

Solución Propuesta

En primer lugar, fue necesario poder leer un archivo csv, para ello se utilizó la librería csv. Se crearon varios inputs de archivos, con distintas cantidades de números. Luego, fue necesaria la definición de reconocimiento del mouse y del teclado, para ello se realizó una programación similar a ex.mouse del repositorio del curso.

Como el objetivo principal del programa era dibujar los nodos, era muy importante definir bien la modelación jerárquica. A partir de su desarrollo se crearon dos secciones principales, los círculos y los números. Para poder trabajar con los números se utilizó como guía a ex.text.render. El esquema de la modelación se encuentra en la imagen (1). Para poder juntar el dibujo de las distintas gpuShapes con distintos pipelines, se modificó el archivo de scenegraph, de tal forma que en adelante solo se dibujase a través de él. Para ello se le agregó un valor a la clase gpuShape, el cual es un int que determina su dibujo.

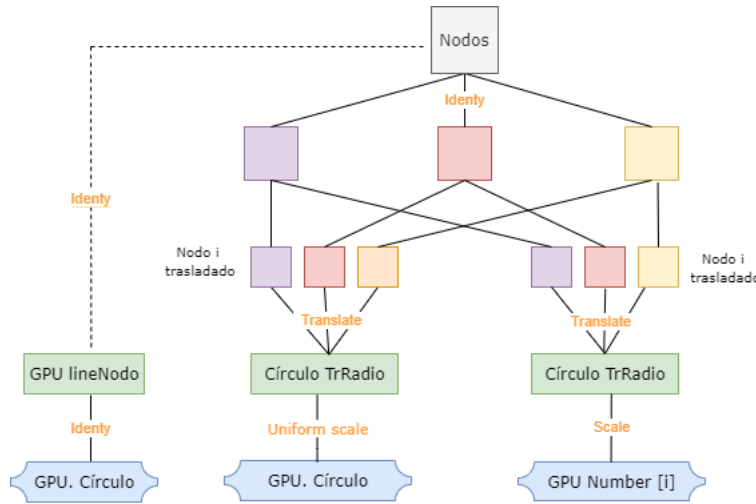


Figura 1: Grafo de Escena

Entre los pasos esenciales previos a la interacción, se re-escalo a través del grafo de escena los distintos Nodos según la cantidad enviada. Además, se definieron distintas figuras y grafos de escena para reemplazar. Para poder determinar si el mouse se encuentra dentro de un Nodo, se utilizó la ecuación de un círculo mostrada a continuación en la ecuación (1):

$$(x - x_0)^2 + (y - y_0)^2 = Radio^2 \quad (1)$$

Luego, la parte fundamental del código corresponde a las distintas salidas del ciclo for que reconoce los nodos. La estructura está determinada por distintos valores booleanos que van permitiendo entrar a distintos if del ciclo. Estos van generando las consecuencias de armado de árbol. Para detectar si se aplicó un click de mouse sobre el lado izquierdo o derecho para

formar el hijo, se utilizó la misma ecuación marcada en 1, pero además considerar que x deberá ser menor a X_0 y mayor a X_0 respectivamente.

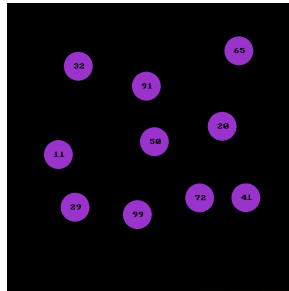
El sistema mencionado logró armar un árbol binario, pero presenta limitaciones. Como posibles "bugs" se encuentra la formación de árboles con hijos en lugares no correspondientes a un árbol binario cualquiera, pero si cumple la condición base: Cada Nodo posee como hijo derecho un nodo con un valor mayor y como hijo izquierdo un nodo con un valor menor". Por términos de tiempo, el programa no es capaz de realizar la animación de encontrar el valor pedido, pero el código se quedó a pasos de lograrlo. Además, para lograr lo planteado se utilizó una detención de tiempo entre varios nodos, lo cual resultó una forma inteligente de lograr obtener un solo click.

Instrucciones de Ejecución

El programa se ejecuta como es pedido en el enunciado. Luego, fue agregado que al apretar la tecla h de "hint", el programa reconoce el nodo que se encuentra en la posición media, de esta forma el usuario puede decidir si dejar este nodo en esa posición. Para mover los nodos el usuario simplemente debe mantener el click izquierdo y soltarlo cuando estime conveniente, dejando el nodo en la posición del mouse en ese momento. Luego para juntar nodos, tras el usuario apretar el click derecho sobre algún nodo, este se volverá verde, señalando que está seleccionado. En este punto pueden moverse los nodos o despintar el nodo seleccionado con otro click derecho. En caso de apretar otra vez el click derecho, pero sobre algún nodo distinto, este pasará a estar seleccionado como el padre del nodo verde. Uno debe seleccionar si el nodo verde será un hijo izquierdo (menor) o derecho (mayor) simplemente con apretar un click izquierdo en el sector del nodo texturizado que se desee. Tras este movimiento los nodos se juntarán con una línea, y ambos nodos volverán a estar morados. Es importante mencionar que los nodos ya no podrán moverse. Dentro de las instrucciones de formar el árbol, en caso de que se desee poner como hijo izquierdo a un nodo de mayor prioridad, o un hijo derecho a un nodo de menor prioridad, saldrá un error por 1 segundo, luego puede volver a conectarse los nodos.

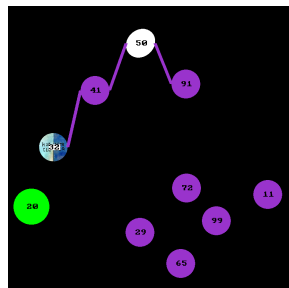
Resultados

Se observa en un inicio como funciona la aplicación de la modelación jerárquica y se distribuyen los nodos en un inicio en una posición al azar. Todo esto en la figura 2. Se representa en la figura 3 un árbol armado de forma completa. Se representa en la figura 4 Todas las estructuras creadas, los colores y las texturas de los nodos e incluso el efecto tras el hint.



Year	Number of people (millions)
1990	11
1995	20
2000	32
2005	41
2008	50
2010	45
2012	31
2015	27
2020	29

Se representa en la figura 4 Todas las estructuras creadas, los colores y las texturas de los nodos e incluso el efecto tras el hint.



3

Autoevaluación

Criterio-Puntaje	0	1	2	3
OpenGL				x
Shaders		x		
Modelos geométricos		x		
Transformaciones				x
Texturas		x		
Modelación jerárquica				x
Curvas	x			
Funcionalidades mecánicas o lógica de juego		x		
Entradas o Control de usuario		x		
Visualización de estado del programa			x	