



Universidad Nacional Autónoma de México

Facultad de Ingeniería

División de Ingeniería Eléctrica (DIE)

Laboratorio de computación gráfica e interacción humano-
computadora

Grupo 2

Práctica No. 7 Animación por KeyFrames

Alumno: Suxo Pérez Luis Axel

Maestro: Ing. Valencia Castro Luis Sergio

Semestre 2021-2

Fecha de entrega: 26 de julio de 2021

Actividades a realizar

1. Explique las modificaciones realizadas, los problemas enfrentados y las soluciones a dichos problemas.

R= Se realizo la creación de variables faltantes como el del giro de cabeza.

```
//Keyframes (Manipulación y dibujo)
float posX = 0.0f,
      posY = 0.0f,
      posZ = 0.0f,
      rotRodIzq = 0.0f,
      rotRodDer = 0.0f,
      giroMonito = 0.0f,
      movBrazoIzq = 0.0f,
      rotCab = 0.0f,
      movBrazoDer = 0.0f;
float incX = 0.0f,
      incY = 0.0f,
      incZ = 0.0f,
      rotInc = 0.0f,
      rotRodDerInc = 0.0f,
      giroMonitoInc = 0.0f,
      movBrazoIzqInc = 0.0f,
      rotCabInc = 0.0f,
      movBrazoDerInc = 0.0f;

#define MAX_FRAMES 9
int i_max_steps = 60;
int i_curr_steps = 0;
typedef struct _frame
{
    //Variables para G
    float posX;    //
    float posY;    //
    float posZ;    //
    float rotRodIzq;
    float rotRodDer;
    float giroMonito;
    float movBrazoIzq;
    float movBrazoDer;
    float rotCab;
}FRAME;
```

Y una de las mas importantes, generar un apuntador de tipo File.

```
FILE *flujo;
```

En la función “void saveFrame(void)” realice la creación del archivo porque es donde se generan los datos que me interesa guardar, con esto solo es necesario ir a la carpeta del proyecto y buscar el archivo “frame.txt”, y ahí se encontraran los frames, y no importa si el archivo existe o no, este es generado automáticamente, también agreguen los KeyFrames de las variables faltantes.

```
KeyFrame[FrameIndex].posX = posX;
KeyFrame[FrameIndex].posY = posY;
KeyFrame[FrameIndex].posZ = posZ;

KeyFrame[FrameIndex].rotRodIzq = rotRodIzq;
KeyFrame[FrameIndex].rotRodDer = rotRodDer;
KeyFrame[FrameIndex].giroMonito = giroMonito;

KeyFrame[FrameIndex].movBrazoIzq = movBrazoIzq;
KeyFrame[FrameIndex].movBrazoDer = movBrazoDer;

KeyFrame[FrameIndex].rotCab = rotCab;
//Guardo los datos de los frames en un documento de texto
err = fopen_s(&flujo, "frame.txt", "a");
fprintf(flujo, "KeyFrame[%d].posX = %.1f; \n", FrameIndex, posX);
fprintf(flujo, "KeyFrame[%d].posY = %.1f; \n", FrameIndex, posY);
fprintf(flujo, "KeyFrame[%d].posZ = %.1f; \n", FrameIndex, posZ);

fprintf(flujo, "KeyFrame[%d].rotRodIzq = %.1f; \n", FrameIndex, rotRodIzq);
fprintf(flujo, "KeyFrame[%d].rotRodDer = %.1f; \n", FrameIndex, rotRodDer);
fprintf(flujo, "KeyFrame[%d].giroMonito = %.1f; \n", FrameIndex, giroMonito);

fprintf(flujo, "KeyFrame[%d].movBrazoIzq = %.1f; \n", FrameIndex, movBrazoIzq);
fprintf(flujo, "KeyFrame[%d].movBrazoDer = %.1f; \n", FrameIndex, movBrazoDer);
fprintf(flujo, "KeyFrame[%d].rotCab = %.1f; \n", FrameIndex, rotCab);
fclose(flujo);

FrameIndex++;
}
```

También le agregue las variables faltantes en la función “void interpolación(void)”

```
void interpolation(void)
{
    incX = (KeyFrame[playIndex + 1].posX - KeyFrame[playIndex].posX) / i_max_steps;
    incY = (KeyFrame[playIndex + 1].posY - KeyFrame[playIndex].posY) / i_max_steps;
    incZ = (KeyFrame[playIndex + 1].posZ - KeyFrame[playIndex].posZ) / i_max_steps;

    rotInc = (KeyFrame[playIndex + 1].rotRodIzq - KeyFrame[playIndex].rotRodIzq) / i_max_steps;
    rotRodDerInc = (KeyFrame[playIndex + 1].rotRodDer - KeyFrame[playIndex].rotRodDer) / i_max_steps;
    giroMonitoInc = (KeyFrame[playIndex + 1].giroMonito - KeyFrame[playIndex].giroMonito) / i_max_steps;

    movBrazoIzqInc = (KeyFrame[playIndex + 1].movBrazoIzq - KeyFrame[playIndex].movBrazoIzq) / i_max_steps;
    movBrazoDerInc = (KeyFrame[playIndex + 1].movBrazoDer - KeyFrame[playIndex].movBrazoDer) / i_max_steps;

    rotCabInc = (KeyFrame[playIndex + 1].rotCab - KeyFrame[playIndex].rotCab) / i_max_steps;
}
```

Y las agregue las variables faltantes para el dibujo de la animación.

```
//Draw animation
posX += incX;
posY += incY;
posZ += incZ;

rotRodIzq += rotInc;
rotRodDer += rotRodDerInc;
giroMonito += giroMonitoInc;
movBrazoIzq += movBrazoIzqInc;
movBrazoDer += movBrazoDerInc;
rotCab += rotCabInc;
i_curr_steps++;
```

En la función “int main()” ingrese una línea de código para borrar un archivo, el archivo que borra es el que guarda los frame, para que cada vez que se ejecute el programa se borre el archivo, y cuando se comiencen a guardar los frames, se guarden en orden sin tener información extra de ejecuciones anteriores.

```
//Borra el documento exista o no
remove("frame.txt");
```

Agregue las respectivas variables correspondientes en las rotaciones, por ejemplo, la cabeza.

```
//Cabeza
model = glm::translate(tmp, glm::vec3(0.0f, -1.0f, 0.0f));
model = glm::rotate(model, glm::radians(rotCab), glm::vec3(0.0f, 1.0f, 0.0));
model = glm::translate(model, glm::vec3(0.0f, 2.5f, 0));
staticShader.setMat4("model", model);
cabeza.Draw(staticShader);
// -----
```

También nuevos botones, para los movimientos faltantes, por ejemplo, el movimiento de cabeza.

```
if (glfwGetKey(window, GLFW_KEY_I) == GLFW_PRESS)
    rotCab--;
if (glfwGetKey(window, GLFW_KEY_O) == GLFW_PRESS)
    rotCab++;
```

El problema que tuve fue el guardado de los datos en el documento, pero la documentación de visual studio me ayudo a solucionarlo, lo que había sucedido es que no estaba usando la función correcta, usaba “fopen(“archivo.txt”, “a”)” y la función necesaria era “fopen_s(&var_tipo_File, “archivo.txt”, “a”)” con ese cambio ya podía escribir en el archivo el flujo de datos.

La sugerencia que puedo dejar es mostrarnos donde encontrar la documentación necesaria para el manejo de archivos, o la función que visual acepta, encontré muchas alternativas y no encontré algunas bibliotecas y no use esas soluciones un poco más rápidas por eso, sería bueno mostrarnos donde encontrar las bibliotecas, dado que en lo personal no lo recuerdo, supongo que se encuentra en los archivos de donde tenemos instalado “Dev-c” o “Visual studio”.



Referencias.

- <https://docs.microsoft.com/en-us/cpp/c-runtime-library/reference/fopen-s-wfopen-s?view=msvc-160>