Main Algorithm:

```
1. Variables:
   - Declare and initialize a 2D array to store image data (imageData).
   - Declare and initialize a boolean variable to control program execution.
   - Declare a string variable to store the image file name (fileName).
   - Declare an integer variable to store user menu selections (menuChoice).
2. Display Menu:
   - Display the main menu options to the user:
   - Load a new image
   - Display the current image
   - Edit the current image
   - Save the edited image
   - Exit the program
 3. Prompt User for Menu Choice:
   - Prompt the user to enter their choice from the menu.
   - Read and validate the user's input as an integer.
 4. Menu Option:
   - If the user chooses to load a new image (menuChoice = 1):
   - Prompt the user to enter the file name (fileName).
   - Call the loadImage function with fileName as input to load the image data into imageData.
   - If the user chooses to display the current image (menuChoice = 2):
   - Call the displayImage function with imageData as input to display the image.
   - If the user chooses to edit the current image (menuChoice = 3):
   - Call the editImage function with imageData as input to edit the image.
   - If the user chooses to save the edited image (menuChoice = 4):
   - Call the saveImage function with the edited image data and a file name prompt.
   - If the user chooses to exit the program (menuChoice = 5):
```

Algorithm: loadImage

```
1. Open the file specified by the file name in read mode.
2. Check if the file opened successfully:
   - If not, display an error message and return.
3. Read the dimensions of the image from the file (rows and columns).
4. Allocate memory for a 2D array to store the image data based on the dimensions.
5. Read pixel data from the file and store it in the 2D array.
6. Close the file.
7. Return the image data array.
```

Algorithm: displayImage

```
1. Loop through each pixel in the image data array:
   - Determine the brightness level of the pixel.
   - Map the brightness level to a corresponding character based on the brightness-value table.
   - Display the character on the screen.
2. After displaying each row of the image, move to the next line for the next row.
```

```
Algorithm: editImage
1. Display the editing menu with options:
   - Crop
   - Dim
   - Brighten
   - Rotate 90 degrees (Extra Credit)
2. Prompt the user to choose an editing option.
3. Based on the user's choice:
   - If crop is chosen:
     - Call the cropImage function.
   - If dim is chosen:
     - Call the dimImage function.
   - If brighten is chosen:
     - Call the brightenImage function.
   - If rotate 90 degrees is chosen (Extra Credit):
     - Call the rotateImage function.
4. Return the edited image data array.

Algorithm: cropImage
1. Create a new 2D array to store the cropped image data based on the specified dimensions.
2. Copy the pixels from the specified range of rows and columns from the original image data to the new array.
3. Return the cropped image data array.

Algorithm: dimImage
1. Loop through each pixel in the image data array:
   - Decrease the brightness level of the pixel by one step.
2. Return the dimmed image data array.

Algorithm: brightenImage
1. Loop through each pixel in the image data array:
   - Increase the brightness level of the pixel by one step.
2. Return the brightened image data array.

Algorithm: rotateImage
1. Create a new 2D array to store the rotated image data based on the dimensions (switching rows and columns).
2. Loop through each pixel in the original image data array:
   - Calculate the new coordinates for the pixel in the rotated image.
   - Copy the pixel value to the corresponding position in the rotated image data array.
3. Return the rotated image data array.

Algorithm: saveImage
```

1. Open a file with the specified file name in write mode.
2. Check if the file opened successfully:
   - If not, display an error message and return.
3. Write the dimensions of the edited image (rows and columns) to the file.
4. Write the pixel data of the edited image to the file.
5. Close the file.
6. Display a message confirming the image is saved.