Homework 3 Part 1: Algorithms

1. Algorithm for Populate Matrix:
   1. Randomly generate the row size using ‘M’
   2. Randomly generate the column size using ‘N’
   3. Randomly generate the matrix of size MxN using randi()
   4. Open a text file in write mode
   5. Write the dimensions of the matrix in the first line
   6. Write the first row in a new line
   7. Close the file
2. Algorithm for process\_file:
   1. Open the input text file that contains the matrix
   2. Read the row count into M
   3. Read the column count into N
   4. Read all elements into a matrix
   5. Return that matrix plus the dimensions of the matrix
   6. Close the file
3. Algorithm for calc\_code:
   1. Convert the given matrix into a linear array or vector
   2. sort the array so that the values are in ascending order
   3. set the count variable to 1
   4. Also declare mode count variable to keep track highest frequency count
   5. Also declare a mode number variable to store mode
   6. iterate through the linear array and compare size by side elements to count their frequency
   7. whenever the side by side elements are equal, increase the frequency count
   8. Whenever the side by side elements are not equal, check for the highest frequency count and if the current count is highest so far, set the current array element as mode. Also reset the count variable to 1.
   9. The final value is stored in the mode is the required mode. Thus print the mode number
4. Algorithm for calc\_avg
   1. convert the given matrix into a linear array or vector
   2. Find the sum of the elements of the array
   3. Also find the length of the array
   4. Divide the sum of the elements of the array by length of the array and store the result in a varaible ‘avg’.
   5. Print avg
5. Algorithm for find\_mmm
   1. convert the given matrix into a linear array or vector
   2. sort the array so that the values are in ascending order
   3. since the array is in ascending order, first element is the minimum. Thus, store it in a variable ‘Mi’
   4. since the array is in ascending order, last element is the maximum. Thus, store it in a variable ‘Ma’
   5. since the array is in ascending order, last element is the median. Thus, store it in a variable ‘me’
   6. Print minimum, maximum and median values
6. Algorithm for output\_results
   1. Open the file called output.txt in write mode
   2. Write the dimensions of the matrix to the file.
   3. Write the mode of the matrix to the file in a new file
   4. Write the average of the matrix to the file in a new line
   5. Write the minimum of the matrix to the file in a new line
   6. Write the median of the matrix to the file in a new line.
   7. Write the maximum of the matrix to the file in a new line
   8. Close the file