**CS381-37: Project 2.2 (CPP)**

**Yida Tao**

**Due date: Feb. 15, 2018**

Algorithm Steps:

step 0: - open the input file and output file

- read the image header, the four numbers

- dynamically allocate mirrorFramedAry

- dynamically alloicate tempAry

step 1: read the input file and load onto mirrowframeAry begin at [1,1]

step 2: mirrowFramed the mirrorFramedAr

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

begin at (1, 1)

- neighborAry <- loadNeighbors // get mirrorframedAry[i,j]'s 3 X 3 neighborhoods

- tempAry[i,j] <-- sort neightborAry using selection sort algorithm,

stop after the fifth smallest item is found,

then return the fifth item

- keep tracking the newMin and newMax of tempAry

step 4: repeat step 3 until all pixels are processed

step 5: output the image header (numRows, numCols, newMin, newMax) to outfile

step 6: output the tempAry, begin at [1,1], to outfile

step 7: close input and output files

**Source Code**

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

class Median3X3 {

private:

int numRows;

int numCols;

int minVal;

int maxVal;

int newMin;

int newMax;

int \*\*mirrorFramedAry;

int \*\*tempAry;

int neighborAry[9] = {};

public:

//constructor

Median3X3(string input){

numRows = 0;

numCols = 0;

minVal = 0;

maxVal = 0;

newMin = 99;

newMax = 0;

ifstream inFile;

inFile.open(input);

inFile >> numRows;

inFile >> numCols;

inFile >> minVal;

inFile >> maxVal;

mirrorFramedAry = new int\*[numRows+2];

tempAry = new int\*[numRows+2];

for(int i = 0; i < numRows+2; i++){

mirrorFramedAry[i] = new int[numCols+2];

tempAry[i] = new int[numCols+2];

for(int j = 0; j < numCols+2; j++){

mirrorFramedAry[i][j] = 0;

}

}

int num = 0;

int counter = 0;

int r = 0;

int c = 0;

while(inFile>>num){

r = counter/numCols + 1;

c = counter%numCols + 1;

mirrorFramedAry[r][c] = num;

counter++;

}

inFile.close();

}

//destructor

~Median3X3(){

for(int i = 0; i < numRows + 2; i++){

delete mirrorFramedAry[i];

delete tempAry[i];

}

delete[] mirrorFramedAry;

delete[] tempAry;

}

void mirrorFramed(){

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

mirrorFramedAry[0][i] = mirrorFramedAry[1][i];

mirrorFramedAry[numRows + 1][i] = mirrorFramedAry[numRows][i];

}

for(int j = 0; j < numRows + 2; j++){

mirrorFramedAry[j][0] = mirrorFramedAry[j][1];

mirrorFramedAry[j][numCols + 1] = mirrorFramedAry[j][numCols];

}

}

void medianFilter(){

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

for(int j = 1; j < numCols + 1; j++){

tempAry[i][j] = loadNeighbors(i,j);

if(tempAry[i][j] < newMin){

newMin = tempAry[i][j];

}

else if(tempAry[i][j] > newMax){

newMax = tempAry[i][j];

}

}

}

}

int loadNeighbors(int r, int c){

int index = 0;

for(int i = -1;i < 2; i++){

for(int j = -1; j < 2; j++){

neighborAry[index] = mirrorFramedAry[r+i][c+j];

index++;

}

}

sortHalfAry();

return neighborAry[4];

}

void sortHalfAry(){

for(int i = 0; i < 5;i++){

int min = i;

for(int j = i + 1; j < 9; j++){

if(neighborAry[j] < neighborAry[min]){

min = j;

}

}

if(min != i){

int temp = neighborAry[i];

neighborAry[i] = neighborAry[min];

neighborAry[min] = temp;

}

}

}

void print(){

cout<< numRows << " " << numCols << " " << newMin << " " << newMax << endl;

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

for(int j = 1; j < numCols + 1; j++){

cout << tempAry[i][j]<<" ";

}

cout << endl;

}

}

};

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

if(argv[1]==NULL) {

cout<<"no parameter"<<endl;

return 0;

}

ofstream out1;

out1.open(argv[2]);

streambuf \*console = cout.rdbuf();

//set output to outfile 1

cout.rdbuf(out1.rdbuf());

//step 0,1

Median3X3 \*m = new Median3X3(argv[1]);

//step 2

m->mirrorFramed();

//step 3,4

m->medianFilter();

//step 5,6

m->print();

//step 7

cout.rdbuf(console);

cout<<"done"<<endl;

out1.close();

delete m;

return 0;

}

**Output**

Median3X3Out1\_PP.txt:

31 40 0 9

1 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 1 1 1 1 1 1 1 1 1 1

1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 1 2 1 2 1 1

2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 1 2 1 1 1

1 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 1 2 1 1 1

1 1 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 1 2 1 2 1 1 1

1 1 2 1 2 2 2 2 2 2 3 2 3 3 3 3 3 2 3 2 3 2 3 2 3 2 3 2 3 2 2 2 2 1 2 2 2 2 1

1 1 2 2 2 2 3 3 3 5 5 6 5 6 5 6 5 6 5 6 5 6 5 6 5 6 5 6 5 6 5 5 2 2 2 2 2 2 1 1

1 2 2 2 2 2 3 4 5 6 7 7 7 7 7 7 7 7 7 7 8 7 7 7 7 7 7 7 7 7 7 6 5 2 2 2 2 2 1 1

1 2 2 2 2 2 2 4 5 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 7 7 5 2 2 2 1 2 1 1

1 2 2 2 2 1 2 5 8 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 8 8 7 5 2 1 2 1 2 1 1

1 1 2 2 2 2 1 2 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 8 7 5 2 1 2 1 1 1

1 2 2 2 2 1 2 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 8 7 2 2 1 2 1 1 1

1 2 3 3 2 2 1 2 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 8 7 2 2 1 2 1 1 1

1 1 2 2 2 1 2 5 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 8 7 2 1 1 2 1 1

1 2 2 2 2 4 6 9 9 9 9 9 9 9 9 1 1 9 9 9 9 9 9 9 9 9 9 9 8 7 2 1 1 2 1 1

1 1 2 2 2 4 5 6 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 8 7 2 1 1 2 1 1

1 1 2 2 4 4 5 6 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 8 7 2 1 1 2 1 1

1 2 3 3 3 4 5 6 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 8 7 1 1 1 2 1 1

1 1 2 2 3 3 5 5 8 9 9 9 9 9 9 9 9 9 9 9 9 8 8 8 9 9 9 9 9 8 7 1 1 1 2 2 1

1 2 2 3 3 3 5 5 8 8 8 8 8 8 8 8 8 8 9 8 9 8 7 7 8 8 8 8 8 7 7 1 1 1 1 2 2 1 1

1 2 1 2 3 3 4 5 6 8 8 7 7 7 7 8 8 8 8 7 8 7 7 7 7 7 7 7 7 7 6 2 1 1 1 2 2 1 1

1 1 2 1 2 2 3 4 3 5 5 6 5 6 5 6 5 6 5 6 5 6 5 6 5 5 5 5 5 6 5 5 2 1 2 1 2 1 1 1

1 1 2 1 2 2 2 2 2 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 2 2 2 2 2 1 2 1 1 1

1 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 1 1

1 1 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 1 2 1 2 1 1 1

1 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 1 1 1

1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 1 1 1

1 2 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 1 1

1 1 2 2 2 2 2 2 3 2 2 2 2 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 1 1 1 1 1 1 1 1 1 1

Median3X3Out2\_PP.txt:

31 40 0 9

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1

2 2 2 2 2 2 1 2 2 2 2 2 2 2 1 2 2 2 1 1 1 1 1 1 1 1 1 1 1 2 2 2 3 3 3 2 1 1 2 2

1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 8 8 8 2 2 1 2 1 2 2 2 2 1 1 2 3 2 1 1 2 2

1 1 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 3 7 8 8 8 7 2 2 2 1 2 2 2 2 1 2 2 3 1 1 1 2 2

1 1 1 1 2 2 1 2 1 1 1 1 1 1 2 7 8 8 8 8 7 7 2 1 1 1 2 2 2 1 1 1 2 1 1 1 2 2

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 7 8 8 8 7 7 7 7 7 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 2 1 1 1 7 8 8 8 7 7 7 7 8 8 8 1 1 1 1 1 1 1 1 1 1 1 1 1 1

2 2 1 2 1 2 1 2 1 2 1 2 2 2 7 8 8 8 7 7 8 8 8 8 8 8 8 2 1 1 1 1 1 1 1 1 1 1 1 1

1 2 2 2 1 1 1 2 1 2 1 2 2 7 8 8 8 7 7 9 9 8 8 8 9 9 9 7 1 1 1 1 2 2 1 1 1 1 1

1 2 2 2 1 1 1 1 2 7 8 8 8 7 7 9 9 8 8 7 7 9 8 8 8 8 2 1 1 1 1 1 1 1 1 1

2 2 1 1 1 1 7 8 8 8 8 8 8 8 8 8 8 7 6 9 8 8 8 8 7 2 1 2 1 1 1

2 1 1 1 1 1 1 8 8 8 8 8 8 8 8 8 8 8 8 7 6 8 8 8 8 8 7 7 3 2 1 1 1 1 1

2 2 1 1 1 1 1 1 7 8 8 8 8 8 8 8 8 8 8 7 7 7 8 8 8 8 8 8 8 8 7 3 2 2 1 1 1 1

1 2 2 1 1 1 1 2 8 8 8 8 8 8 8 8 8 8 8 8 7 7 7 8 8 8 8 8 8 7 8 8 8 3 2 1 1 1 1

1 1 1 1 2 8 8 8 8 8 8 8 8 8 9 8 8 7 7 8 8 8 8 8 8 8 8 8 8 8 3 2 1 1 1 1

1 1 8 8 8 8 8 8 8 7 7 9 9 8 8 8 9 9 8 7 8 8 8 8 8 8 8 2 2 1 1 1

1 1 8 8 8 8 8 7 7 8 9 8 8 8 9 9 8 7 6 6 7 8 8 8 7 2 1 1 2 2

2 1 1 1 1 8 8 8 8 8 7 7 8 8 8 8 8 7 8 8 6 7 7 8 7 7 2 2 1

1 2 1 1 1 1 7 8 8 8 8 8 8 8 8 9 9 7 7 8 8 8 8 8 7 2 2 2 2

1 2 1 1 1 1 1 1 8 8 8 8 8 8 8 8 8 8 7 8 9 8 8 8 7 2 1 1 2 2 1 1 1 1 1

2 3 1 1 1 1 1 1 1 1 7 8 8 8 8 8 8 8 8 8 9 9 8 8 7 2 1 1 1 1 2 1 1 1 1 1 1

2 2 1 1 1 1 1 1 1 8 8 7 7 9 9 8 8 8 9 9 8 8 2 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 8 8 7 7 9 9 8 8 8 8 7 2 1 1 1 1 1 1 2 2 1 1 1 1 1

1 2 2 2 1 1 1 1 1 1 1 8 8 7 7 9 9 8 8 7 2 1 1 1 1 1 1 1 2 2 1 1 1 1 1

1 2 2 2 1 2 2 1 1 1 1 1 1 1 8 7 7 9 9 8 7 2 1 1 1 1 1 1 1 1 2 2 1 1 1 1 1

2 3 2 2 2 2 2 1 1 1 1 7 8 9 9 8 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

2 1 1 1 2 1 1 1 2 1 1 1 1 8 8 8 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 2 2 2 1 1 1 1 1 1 1 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 2 2 1 1 1 1 1 1 1 1 1 1

Median3X3Out1\_thr\_7\_PP.txt:

31 40 0 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

Median3X3Out1\_thr\_8\_PP.txt:

31 40 0 1

1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1

Median3X3Out2\_thr\_7\_PP.txt:

31 40 0 1

1 1 1

1 1 1 1 1

1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1

1 1 1 1 1

1 1 1

Median3X3Out2\_thr\_8\_PP.txt:

31 40 0 1

1 1 1

1 1 1

1 1 1 1

1 1 1

1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1

1 1 1 1 1 1

1 1 1 1

1 1 1 1

1 1 1