**A)** Phase 2 of the project was relatively easy, and I completed it without any major obstacles. However, phase 3 of the project involved significant challenges. I needed to parse each element of the string and read commands one by one, along with the characters or numbers that followed. At first, I used a switch statement to account for the number of digits/characters that would follow a command. I soon realized this was a terrible idea and changed to a while loop until the next command was reached. This solved my first problem, but I also needed a method to increment and read the letters. Understanding that my integer i represented the index of a character in a string and having to increment it each time a command was read proved difficult. I was able to implement this necessary feature after returning to the drawing board and writing out pseudo code on paper, which helped immensely.

**B)**

clear grid, set size to 20x30, set character to \*, set starting point to (1,1)  
repeatedly:  
 ask for a command string to draw  
 call performCommands function  
 display status of function  
  
plotHorizontalLine function  
 check if distance is positive or negative  
 check if mode is foreground or background  
 plot horizontal line

plotVerticalLine function  
 check if distance is positive or negative  
 check if mode is foreground or background  
 plot vertical line  
  
bool VerticalCheck  
 return true if plotting a line will end up outside vertical boundaries  
  
bool HorizontalCheck  
 return true if plotting a line will end up outside horizontal boundaries  
  
boolean plotline  
 check for direction, printable character, and foreground or background  
 return false if any checks fail  
 return false if plotting the line will end up outside boundaries  
 plot the line if all test cases do not fail, return true

performcommands function  
 set up a counter to represent the index in the string

repeat until every command is read in the input string  
 if command is H  
 return syntax error if a negative or number is not after the command  
 identify the number of digits following the command  
 return syntax error if third digit is detected  
 convert the substring of the numbers into an int for distance  
 check if it is a command error outside the boundaries  
 plot the line  
 change the column position by the distance  
 increment the counter   
  
 if command is V  
 follow same process as H except check vertically for a command error, plot the line vertically, and   
 change row position by the distance  
  
 if command is C  
 clear the grid  
 set character to \* and position to (1,1)  
  
 if command is B  
 if next character is not printable, return syntax error  
 change mode to BG and update character  
  
 if command is F  
 follow same process as B, except the mode will be changed to FG  
  
 if none of these commands are found, return a syntax error  
  
Once checking loop finishes:  
 if a command error was logged, return with a command error  
 otherwise draw the grid

**C.**

v86f - syntax error takes priority over command error  
v86f@ - command error takes place for syntactically correct input  
v2V2 - capital and lowercase  
h5v1h-3 - negative sign works for h  
v5h1v-3 – negative sign works for v  
b h3f@h5b/h-8 - foreground and background commands work (3 ‘/’ and 5 ‘@’ following on first line)  
b^h3v3ch3v3 - clear command resets grid, position, character, and mode  
h25,h-10 - unrecognized commands are returned as syntax errors  
h3f - character after f is not printable  
h3b - character after b is not printable  
h125 – more than two digits is read as a syntax error for h  
v125 – more than two digits is read as a syntax error for v