Documento de Diseño Para Proyecto Laboratorio de Marcha

Juan Pablo Peñaloza Botero y José Rafael Domínguez Nolasco

Introducción a la Computación Gráfica

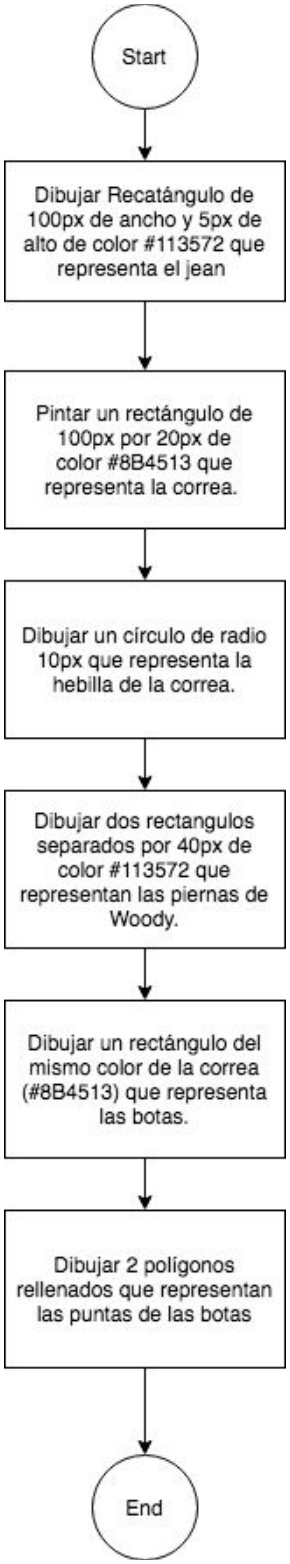
Análisis

Objetos Conocidos	● La silueta del personaje a dibujar. ● Algoritmos para dibujar: ○ Lineas ○ Elipses ○ Círculos ● Dimensiones del avatar.
Objetos Desconocidos	● Coordenadas exactas de los detalles del personaje. ● Tamaño en píxeles de las extremidades. ● Colores del personaje en formato openGl.
Condiciones	● El personaje debe tener piernas y articulaciones definidas para que más adelante se pueda animar. ● Debe caber un la ventana. ● Utilizar los algoritmos vistos en clase.

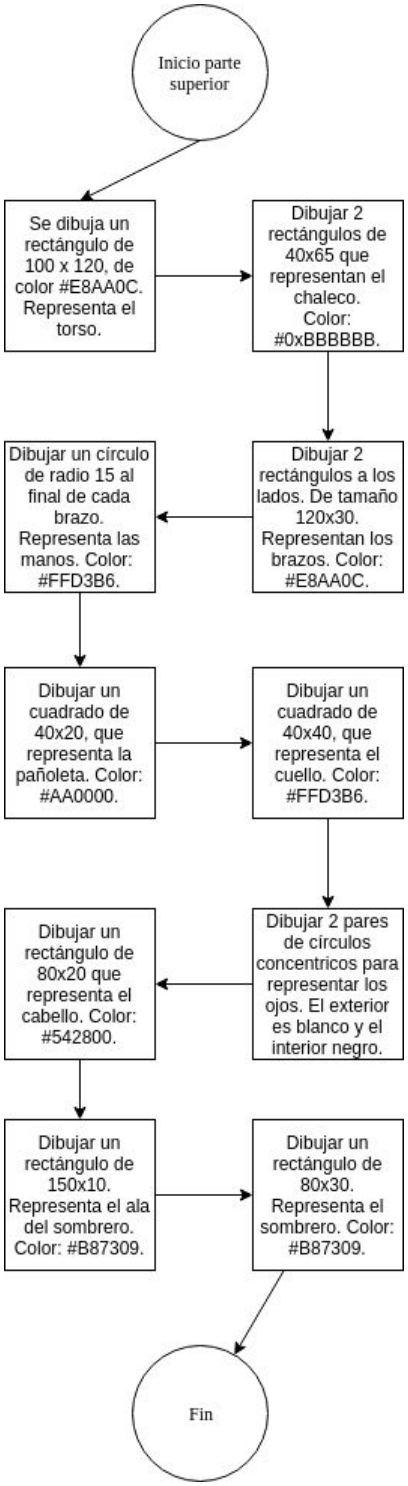
Diseño

Entradas	● Imagen del personaje a dibujar. ● Dimensiones del personaje. ● Dimensiones de la pantalla.
Salidas	● Coordenadas exactas de los detalles del personaje. ● Dibujo del personaje en OpenGL.
Condiciones	● Se debe utilizar los métodos vistos en clase de openGl ● El personaje debe tener piernas para posteriormente animar. ● Utilizar los algoritmos de elipse, círculo y línea vistos en clase.

Dibujar Cadera, Piernas y Pies



Dibujar Cabeza, Brazos y Torso



Problemas Encontrados

Problema: Cuando se fue a compilar la estructura básica del programa en el sistema operativo MacOS, cambiaban un poco el nombre de las librerías para Linux y para Mac.
Solución: Se agregó un `#if define` para que importe las librerías para Mac en caso de que se compile en Mac o de Linux en caso de que se compile en Linux.

Problema: Los colores en la mayoría de escenarios están descritos en RGB donde cada canal es un número de 0 a 255, en openGl también son los mismos tres canales pero de 0.0 a 1.0, por lo tanto al momento de buscar un color no se encontraba en términos de 0 a 1.0.
Solución: Se divide cada canal por 255 para sacar el porcentaje, pero para hacer las cosas más automáticas encontramos una página que convierte cualquier formato de color a formato de openGl.

Problema: Es común encontrar colores en notación hexadecimal, sin embargo, OpenGL no cuenta con funciones dedicadas para utilizar esta notación.
Solución: Se creó una función que recibe los valores hexadecimales de los colores como un número entero, y procede a hacer la conversión para poder ser utilizado con OpenGL.