//Name: Mehmet Fatih Celik

//ID: 2385268

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

struct RGB\_Image{

int R[5][5];

int G[5][5];

int B[5][5];

};

struct grayScale\_Image{

int myArray[5][5];

};

struct RGB\_Image \*formRGBImage(int num){

int i,j,k;

srand(time(NULL));

struct RGB\_Image \*my\_image;

my\_image= (struct RGB\_Image\*)malloc(num\*sizeof(struct RGB\_Image));

for(k=0;k<num;k++){

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

for(j=0;j<5;j++){

my\_image[k].R[i][j] = rand() %256;

my\_image[k].G[i][j] = rand() %256;

my\_image[k].B[i][j] = rand() %256;

}

}

}

return my\_image;

}

struct grayScale\_Image \*thresholding(struct RGB\_Image \*my\_image, int thresholding\_num, int num){

int i,j,k;

struct grayScale\_Image \*grayScale;

grayScale = (struct grayScale\_Image\*)malloc(num\*sizeof(struct grayScale\_Image));

for(k=0;k<num;k++){

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

for(j=0;j<5;j++){

if (thresholding\_num > (my\_image[k].R[i][j]+ my\_image[k].G[i][j] +my\_image[k].B[i][j])/3)

grayScale[k].myArray[i][j] = 0;

else

grayScale[k].myArray[i][j] = 1;

}

}

}

return grayScale;

}

void displayThresholdedImage(struct grayScale\_Image \*grayScale, int num){

int i,j,k;

for(k=0;k<num;k++){

printf("\n\*\*\*\*\*\*Image %d\*\*\*\*\*\n",k+1);

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

for(j=0;j<5;j++){

if (j==4){

printf("%d\n",grayScale[k].myArray[i][j]);

continue;

}

printf("%d\t",grayScale[k].myArray[i][j]);

}

}

}

}

int main(){

struct RGB\_Image \*my\_image;

struct grayScale\_Image \*grayScale;

int num, thresholding\_num;

printf("Please enter how many images you want to create: ");

scanf("%d",&num);

my\_image = formRGBImage(num);

printf("Please enter the thresholding value: ");

scanf("%d",&thresholding\_num);

grayScale = thresholding(my\_image, thresholding\_num, num);

displayThresholdedImage(grayScale, num);

return 0;

}