

# Programming Project #7

## EGRE245 Fall 2017

### Darkest Points

## 1 Overview

Astronomers have come to you with a problem - they want you to write a program to find the darkest points of the sky from an image taken from one of their telescopes. After careful analysis of their images you determined that all of them are rather “noisy”, meaning that they contain a significant number of pixels that are distorted in some manner. As a result of this absolute darkness (total blackness) rarely occurs in any of their images. Despite these problems, however, you were able to come up with a simple but effective scheme to find the darkest points in these images. After finding the darkest point (the point closest to total blackness), you discovered that all points within 30% of this point (as defined below) can also be considered a significantly dark point in the image.

Write a C program that inputs a PPM image file (as defined below) of up to 500,000 RGB points, with each point defined by 3 integer values (an RGB triplet) in the range of 0-255 inclusive, where (0,0,0) equals black and (255,255,255) is white. We will define a pixel’s darkness value  $DV$  as the sum of its R, G, and B values, with absolute black therefore having  $DV = 0$ . Your program should first find the darkest point in the image, then output the coordinates (in order top-to-bottom, left-to-right, with the row first followed by the column) of all pixels that have a  $DV$  within (i.e. up to and including) 30% of this value. For example, if  $DV = 29$  for the darkest point in an image, you should report the locations of all pixels with a  $DV$  up to and including 38. When reporting the location of your dark pixels, assume the upper left corner of the image is location 1, 1. If you need to convert from a float to an integer, your program should round off using the `math.h` function `rintf`. You need to input the image file name from the command line as demonstrated in class (see the sample run below). Also you should only input the data file once. All of your output should match the sample run *exactly* (with the exception of the first line). If the input file does not exist or if it contains an illegal number of points, report an error and immediately exit the program. You may assume that each triplet is complete (i.e. if a pixel is defined at all, 3 values will be contained in the file for that pixel).

You should divide your code into subprograms (with a minimum of separate functions to input your data and report the darkest points) with any one subprogram not greater than 25-30 lines of code. You are also not allowed to use global variables (with the exception of preprocessor definitions).

## 2 PPM File Format

A PPM image file is an ASCII text file with a specified format; each file consists of the following in this order:

1. The letter P followed immediately by a “magic number” indicating the type of PPM file. We will only use files of type P3.
2. The width and height of the image
3. The maximum color value in the image; this will always be 255 in our data files
4. All the RGB pixel data follows as  $(\text{width} \times \text{height})$  number of triplets (i.e. one RGB triplet defines one pixel)

Note that all numbers are integers, and that all values are separated by white space. Pixel data is stored in the file in row-major form.

Here is a partial example:

```
P3      ← “magic number”
200 200 ← width,height of image
255     ← maximum color value
18 18 122 ← (width × height) # of RGB triplets
22 0 0      ↓
0 255 255   ↓
45 90 200   ↓
⋮
```

## 3 Sample Run

File `n44f.ppm` (shown below) is down-loadable off of the class web site along with a viewer that allows you to view PPM files in your browser. Many text editors and browsers also display PPM files as images.



Here are the first 10 lines of file `n44f.ppm`:

```
P3
600 339
255
44 5 8
44 5 8
43 4 7
42 3 6
42 3 4
44 5 6
46 6 7
```

And a sample run on file `n44f.ppm`:

```
Terminal — tcsh — 55x13
liberty:~/tmp/% gcc darkest.c
liberty:~/tmp/% a.out n44f.ppm
Darkest Part of Sky - D.Resler 3/2014
image name: n44f.ppm
magic number   = P3
width, height  = 600, 339
color val      = 255
# points       = 203400
darkest points of sky within 30.0% of darkest point:
[57,256][116,594][119,593][123,595][257,244]
liberty:~/tmp/%
```

## 4 Deliverables

Name your source code file `proj7XXXX.c` where `XXXX` is the last 4 digits of your student id number. For example, if your student id number is V12345678, your file will be named `proj75678.c`. Document your code in the manner discussed in class, and submit your project in the usual way via the web using the link off of the class web page.

**Due date: Tuesday, December 5**