# Name & Student ID DEEN SOOKY 20157875

# Title Robot Grid drawing

# Software Description

The software being constructed should be able to read the entire ShapeStrokeData.txt file and store it in an appropriate format. The X and Y coordinates should also be scaled so that the shape drawn is 20mm in height and should be drawn in the centre of the grid square. Then allow a user to input the file for the grid drawing instructions, if the input is DrawShpes.txt then the file should also be read. After, G code commands should be generated based off these drawing instructions to draw the shapes at specific grid positions and sent to the Arduino so the robot can carry out said instructions.

This done using 3 structs to store related data in each struct ad well as relate each struct to each other. Furthermore, in total 6 functions are used where 4 of them are described below and the other two are related to sending the commands and printing to the buffer. The program should also not take up too much space when ran as the data stored throughout the program is freed up once the COM port close to close the program,

# Project Files

DrawShapes.txt

ShapeStrokeData.txt

main.c

rs232.c

serial.c

rs232.h

serial.h

# Functions

*Function to draw out the grid.*

*Void DRAW\_GRID (char\*buffer)*

*Parameters:*

*\*buffer – points to string storing the converted data from the ShapeStrokesData file*

*Return value – returns 1*

*Loops through the string DataFile line by line and stores the number of lines in the file*

*Int lineCount (FILE \* DataFile)*

*Parameters:*

*\*DataFile - points to string storing the ShapeStrokeData file*

*Return value – returns 1 (i.e., updated Line Count)*

*Stores and reads the data from the ShapeStrokeData file*

*Void Store\_Read\_ShapeStrokeData (FILE \* DataFile, int lineCount)*

*Parameters:*

*\*DataFile – points to string storing the ShapeStrokeData file*

*lineCount – stores the number of lines in the DataFile string*

*Return value – returns 1*

*Converts the stored data from the files to G code and transfers the converted data to the Arduino*

Void Gcode\_conversion\_transferData (int Xcoordinate, Ycoordinate, char \* Shape\_Data, char\* buffer)

*Parameters:*

*Char \*Shape\_Data – points to string storing the ShapeStrokeData file within the struct*

*Char \*buffer – – points to string storing the converted data from the ShapeStrokesData file*

*Int Xcoordinate -* store the X coordinate read from the ShapeStrokeData file

*Int Ycoordinate -* store the Y coordinate read from the ShapeStrokeData file

*Return value – returns 1*

# Key Data Items

|  |  |  |
| --- | --- | --- |
| Name | Data type | Rationale |
| Grid\_Coordinates | Struct | Struct to group the variables related to and read from the ShapeStrokesData file |
| Shape\_info | Struct | Struct to relate the number of shapes (i.e. 6) to the information to construct each shape (i.e. X, Y and P). |
| Storage\_info | Struct | Struct to relate and store the file data into the variables Xcoordinate, Ycoordinate and Shape\_Data |
| Shapes\_number | Directive | Defines the number of shapes in the file ShapeStokesData as 6 makes it a constant throughout the program. |
| DataFile | Char Pointer | Stores and opens the ShapeStrokeData file in read mode thus allowing the data to be accessed via the address |
| buffer | String | String to store the contents of the ShapeStrokeData file |
| Xcoordinate | int | Variable to store the X coordinate read from the ShapeStrokeData file |
| Ycoordinate | Int | Variable to store the Y coordinate read from the ShapeStrokeData file |
| Shape\_Data | Char Pointer | Variable to access the string of data stored from the ShapeStrokeDatafile and manipulate the contents via the address |
| Shape\_quantity | array | Array to store the number of shapes separately from the shape data itself |
| PMode | int | Variable to store the pen state read from the ShapeStrokeData file |
| Shape\_name | array | Array to store the name of the shapes separately from the shape data itself |
| XYPmode | pointer | Variables to access the struct Grid\_Coordinates via the address of the XYPmode |
| Storage\_data | pointer | Variable to access the struct Storage\_info via the address of the Storage\_data |
| Draw\_shape\_Data | pointer | Stores and opens the DrawShapes file in read mode thus allowing the data to be accessed via the address |
| LineCount\_shape | int | Stores the line count in the ShapeStrokesData file so it can be used to loop through the data in the related function |
| Storage\_data | pointer | Variable to store all the data read allowing easy removal of the data once the data is no longer needed |
| instructionLineCount | Int | Stores the line count in the DrawShapes file so it can be used to loop through the data in the related function |
| XGcode | Double | Variable to store the converted x coordinate data from the read file to G code |
| YGcode | Double | Variable to store the converted y coordinate data from the read file to G code |
| gCodePMode | int | Variable to store the converted pen state data from the read file to G code |

Extend table as required

# Testing Information

|  |  |  |  |
| --- | --- | --- | --- |
| Function | Test Case | Test Data | Expected Output |
| DRAW\_GRID | Drawing the grid | G1 X90 Y0\n | Draw a straight vertical line downwards |
| LineCount | Line count is printed | DataFile | 48 line in the DataFile |
| Store\_Read\_ShapeStrokeData | Storing and reading data | DataFile and lineCount | Prints out contents of DataFile |
| Gcode\_conversion\_transferData | Converts to G code | Xcoordinate, Ycoordinate , Shape\_Data and buffer | S0\n  G1 X16 Y0\n |

Extend table as required

# Flowchart(s)

May be included as separate pdf