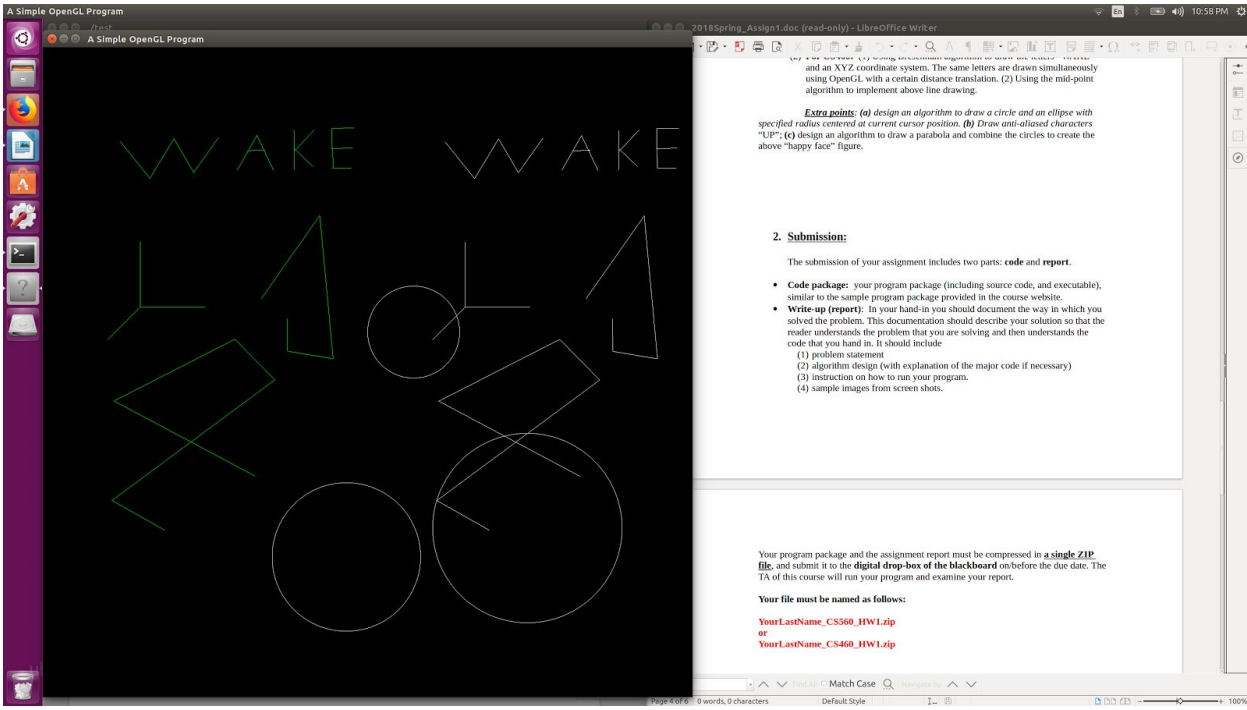
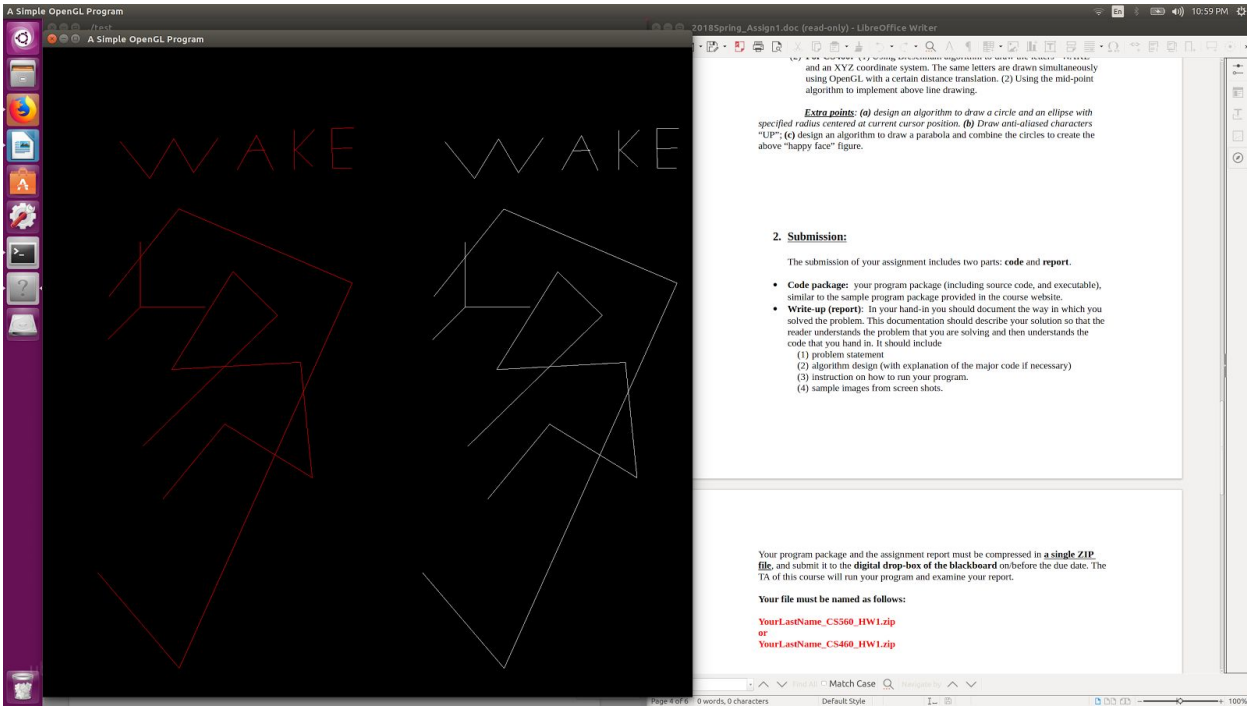
1. Problem statement
    a. The assignment was to develop a program which would draw lines using Bresenham's line algorithm and a midpoint line algorithm. The program had to be able to draw lines at any angle as well as styling the lines with a stipple. It also had to automatically draw the word wake and mirror the lines using opengl.
2. Algorithm Design
    a. Bresenham's
        i. This algorithm works by accumulating error as it draws the line. Once the error passes a certain point the program moves the pixel up and continues drawing. When the slope goes above one x is calculated with respect to y instead of y with respect to x. There is also code to deal with lines going to the left which just flips what direction a for loop goes
    b. Midpoint
        i. This algorithm works by finding the middle of the next 2 possible points. This is compared with what the math determines the point we want to draw is and is used to determine if the bottom or top point should be chosen. Reflections are used to transform to 3 other octants and there is similar code which calculates x with respect to y for octants 2, 3, 6, 7.
    c. Circle
        i. This algorithm works by drawing parts of each octant and doing transformations to find the other points.
    d. Overall program
        i. Lines and circles are saved in a global vector which gets redrawn every frame. There are also temporary structures which get updated and redrawn by the display function every time the mouse moves.
3. Instructions
    a. make test
    b. ./test
    c. 1, 2, 3, 0 keyboard keys switch stipple settings
    d. left click to create lines, right click to end a line sequence
    e. b to use bresenham's and m to use midpoint
    f. o key draws circles. hit key to start and again to end.
    g. may act strangely on non 4k monitor
4. Images

a.


A Simple OpenGL Program
WAKE
WAKE
2018Spring_Assign1.doc (read-only) - LibreOffice Writer
and an XYZ coordinate system. The same letters are drawn simultaneously using OpenGL with a certain distance translation. (2) Using the mid-point algorithm to implement above line drawing.
Extra points: (a) design an algorithm to draw a circle and an ellipse with specified radius centered at current cursor position. (b) Draw anti-aliased characters "UP"; (c) design an algorithm to draw a parabola and combine the circles to create the above "happy face" figure.
2. Submission:
The submission of your assignment includes two parts: code and report.
Code package: your program package (including source code, and executable), similar to the sample program package provided in the course website.
Write-up (report): In your hand-in you should document the way in which you solved the problem. This documentation should describe your solution so that the reader understands the problem that you are solving and then understands the code that you hand in. It should include
(1) problem statement
(2) algorithm design (with explanation of the major code if necessary)
(3) instruction on how to run your program.
(4) sample images from screen shots.
Your program package and the assignment report must be compressed in a single ZIP file, and submit it to the digital drop-box of the blackboard on/before the due date. The TA of this course will run your program and examine your report.
Your file must be named as follows:
YourLastName_CS560_HW1.zip
or
YourLastName_CS460_HW1.zip

b.


A Simple OpenGL Program
WAKE
WAKE
2018Spring_Assign1.doc (read-only) - LibreOffice Writer
and an XYZ coordinate system. The same letters are drawn simultaneously using OpenGL with a certain distance translation. (2) Using the mid-point algorithm to implement above line drawing.
Extra points: (a) design an algorithm to draw a circle and an ellipse with specified radius centered at current cursor position. (b) Draw anti-aliased characters "UP"; (c) design an algorithm to draw a parabola and combine the circles to create the above "happy face" figure.
2. Submission:
The submission of your assignment includes two parts: code and report.
Code package: your program package (including source code, and executable), similar to the sample program package provided in the course website.
Write-up (report): In your hand-in you should document the way in which you solved the problem. This documentation should describe your solution so that the reader understands the problem that you are solving and then understands the code that you hand in. It should include
(1) problem statement
(2) algorithm design (with explanation of the major code if necessary)
(3) instruction on how to run your program.
(4) sample images from screen shots.
Your program package and the assignment report must be compressed in a single ZIP file, and submit it to the digital drop-box of the blackboard on/before the due date. The TA of this course will run your program and examine your report.
Your file must be named as follows:
YourLastName_CS560_HW1.zip
or
YourLastName_CS460_HW1.zip