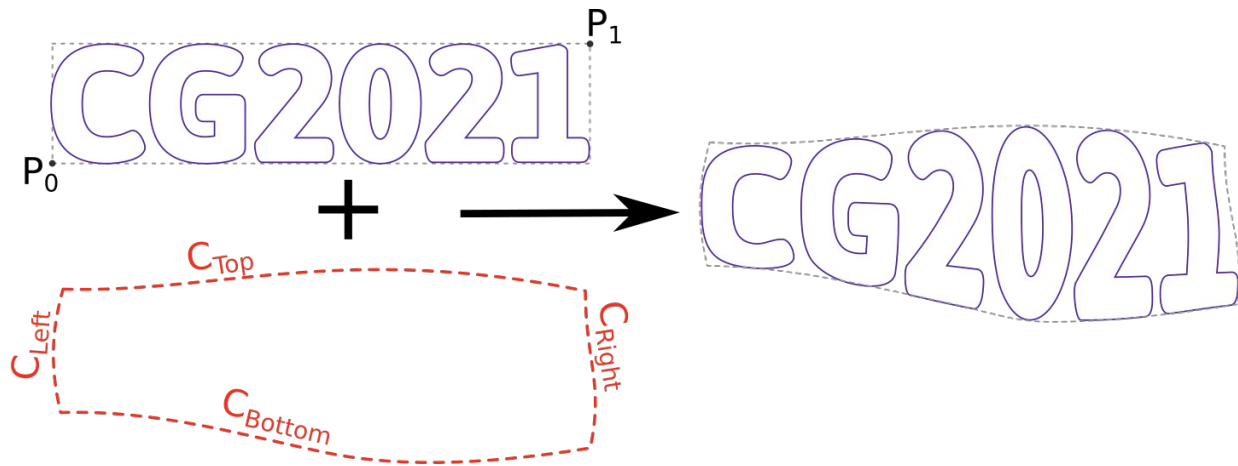


Computación Gráfica 2021 – Recuperatorio – 17/11/2021 – Ejercicio 1

Se tiene un dibujo 2D definido por un conjunto de curvas de Bezier (en el ejemplo, la palabra CG2021). Y se quiere deformar el mismo para que se “ajuste” al nuevo envoltorio (la líneas rojas).



Describe un algoritmo para rasterizar las curvas deformadas a partir de las originales. Los datos de entrada serán:

- Las curvas originales que definen el dibujo (las azules en el ejemplo)
- Los dos puntos extremos de su bounding-box (P_0 y P_1 en el ejemplo)
- Las 4 curvas que definen el nuevo envoltorio (C_{xxx} , en rojo en el ejemplo)

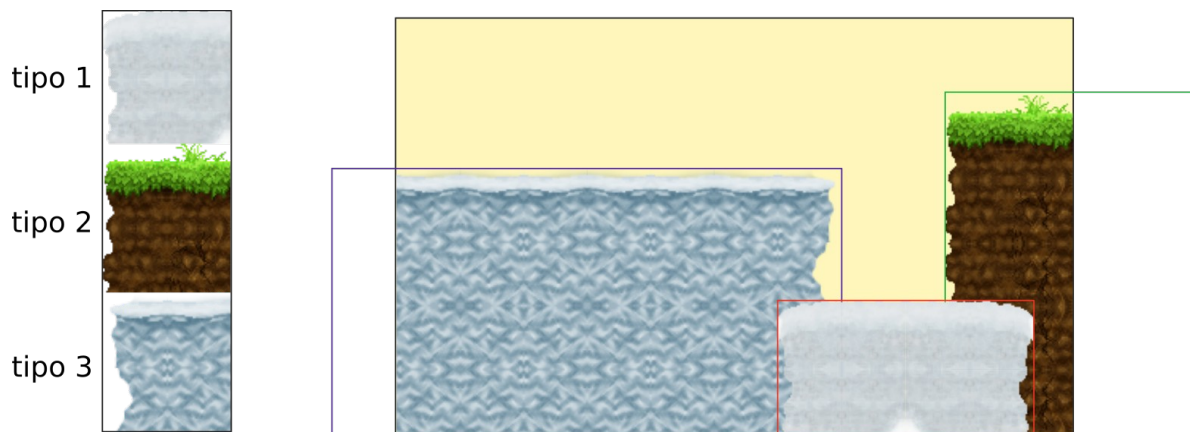
Sugerencia: Primero indique cómo calcularía la nueva posición de un punto cualquiera; y luego piense cómo podría usar ese cálculo para modificar algún método de rasterización de curvas.

Computación Gráfica 2021 – Recuperatorio – 17/11/2021 – Ejercicio 2

En un juego pixel-art (donde los gráficos se ven pixelados adrede para lograr una estética retro), se necesita una función para graficar las plataformas. La función será invocada una vez por cada plataforma del mapa, y recibirá los siguientes argumentos:

- Posición de la plataforma: un punto x,y , que indica dónde debe ubicarse el límite inferior izquierdo de la plataforma en el mapa.
- Ancho y alto de la plataforma.
- Un entero (1, 2 o 3) indicando el tipo de plataforma a dibujar.

La función utilizará una textura como la de la izquierda, donde se encuentran 3 plataformas, una de cada tipo. Esa textura debe utilizarse para dibujar plataformas de cualquier ancho, repitiendo parte de la misma tantas veces como sea necesario para lograrlo, como se observa en la figura de la derecha.



En este ejemplo, se harían 3 llamadas a la función, una para cada plataforma (las recuadradas en rojo, verde y azul).

¿Cómo implementaría esta función? ¿Qué información debería enviar a la GPU? (¿vértices? ¿coordenadas de texturas? ¿normales? ¿materiales? ¿configuraciones como el modo de aplicación, los mecanismos de filtrado, etc?)

Nota: la textura es de 80x240, y la función siempre tendrá que dibujar plataformas con tamaños múltiplos de 80.

Computación Gráfica 2021 – Recuperatorio – 17/11/2021 – Ejercicio 3

Se tiene una escena compleja formada por millones de primitivas. Para acelerar algunos cálculos, las primitivas han sido agrupadas en una Bounding Volume Hierarchy (un árbol de envolventes, potencialmente solapadas) donde la envolvente en cada nodo es simplemente un Axis-Aligned-Bounding-Box.

La posición de la cámara y la proyección perspectiva están definidas a través de los siguientes parámetros:

- Un punto **C** define la posición de la cámara y un punto **T** define el objetivo hacia el cual apunta.
- Un vector **R** define la orientación de la cámara: en la imagen final cualquier vector en la dirección de **R** debería verse proyectado como un vector horizontal apuntando hacia la derecha.
- Un ángulo θ define el ángulo de apertura de la cámara en sentido horizontal (a lo ancho).
- Dos enteros **W** y **H** definen el ancho y el alto de la ventana donde se renderizará.

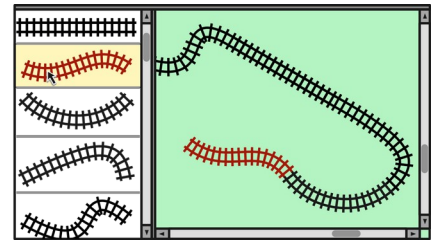
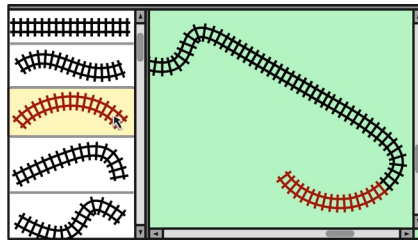
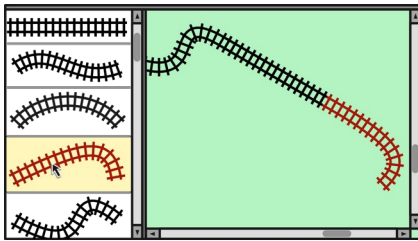
Dados estos datos (**C**, **T**, **R**, θ , **W**, **H**, y el árbol BVH donde cada nodo tiene un AABB) escribir un algoritmo que permita determinar cuáles nodos del árbol podrían contener primitivas que se vean en la imagen final, y cuales pueden descartarse por completo. Incluir todas las ecuaciones necesarias.

Computación Gráfica 2021 – Recuperatorio – 17/11/2021 – Ejercicio 4

El reverendo Alegría ha contratado a Martin Prince para que programe un software que le ayude a diseñar sus maquetas de trenes en miniatura.



Martin ya ha cargado en el software los diferentes tramos de vías de los que dispone Alegría, cada uno modelado con una curva de Bezier. El software le permite al usuario ubicar donde quiera y con la orientación que quiera (dentro de un plano XY, ya que el diseño es 2D) al primer tramo de vía. Pero luego, a medida que el usuario va seleccionando otro tramos, el programa debe ubicarlos automáticamente uno a continuación del otro, rotándolo y trasladándolo como corresponda. Sin embargo, dado que Martin solo está en 4to grado, a pesar de su alto IQ requiere de su ayuda para calcular la matriz de transformación necesaria.



¿Cómo armar la nueva matriz, sabiendo los puntos de control originales del último tramo colocado en el diseño y su matriz de transformación, y los puntos de control del nuevo tramo sobre el que el usuario hace click? ¿Qué tipo de transformación es?