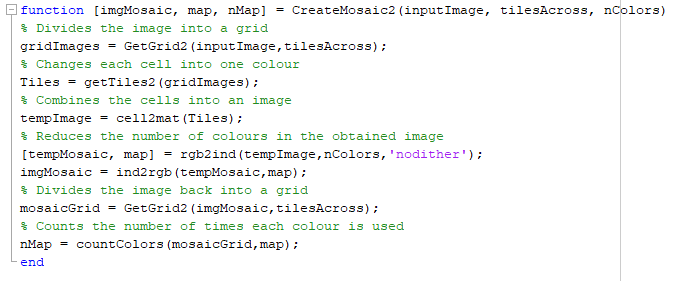
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| * Solid Colors * Graphical * GUI * Output | **Description :**  A mosaic is a piece of [art](https://en.wikipedia.org/wiki/Art) or image made from the assemblage of small pieces of colored images.  This program offers the user two methods to convert their desired image into a mosaic (using solid colors or paintings from the directory assembled for the program).   |  |  | | --- | --- | | **TEAM ( Project No. : 10 )** | **ID** | | Ibrahim Amir Ibrahim  Rowan Hisham Mohamed  Bichoi Wael Badie  Safynaz Tarek Hussein  Yasmin Hesham Mohamed | 5424  5452  5436  5454  5368 | |

# **Solid Colors**

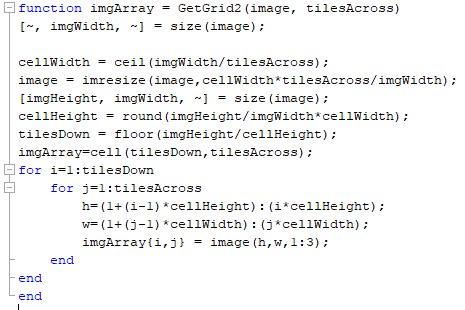
# **CreateMosaic2:**

This function takes input from the user: the image to be converted to a mosaic, the number of tiles across the image width, and the number of colors to be used in the mosaic. It outputs the final mosaic image, the color map, and the number of times each color is used.



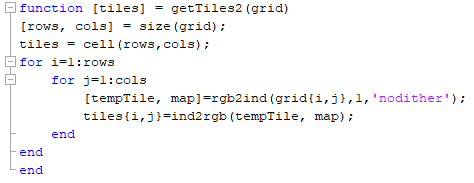
# **GetGrid2:**

This function takes the image and the number of tiles across the mosaic image, calculates the dimensions of each cell and divides the image into a grid in a cell array.



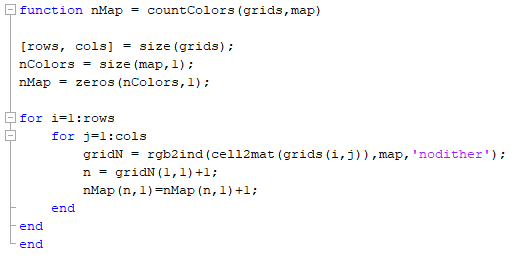
# **GetTiles2:**

Takes the grid array as input and outputs the same array but with each cell having one color.



# **countColors:**

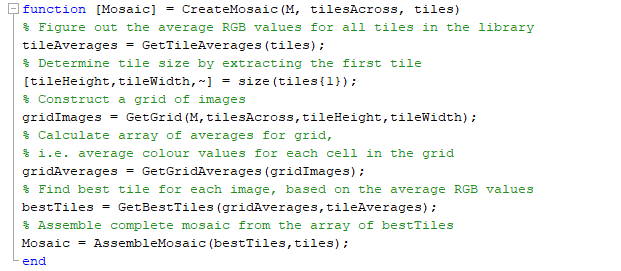
Takes the final mosaic image as a cell array and the color map and outputs the number of times each color was used.



# **Graphical**

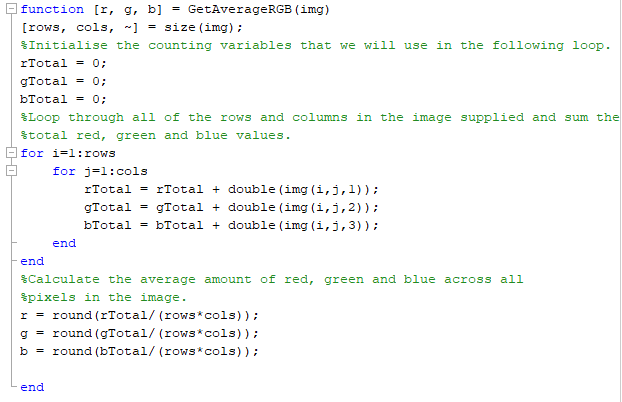
# **CreateMosaic:**

Takes input the number of tiles across the image width and the image and tiles from directory, then makes a grid over the image for each cell on the grid, calculates the average RGB values for each cell on the grid, finds the closest and matches from the list of tiles. It assembles the matched tiles together into an image.



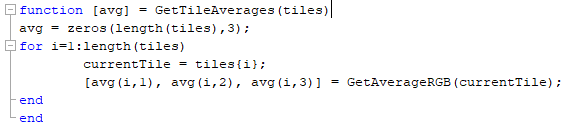
# **GetAverageRGB:**

GetAverageRGB will find the average amount of red, green and blue in an image which takes an input a 3D array containing an RGB image and outputs the average amount of red, green and blue in the image supplied.



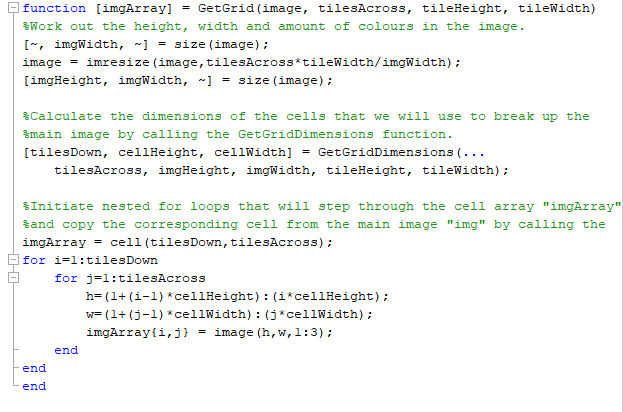
# **GetTileAverages:**

GetTileAverages finds the average amount of red, green and blue for all of the tiles contained within a 1D cell array. It takes input: tile which is a 1 dimensional cell array containing a list of tiles and outputs: avg which is a 2 dimensional numerical array where each row corresponds to a different tile and columns 1 through 3 represent the amount of red, green, and blue respectively in each tile.

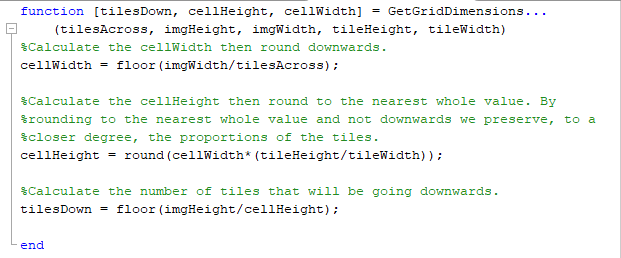


# **GetGrid:**

GetGrid splits an image up into a specified grid and take inputs: the image that will be broken down into tiles, the number of tiles that will go across the image, the height of each tile, the width of each tile and outputs the cell array of the individual tiles that have been created from the input.

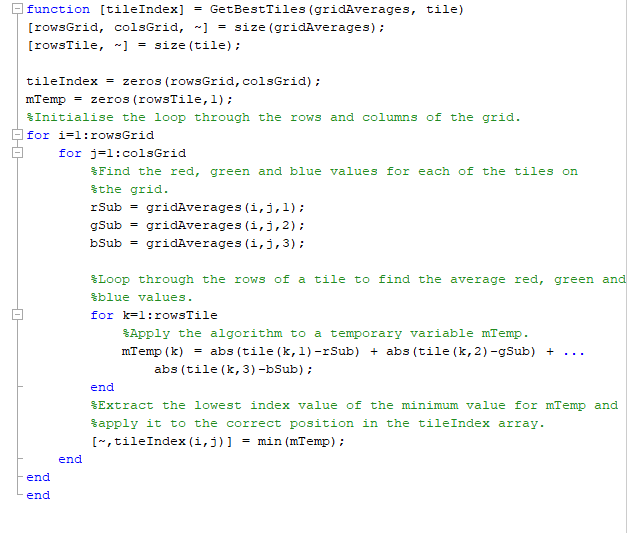


# **GetGridDimensions:**



# **GetBestTiles:**

GetBestTiles will return the best tiles to use to fill the grids that have been created over our original image which takes input: a 3D array that contains the average RGB values for each of the cells that have been created over the original image, a 2D numerical array where each row corresponds to the average red, green, and blue values for a tile and finally outputs the index number of the tile that should be used in the place of the grid cell.



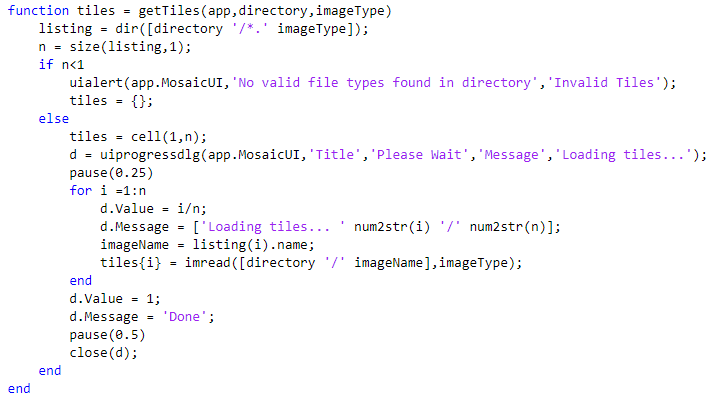
# **AssembleMosaic:**

# 

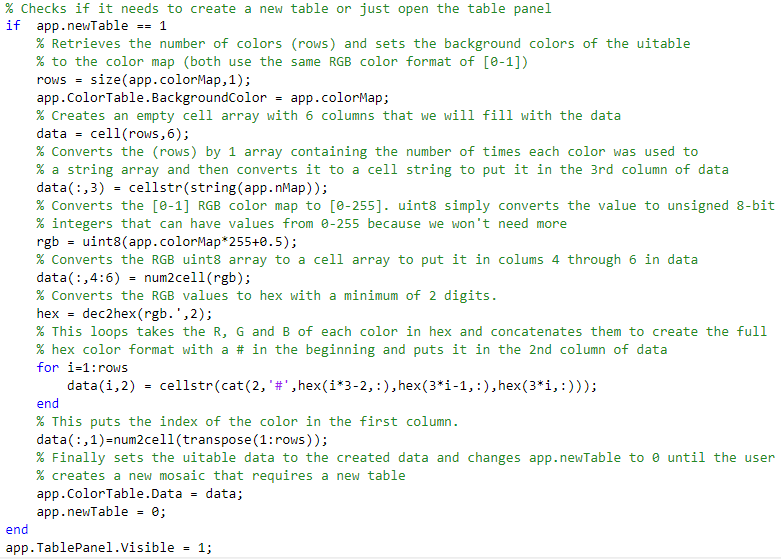
# **GUI**

# **GetTiles:**

GetTiles loads all image files from the specified directory that match the imageType file extension and returns them in a cell array structure. The valid values for image Type are : bmp, gif, jpg, jpeg, png, tif, tiff. If no tiles were found an error message is displayed and an empty cell array returned.

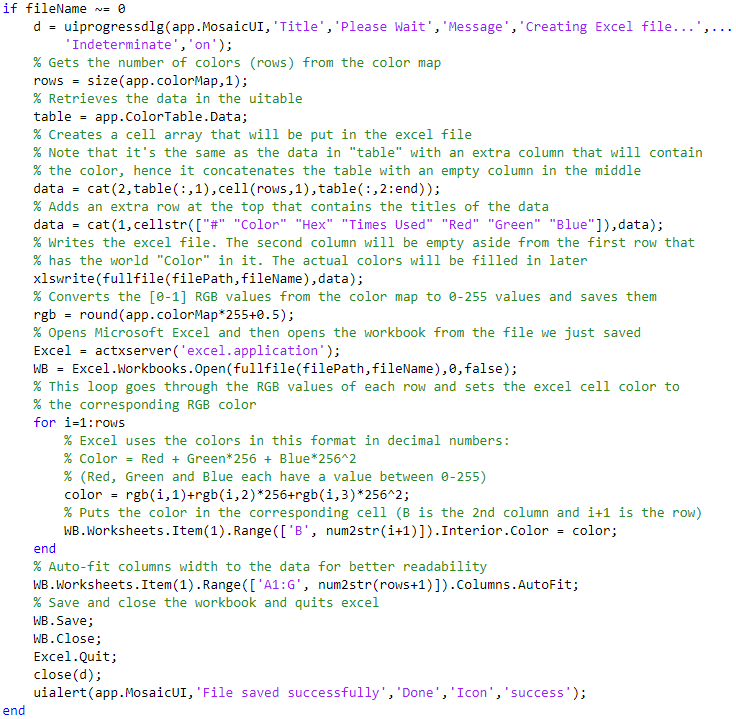


# **Table:**



# **Excel sheet**:

# 



# **Output:**





