numRows, numCols, minVal, maxVal};

//make 2D array of pixel vals

int\*\* intData = new int\*[numRows];

for(int i = 0; i < numRows; i++) intData[i] = new int[numCols];

//fill data array

for(int r = 0; r < numRows; r++){

for(int c = 0; c < numCols; c++){

inFile1 >> intData[r][c];

}//inner for

}//outer for

//create empty histogram

int \*hist[maxVal + 1];

for(int i = 0; i < maxVal + 1; i++) hist[i] = new int[2];

//fill hist with zeros

for(int r = 0; r < maxVal + 1; r++) hist[r][1] = 0;

//compute the histogram

computeHistogram(hist, intData, numRows, numCols);

//print histogram to outfile

for(int i = 0 ; i < 4; i++) outFile1 << header[i] << " ";

outFile1 << endl;

for(int r = 0; r < maxVal + 1; r++){

outFile1 << r << " " << hist[r][1];

outFile1 <<endl;

}//for

outFile1 << endl << endl;

//close files

inFile1.close();

outFile1.close();

return 0;

}

**OUTPUT:**

31 40 0 9

0 313

1 294

2 196

3 64

4 0

5 0

6 6

7 102

8 124

9 141