

# CpSc 1010 Fall 2022

## Programming Assignment 4

### Fun With ppm Images

Due 11:59 pm **Wednesday, December 7<sup>th</sup>**

#### 1. Overview

For this assignment, you will take the lab assignment that you have been working on and modify it as well as add another manipulation to it.

Concepts covered in lecture for this assignment:

- file I/O
- command-line arguments
- multi-file program, including a header file
- makefiles
- creation of pnm images
- programmer-defined functions
- loops & conditional statements

#### 2. Notes on Collaboration

You are required to **WORK INDIVIDUALLY** on this assignment with the code being written totally by yourself. **Please do not consult anyone other than your instructor or the lab assistants on any aspect of this assignment.** Other tutors that are available may be able to help with more general C concepts that you may be struggling with, but not with questions specific to the assignment. **All of the programs will be checked with MOSS, a plagiarism detection tool, and any programs found to have a high incidence of similarity will be automatically reported to the Office of Academic Integrity.**

Review the section on “Academic Integrity” on the Course Syllabus for the policy on what happens when copying or sharing code.

For any of the programming assignments in lecture and/or lab, **it is considered *cheating* to do any of the following:**

- discuss in detail the C code in your program with another student
- show or share the C code in your program with another student
- use C code obtained from another student, or any other unauthorized source, either modified or unmodified (each student is responsible for protecting his or her files from access by others)
- use re-engineering tools
- submit work of others, from the Internet or any other source, even if attributing the work to others
- post the assignment, or a request for help, on any web sites

#### 3. Program Basics

This program will present a menu to the user:

```
Image manipulation program.

1. reduce size of image #1
2. superimpose image #2 onto image #1
3. reduce size of superimposed image
4. quit the program
```

When the user chooses 1, the program will create and save to a file a smaller version of the image (Disney.pnm, for example).

When the user chooses 2, the program will superimpose image #2 onto image #1 (ClemsonPaw.pnm superimposed onto Disney.pnm, for example).

When the user chooses 4, the program will quit.

Menu choice 3 will be worth extra credit (explained on page 4 of this write-up).

The images files to manipulate will be sent in to the executable using command-line arguments (**NOT redirection**). That means, to run this program with the Disney.pnm and ClemsonPaw.pnm images, you will type the following at the command-prompt:

```
./a.out Disney.pnm ClemsonPaw.pnm
```

### ADVICE:

**Get started ASAP!!** Procrastination leads to panic programming which leads to incomplete programs and really bad grades which then leads to a lot of heartache.

It is **HIGHLY RECOMMENDED** that you practice programming incrementally. Split this program into smaller parts, perhaps using the outline provided.

Also, **to prevent change-blindness** - - > take breaks. If you get stuck on something, walk away. Take a shower, go work out, get a snack, something. Most times, you find that after taking a break, you are thinking more clearly and can resolve the issue you were having.

**Comment your code as you go**, it's harder to add the comments after you are done with the program. Don't forget to look at the two documents that I have posted on Canvas where you found this assignment write-up concerning on **coding requirements and formatting**.

## 4. More Specifics

Your program should consist of the following files:

defs.h	containing the needed #include statements, header & pixel struct definitions, and function prototypes
makefile	containing at least 3 targets
mainDriver.c	containing the main() function and a function called showMenu()
getHeader.c	contains the getHeader() function (no change to this function from labs 10 & 11)
initInputImageArray.c	same code as fillImageArray.c from labs 10 & 11 with no change to the function called fillImageArray()
manip.c	contains two functions: reduceSize() and greenScreen() the reduceSize() function is the halfSizeWritePixels() function from labs 10 & 11 with a small modification
writeHeader.c	contains the writeHeader() function (no change to this function from labs 10 & 11)
writePixels.c	contains only the writePixels() function (no change to this function from labs 10 & 11)
ClemsonPaw.pnm	provided to you
Disney.pnm	same as with labs 10 & 11

The 2 original image files can be copied from this location:

/home/chochri/101/Assignments/F22/PA4/PA4Files

## Program outline:

### mainDriver.c file

#### main() function:

1. open input files for reading
2. get headers for images and create header for the reduced size image
3. declare arrays to hold the input images
4. fill the arrays for the images
5. present the menu to the user by calling `showMenu()`
  - a. loop for multiple menu choices
    - i. if or switch statement for menu choices
      1. menu choice 1
        - a. malloc space for a smaller image
        - b. call `reduceSize()` function sending the original image and the smaller image that was just declared; after the function is done, the smaller image will contain the array of the smaller image
        - c. declare file pointer for smaller image output
        - d. call `writeHeader()`
        - e. call `writePixels()`
      2. menu choice 2
        - a. malloc space for a new image for the green screen (superimposed image)
        - b. call `greenScreen()` function sending both of the original images and the new image that was just declared; after the function is done, the new image will contain the array of the first image with the second image super-imposed on to it
        - c. declare file pointer for smaller image output
        - d. call `writeHeader()`
        - e. call `writePixels()`
      3. explained on page 4 of this write-up
      4. quit the program
    - ii. call `showMenu()` again

#### showMenu() function:

1. present menu to user and return their menu choice back to the function from where it was called
2. force the user to enter a valid menu choice in this function, repeatedly showing the menu until a 1, 2, 3, or 4 is chosen

manip.c file

reduceSize() function:

1. same code as halfSizeWritePixels() function from labs 10 & 11 except the new pixel values will be written into a new array instead of sent to stdout
2. the new array is another parameter added to this function in addition to the original image

greenScreen() function:

1. this function will take in 3 images: both of the original images (same sized images) as well as the new image that will be created
2. the second original image will be an image where the background is green (0 255 0)
3. the new image will be a contain the pixels of the first image wherever the background of the second image is green; otherwise is will contain the pixels of the second image

## 5. Extra Credit – menu choice 3

If you get the above completed and working, and want to continue with one last addition to your program, you can earn extra credit in the form of replacing a low grade for PA1, PA2, or PA3 with the grade of this extra credit portion.

Menu choice #3 will create a reduced size of the superimposed image.

This can be accomplished one of two ways:

1. reduce size of each image and then superimpose the 2 images
2. superimpose first, then reduce the size of that image

## 6. What to Hand In

The following files: defs.h, makefile, getHeader.c, initInputImageArray.c, mainDriver.c, manip.c, writeHeader.c, and writePixels.c are to be submitted on Gradescope by midnight **Wednesday, December 7<sup>th</sup>**.

**Note:** If you choose to do the extra credit 3rd menu choice, you must first submit the version of the program *without* the extra credit implemented to the Gradescope submission not marked “extra credit”. Then, submit the extra credit version to the Gradescope submission marked “extra credit”. The autograders are different, and your submission for the base program is required.

## 7. Grading Rubric

15	File I/O	
50	Functionality	display of menu, forcing valid input creation of all the images according to the chosen menu choices
10	Multi-file program according to instructions	
5	Use of command-line arguments	
10	Makefile	
10	Code Style & Formatting	

## 8. Sample Output and Images (red items are the user's input)

**Note:** The text in red is meant to show what inputs are used for this sample run, and should not be printed by your program.

```
[20:03:13] chochri@joey15: [7] make run
./a.out Disney.pnm ClemsonPaw.pnm
```

Image Manipulation Program

1. reduce size of image #1
  2. superimpose image #2 onto image #1
  3. reduce size of superimposed image
  4. quit the program
- - > 1

Image Manipulation Program

1. reduce size of image #1
  2. superimpose image #2 onto image #1
  3. reduce size of superimposed image
  4. quit the program
- - > 1

Menu 1 option already chosen.

Image Manipulation Program

1. reduce size of image #1
  2. superimpose image #2 onto image #1
  3. reduce size of superimposed image
  4. quit the program
- - > 2

Image Manipulation Program

1. reduce size of image #1
  2. superimpose image #2 onto image #1
  3. reduce size of superimposed image
  4. quit the program
- - > 3

Image Manipulation Program

1. reduce size of image #1
  2. superimpose image #2 onto image #1
  3. reduce size of superimposed image
  4. quit the program
- - > 4

Disney.pnm (original image)



ClemsonPaw.pnm (original image)





smallerImage1.pnm (reduced-size)



greenScreen1.pnm (superimposed of the 2 original images)



smallerImage2.pnm (the extra credit image - reduced size of green-screen superimposed image)

