Dallas Ponzo & Terron Mogensen

Data:

- Image - 2D array of characters.
- Size - Integer.
- Choice - an integer to represent the users choice in the menu.

Program:

A menu is displayed to the user which asks them if they want to load an image, display the current image, edit the current, and exit. The program will attempt to load an image from a file provided by the user. If user chooses edit, a new menu is displayed with options: crop, dim, bright, or rotate. After the user chooses one of those options they will then be prompted to save the new image to a file and if they choose yes, they will be able to choose the file name. It will then return the user to the first menu.

Functions:

main():

Data: Character array for image, int variable for size of image and int variable for user menu choice.

Functionality: Display menu to the user. Using a loop, load the image, edit it if necessary, bring the user to a secondary menu, and repeat.

displayMenu():

Data: None.

Input parameters: None.

Returned output: None.

Functionality: Print menu options on screen to the user.

loadImage():

Data: Integers I and J for array navigation.

Input parameters: 2D character array, file pointer for size.

Returned output: None.

Functionality: This function initializes the file, and two integers i and j. The user is prompted to enter the file name and the file is then opened and parsed into the 2D character array.

displayImage():

Data: Integers I and J for array navigation and a character legend array to denote the different pixel types and brightness level.

Input parameters: 2D character array, integer for size.

Returned output: None.

Functionality: This function initializes i and j and the prints a statment to the current user that says "current image:". The function then iterates through the 2D character array and prints the contents on screen.

editImage():

Data: Integer called choice for menu selection.

Input parameters: 2D character array, integer for size.

Returned output: None.

Functionality: In a loop, display a new menu to the user giving them 5 options. Crop, dim, brighten, save, or return. As long as the user chooses not to return to the main menu they will be able to edit the image continously in this loop.

cropImage():

Data: Integers for new size, row start, row end, column start, and column end.

Input parameters: 2D character array, file pointer for size.

Returned output: None.

Functionality: Function initializes 7 new integer variables to represent a new size, row start, row end, column start, and column end and integers i and j. The program will prompt the user to enter the new size, the starting bounds, and the ending bounds. If the bounds don't make sense, it will return an error. Otherwise, it starts the image at the specified locations and stores it into the 2D array.

dimImage():

Data: Integers I and J for array navigation.

Input parameters: 2D character array, integer for size.

Returned output: None.

Functionality: This function initializes integer values i and j, and does a nested for loop which checks if the pixel is already at a dimmest value, and if it is, it does not change it, but if it is not at the dimmest value, then it subtracts the brightness by 1.

brightenImage():

Data: Integers I and J for array navigation.

Input parameters: 2D character array, integer for size.

Returned output: None.

Functionality: This function initializes integer values i and j, and does a nested for loop which checks if the pixel is already at a brightest value, and if it is, it does not change it, but if it is not at the brightest value, then it adds the brightness by 1.

saveImage():

Data: Integers I and J for array navigation and a character array called filename.

Input parameters: 2D character array, integer for size.

Returned output: None.

Functionality: The function initializes a file pointer and integers i and j. It then prompts the user to enter a filename they wish to save the edited image to. After words, it opens the file, prints the contents of the 2D character array, and closes the file.

rotateImage():

Data: An integer array for temporary image storage and two integers I and J for array navigation.

Input parameters: 2D character array, integer for size.

Returned output: None.

Functionality: This program initializes a temporary array and integers i and j. It then uses a nested for loop to copy the elements of the array into the temporary array. It then iterates through the temporary array in another nested for loop that rotates them and saves the values to the normal array.