Assignment 6 – Color Blindness Simulator

Aiden Trager

CSE 13S – Fall 2023

Purpose

The primary goal of this program is to address the needs of individuals with color blindness, particularly focusing on the most common type—red-green color blindness, specifically deuteranopia. By simulating the visual experience of individuals with deuteranopia, the program aims to raise awareness among user-interface designers about the impact of color choices on this demographic. This initiative is crucial in fostering inclusive design practices and enhancing the overall user experience.

How to Use the Program

```
make all
```

Followed by:

```
./colorb *Flags will be needed*
```

The flags necessary can be described by the output of the -h flag, which gives:
Usage: ./colorb -i infile -o outfile
colorb -h

Program Design

colorb.c:

Program Structure Main Function (`main`):

- Handles command-line arguments for input and output files.
- Reads the input BMP file, reduces its palette using `bmp_reduce_palette` function, and writes the modified BMP to the output file.

Functionality `bmp_reduce_palette` Function:

- Accepts a BMP structure and modifies its color palette according to a specific algorithm.
- The algorithm involves calculating new RGB values based on certain conditions.

File Input/Output

- File Reading (`bmp_create`):
  - Reads BMP file header and palette information.
  - Allocates memory for BMP structure and pixel data.
  - Populates the BMP structure with data from the file.
- File Writing (`bmp_write`): Writes BMP file header, palette, and pixel data to the output file.

Memory Management `bmp_free` Function: Frees memory allocated for the BMP structure and pixel data.

io.c:

Input Functions Read Functions (`read_uint8`, `read_uint16`, `read_uint32`): Read 8, 16, and 32-bit unsigned integers from a file.

Output Functions Write Functions (`write_uint8`, `write_uint16`, `write_uint32`): Write 8, 16, and 32-bit unsigned integers to a file.

bmp.c:

Data Structures

- `Color` Struct: Represents an RGB color with three 8-bit components.
- `BMP` Struct: Represents BMP image data, including height, width, palette, and pixel array.

Helper Functions `round_up` Function: Rounds up a given value to the nearest multiple of another value.

Palette Reduction Algorithm `bmp_reduce_palette` Function: Applies a color transformation algorithm to reduce the palette of a BMP image.

iotest.c:

Testing Framework Testing Macros (`TEST`, `fatal`): Macros for writing test conditions and handling fatal errors.

File I/O Testing

- File Creation (`mkstemp`): Creates a temporary file with a random filename.
- File Reading (`read_uint8`, `read_uint16`, `read_uint32`): Tests reading functions on a file with known data.

Results

My code successfully achieves its intended purpose of processing BMP files, applying a palette reduction algorithm, and handling file input/output operations. The proof will be in a later version of the report.