Computer Graphics Assignment 1

Manideep Nizam

January 2019

1 Problem Statement

Create a 2D graphical application for rendering a touch-screen calculator

1. The keys and output text box must be modeled using rectangles, as Build a 2-dimensional geometric model for the keys and text box.

2.when a key is pressed, its corresponding number or operator is outputted as text in the console.

3.The red overlay box must be a scaled down duplicate of the selected key.

4. Use keyboard keys “left” ”right” ”bottom” ”up” to translate the calculator.

5. Use keyboard keys, e.g. “r” or “R” to rotate the calculator (this is different from rotating each key when pressing a key).

6. Use keyboard keys “+” and “-” to zoom in and out, respectively.

7. Generate a new action of pressing and dragging the right mouse button for moving a key using pick- point action.

2 Approach

I use 4 classes **point**,**Rectangle**,**Calculator**, **HandleEvents**.

Point has 3 coordinates as attributes with set and get methods.

Rectangle has its point corresponding to its bottomleft corner and length and breadth as its attributes with get and set methods.

it has a function check which has a point as argument and returns true is the given point lies inside the rectangle ans false else.

It has **drawrectangle** function which draws a unshaded rectangle with a given angle theta about its midpoint.The logic hear is i first translate the mid point of the rectangle. and the rotate the axes and draw axes and draw a rectangle with coordinates as +/- its length/2 and +/breadth/2. which will be the coordinates of rectangle in translated coordinates.

In Calculator the attribute is a array of rectangle objects with length 19. with set and get methods.

In **drawCalculator** i draw the calculator . I also have a method called findpos which takes a point and returns in which rectangle the point lies in using check function of rectangle. it returns -1 is it doesn't lie i n the calculator

In **HandleEvents** function I have **Update position** function which updates global variables **xfinal** and **yfinal** using **glunUnproject** function.

and i have **handlekeys** which updates x,y,scale. and in handle mouse, if left mouse is pressed i set a global variable **leftStatus** to true. and call **updatePosition**.

if right mouse is clicked i updatePosition and set global variable to zero.and to one if it is released.

In main I continuously call **drawCalculator** followed by rotating a key by calling **drawRectangle** and if **rightstatus** is zero i find the distance of bottom edge in terms of x,y from the position of curser and store it in **tempx**,**tempy**. and if rightstatus is one i update coordinates of point in rectangle in calculator to a distance of **tempx**,**tempy** from released position.

3 Answers

1)To rotate the calculator first i rotate the coordinate system and draw calculator in rotated coordinate system.

To rotate a key , after drawing the calculator i shift the origin of to the midpoint of the key and rotate the coordinate system in the steps of 10 degrees each time and draw the rectangle every time.

The difference is for rotating calculator i rotate and draw. for rotating button i draw calculator and create new coordinate system and rotate and draw rectangle.

2)I first find the button on which it is clicked and then calculate the distance of the left bottom corner of the button and the position where i click along x and y. and then i position the left bottom corner of the clicked

button at the distance calculated from the place i release the right click.

And while finding the coordinates i use **glUnproject** which gives the untransformed coordinates and this is the main reason for pick-point action to accommodate these prior transformations