CI4810/6810 Spring 2020

Assignment: Program 3 (20 points)

Due Date: July 15, 2020 – Wednesday – 12:01pm

The whole purpose of this assignment is for you to implement a program that given a graphics image made up of lines, your program can demonstrate that it can Translate, Scale, Rotate, ... the image (using the matrix representations of the transformations). You should build a simple user-interface that receives commands from the user, ... The following functions are my suggestions – but I am sure that you are more creative than I am; therefore, use your own program design if you wish. Assume that the main purpose of writing this program is for you to build a simple graphics program that you can use to demonstrate to an individual (CEO of a company, manager of a corporation, ...) that your program can perform geometric operations. Thus, what appears below (in terms of functions, ...) are only suggestions.

Implement each of the following functions:

- **Inputlines** (datalines, num)
 - { Reads 'datalines' from an external file (name of file is provided by the user). On return `num' will contain the number of lines read from the file. }
- **ApplyTransformation** (matrix, datalines) {applies the transformation matrix to the lines that appear in "datalines"}
- **Displaypixels** (datalines, num)

{ Displays (i.e., scan-converts) 'datalines' containing `num' lines }

- Outputlines (datalines, num)
 - { Outputs 'datalines' containing `num' lines to an external file (name of file is provided by the user). }
- **BasicTranslate** (Tx , Ty)

{ Translation - `Tx' is the horizontal and `Ty' is the vertical displacements. }

• **BasicScale** (Sx, Sy)

{ Scale - `Sx' and `Sy' are the horizontal and vertical scaling factors; center of scale is at the origin of the Coordinate System. }

- BasicRotate (angle)
 - { Rotation angle of rotation is `angle' degrees (clockwise); Center of rotation is at the origin of the Coordinate System. }
- **Scale** (*Sx*, *Sy*, *Cx*, *Cy*)

{ Scale - `Sx' and `Sy' are the horizontal and vertical scaling factors; center of scale is at Cx, Cy. }

- **Rotate** (angle, Cx, Cy)
 - { Rotation angle of rotation is `angle' degrees (clockwise); Center of rotation is at Cx, Cy. }

Embed the suggested functions above (together with other functions that may be needed) into a complete program to build a simple graphics system.

Notes:

- o Build a suitable user-interface so that the functionality of your program can easily be demonstrated.
- o Your program must be well structured/engineered.
- The ONLY built-in drawing function you are permitted to use is the function that when called would activate a pixel on the display monitor.
- Use the matrix representation of the transformations.
- o Concatenation must be done during execution time.