

Tarea 2

Gráficas por Computadora

Fecha de entrega: jueves 26 de marzo, 2015

1 Temas

- Modelos jerárquicos y estructuras de datos.
- Pilas de matrices, pilas de acciones (undo/redo).
- Lectura de un modelo jerárquico desde un archivo.

2 Modelando una marioneta

Esta tarea se trata de crear una “marioneta” de manera jerárquica a partir de un número de instancias de una primitiva (esfera). Construirás y manejarás una estructura jerárquica para representar la marioneta, que será dibujada (*rendered*) utilizando OpenGL. Mantendrás también una pila de transformaciones de tipo *undo* y *redo*.

Las caras de la primitiva *esfera* deben estar formadas de rectángulos llenos (“quads” en OpenGL) para hacerla parecer sólida. Los modelos se pueden desplegar con prueba de profundidad (*depth-test*), *backface polygon culling* y *frontface polygon culling*, cada uno de ellos encendiendo o apagando el modo respectivo en OpenGL.

Para crear el modelo de la marioneta deberás instanciar y transformar apropiadamente la esfera para crear el torso (con su centro como el origen del sistema coordenado de la marioneta). Los hombros y la cadera serán instancias más pequeñas de la esfera y estarán posicionadas relativamente al sistema coordenado del torso. Sus centros formarán el origen de sistemas coordenados secundarios. El centro de la esfera del cuello y la parte superior de los brazos serán el tercer sistema coordenado subordinado al sistema coordenado de los hombros. La esfera de la cabeza se posicionará relativamente al centro del cuello y cada esfera del antebrazo posicionada relativamente al centro de la parte superior del brazo. Las manos serán esferas pequeñas posicionadas relativamente al antebrazo. Las piernas, con muslo, pantorrilla y pie se construirán de manera similar respecto al sistema coordenado de la cadera.

El esqueleto general será como el mostrado en la Figura 1.

donde la figura de la izquierda muestra el traslape de las esferas escaladas y en la figura a la derecha se han eliminado las superficies no visibles. El círculo en el centro de la cara es la nariz. Deberás tener al menos un elemento dentro de la cara para que podamos determinar si las rotaciones de la cabeza son correctas, no tiene que ser una nariz, puedes agregar ojos, orejas, cabello, boca, antenas, etc. a tu marioneta.

Puedes ser creativo con tu modelo siempre y cuando tenga al menos el mismo número de grados de libertad (número de articulaciones) que este. Puedes pensar por ejemplo en hacer un gorila, perro, oso, dinosaurio, extraterrestre, etc. pero cambiar el modelo no es requisito para esta tarea. De la misma forma puedes usar otras primitivas de modelo además de la esfera, pero tampoco es un requisito.

El sistema coordenado de cada parte del cuerpo, excepto el torso, se define relativamente al sistema coordenado de su padre. Las dependencias se muestran en la figura 2:

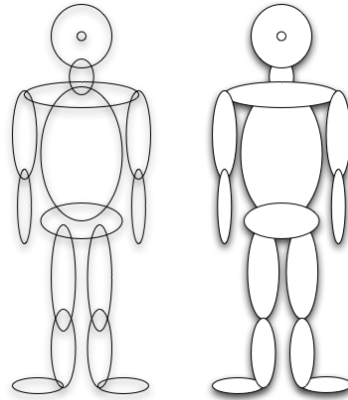


Figure 1: Modelo de la marioneta

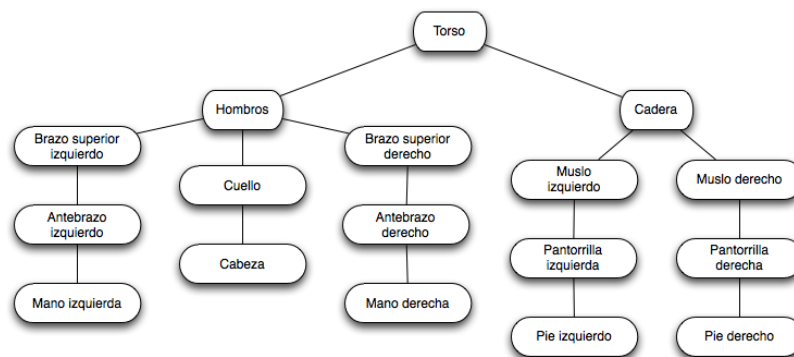


Figure 2: Modelo jerárquico de los objetos que forman la marioneta

La nariz se dejó fuera de la jerarquía puesto que es parte del modelo de la cabeza.

La definición de la marioneta deberá incluir al menos 31 ángulos libres (o cuaternios para los joints con 3 grados de libertad), 7 para cada brazo y pierna y 3 para la cabeza y cuello. Estos ángulos se usarán más tarde para animar a la marioneta. Los 7 ángulos del brazo especificarán la rotación del brazo superior en la articulación del hombro (3 gdl), la rotación del antebrazo alrededor de la articulación del codo y la rotación de la mano alrededor de la articulación de la muñeca, las tres en el mismo plano. Los tres ángulos de las piernas especificarán la rotación del muslo alrededor de la articulación a la cadera, la rotación de la pantorrilla alrededor de la articulación de la rodilla y la rotación del pie alrededor de la articulación del tobillo. De los tres ángulos restantes, uno especificará la rotación de la base del cuello hacia adelante (atrás) respecto al torso/hombros, el segundo especificará la rotación de la base de la cabeza hacia adelante (atrás) respecto al cuello y el último especifica la rotación de la cabeza a la izquierda y derecha alrededor del cuello.

Todos los ángulos deben tener un valor mínimo y un máximo que será controlado por la interfaz. Por ejemplo, no debe ser posible rotar la rodilla o el codo en el sentido “equivocado” o rotar la mano o el pie más de 90 grados de su posición neutral.

Antes de hacer tu script te recomiendo hacer tu modelo en papel.

3 OpenGL

Lo necesario para implementar esta tarea con OpenGL se puede encontrar tanto en el libro rojo como en las páginas de ayuda de OpenGL. Los siguientes puntos, aunque no son requerimientos de la tarea pueden ser útiles en la implementación:

- Normales unitarias. Para obtener buenos cálculos de iluminación debes usar normales unitarias. A pesar que es trivial especificar normales unitarias para las caras de la esfera, cuando se hacen escalamientos las normales también serán escaladas quedando de largo no-unitario. Para evitar este problema se utiliza los comandos apropiados dentro de `glEnable` y `glDisable` para que OpenGL reescale automáticamente las normales.
- A pesar de que puedes usar el comando `gluSphere` también puedes querer generar la esfera por ti mismo usando senos y cosenos, y dibujando la esfera como un conjunto de rectángulos. Para mejorar el desempeño dibuja tu marioneta como una secuencia de *quad strips* (esto no es obligatorio). Debes seleccionar una resolución para la esfera que resulte en una buena calidad de tal manera que se vea suave.

Objetivos:

Fecha de entrega: jueves 26 de marzo, 11h59 pm

Nombre: _____

- ___ 1: Las proporciones de la marioneta son razonables y las esferas están unidas de manera lógica. Las esferas están "atadas" a sus vecinas por las orillas, estas no rotan alrededor de su centro.
- ___ 2: Se utiliza una estructura jerárquica bien diseñada. Cada esfera se dibuja como instancia de una sola lista de despliegue.
- ___ 3: Están implementadas las funciones de profundidad (Z-buffer) así como la eliminación de caras al frente y atrás. Estas funciones pueden activarse y desactivarse.
- ___ 4: La marioneta está iluminada de tal manera que su estructura 3D es claramente visible.
- ___ 5: Se mantiene correctamente una pila `undo/redo`.

Recuerda que si no pudiste hacer funcionar alguno de los objetivos anteriores puedes escribir en el archivo README por qué no funciona y tus ideas para solucionarlo. Si algún objetivo no está hecho y no hay explicación sobre este no será tomado en cuenta en las tareas.

Recuerda también que el código y la documentación que entregues es personal y deberás citar tus fuentes.