****

**INSTITUTO TECNOLÓGICO AUTÓNOMO DE MÉXICO**

**Gráficas por computadora**

Agosto - Diciembre 2018

**Proyecto Final OpenGL**

**PROFESOR**: Wilmer Pereira

INTEGRANTES

**José Francisco Altamirano Zevallos**

**Fernando Merino Benítez**

**Algoritmos para cada opción del menú:**

ESC - Salir

El algoritmo consiste en que, al presionar la tecla ESC presente en el teclado, se indique que se ha llamado a una acción que corresponde a un comando exit(0). Así se nos permite cerrar la ejecución del programa

switch (key) {

case 27:

exit(0);

break;

}

M - Mover

Para el movimiento, nuestro algoritmo primero utiliza una variable global en la que se indica si se ha presionado la tecla M. Si este es el caso, dependiendo de sobre qué figura se quiera hacer el movimiento, se cambia el valor de la variable global correspondiente.

Al ser esto, en nuestro método para dibujar la figura, se activará una condicional que nos dará las coordenadas para el movimiento. Estas coordenadas son generadas con un número al azar que utiliza una función en la que se obtiene el módulo de un número al azar con respecto del largo o ancho de nuestro plano para x e y respectivamente.

En nuestro condicional, revisamos que no se haya tocado ningún borde, es decir, que nos encontremos dentro del área de visualización. Para esto, utilizamos distintas variables globales que dependiendo su valor nos indican si ya tocamos un borde o no. Dependiendo de los bordes que se hayan tocado, se hacen los ajustes necesarios a las coordenadas, sea reduciendo para alcanzar el borde mínimo o aumentando para llegar al borde máximo.

if (\_xe == 1) {

if (rrry + j == 250) { \_botea = -1; \_boteb = 1; }

if (rrry + j == 130) { \_botea = 1; \_boteb = -1; }

if (rrx + i == 560) { \_botec = -1; \_boted = 1; }

if (rrx + i == 135) { \_botec = 1; \_boted = -1; }

if (\_botea == 1) { j = j + 0.25; }

if (\_boteb == 1 && \_botea == -1) { j = j - 0.25; }

if (\_botec == 1) { i = i + 0.25; }

if (\_botec == -1 && \_boted == 1) { i = i - 0.25; }

if (k == 0) { \_botee = 1; \_botef = -1; }

if (k == -25) { \_botef = 1; \_botee = -1; }

if (\_botee == 1) { k = k - 0.25;}

if (\_botee == -1 && \_botef == 1) { k = k + 0.25;}

}

dibujaEsfera(rrx + i, rrry + j, 0 - k);

C - Cambia de color

Para el cambio de color, utilizamos tres variables globales que van a cambiar su valor al presionar la tecla c. Se calcularán nuevos valores en el rango de 0 a 1 utilizando la función random de OpenGL, y éstos se insertarán como parámetros para cada componente de color que tendrá la figura.

case ‘c’:

if (x <= 180 && x >= 60 && y <= 180 && y >= 60) { \_ccr = random(); \_ccg = random(); \_ccb = random(); }

break;

glColor4f(\_cccr, \_cccg, \_cccb, \_tcc);

R - Rotar

En el caso de la rotación, se tienen variables globales que cambian de valor al presionar r en el teclado. Haciendo esto, se indica que se iniciará la rotación, activando un condicional en la creación de la figura. En esta, se realizarán rotaciones en los ejes antes de trasladar la figura a su posición deseada, y se irá incrementando el valor de la rotación, que es dado por la variable global previamente definida.

case ‘r’:

if (x <= 170 && x >= 70 && y <= 300 && y >= 200) { \_rc = 1; }

break;

if (\_rc != 0) {

glRotated(\_rc, 1.0, 0.0, 0.0); glRotated(\_rc, 0.0, 1.0, 0.0); glRotated(\_rc, 0.0, 0.0, 1.0); rc = \_rc + 1; }

E - Escalar

Para el escalado, se tiene una variable global que indica si se ha presionado la tecla e. Si es así, se realizan casos para los números en el intervalo de 0 a 9. Al presionar alguno de esto, se calculará un nuevo valor de tamaño para la figura deseada. Dependiendo de los factores requeridos por la función de dibujo de la figura, los valores que se calcularán. Y cada tecla numérica en el teclado tiene asociado un distinto valor que incrementará o disminuirá el tamaño de la figura.

case 'e':

\_h = 1;

break;

if (\_h != 0) {

if (\_h == 1) {

switch (key) {

case '1':

if (x <= 180 && x >= 60 && y <= 180 && y >= 60) {

\_ecx = \_ecx \* 1.1; \_ecy = \_ecy \* 1.1; \_ecz = \_ecz \* 1.1;

}

if (x <= 170 && x >= 70 && y <= 300 && y >= 200) {

\_ec = \_ec \* 1.1;

}

if (x <= 200 && x >= 40 && y <= 400 && y >= 320) { \_eex = \_eex \* 1.1; \_eey = \_eey \* 1.1; \_eez = \_eez \* 1.1;

}

break;

glutSolidSphere(\_ecx, \_ecy, \_ecz);

T - Transparencia

Para la transparencia, se tiene una variable global que indica si se ha presionado la tecla t. Si es así, se realizan casos para los números en el intervalo de 0 a 9. Al presionar alguno de esto, se calculará un nuevo valor de transparencia para la figura deseada. Este nuevo valor se insertará en la función de color utilizada en la generación de la figura, en la cual se inserta un factor de transparencia.

case 't': \_h = 2; break;

if(\_h != 0){

else {

switch (key) {

case '1':

if (x <= 180 && x >= 60 && y <= 180 && y >= 60) {

\_tc = \_tc \* 1;

}

if (x <= 170 && x >= 70 && y <= 300 && y >= 200) {

\_tcc = \_tcc \* 1;

}

if (x <= 200 && x >= 40 && y <= 400 && y >= 320) {

\_te = \_te \* 1;

} break;

glColor4f(\_cccr, \_cccg, \_cccb, \_tcc);