



Programación 2D

Práctica 5: Renderizado de texto

En esta práctica, vamos a añadir a nuestro motor soporte para el renderizado de fuentes TrueType por medio de la clase **Font**. Utilizando dicha clase, vamos a hacer una demo donde el mensaje "**Hello, world!**" se repite en pantalla moviéndose de derecha a izquierda.

Ese mensaje se mostrará utilizando distintas tipografías. Se pueden obtener en varias webs, por ejemplo <https://www.dafont.com>. Podemos descargar varias que nos gusten y cargarlas en el programa (cada una a un tamaño diferente).

Clase Font

Vamos a crear esta clase, que permitirá cargar fuentes TrueType, obtener el tamaño que ocupa en pantalla un determinado texto con dicha fuente, obtener el alto con el que se cargó la fuente, y dibujar un texto en las coordenadas de pantalla especificadas. La clase puede tener los siguientes métodos (además de las variables miembro que consideremos necesarias):

```
static Font* load      (const char* filename, float height);  
float        getHeight () const;  
Vec2         getTextSize (const char* text) const;  
void         draw      (const char* text, const Vec2& pos) const;
```

Si se ha podido cargar la fuente TrueType indicada, el método `load` debe devolver un puntero a un nuevo objeto `Font` con los datos de la fuente. En caso contrario, devolverá `nullptr`.

Como los objetos de esta clase los vamos a crear por medio del método estático `load`, debemos hacer un constructor protegido o privado. Si es necesario, implementaremos también un destructor para liberar los recursos al destruir el objeto.

Bucle principal

Calcularemos un valor al azar entre 0 y 100. Si dicho valor es igual a 0, creamos un nuevo texto en pantalla con las siguientes propiedades:

- Una fuente escogida al azar.
- Una posición Y escogida al azar entre 0 y el alto de la ventana.
- Una velocidad al azar de desplazamiento entre 20 y 200.
- Un color al azar.

Además, siempre se debe comenzar a pintar el texto en la parte derecha de la pantalla, y a cada frame se irá desplazando a la izquierda a la velocidad indicada. Cuando el texto se salga por la parte izquierda de la pantalla, debemos eliminarlo.



Pasos a seguir en el método Load:

1. Abrir fichero `fopen_ (fontfile...`
2. Saber tamaño del mismo
 - a. `fseek (fontfile, 0, SEEK_END)`
 - b. `filesize = ftell(fontfile...`
 - c. `fseek (fontfile, 0, SEEK_SET)`
3. Crear buffer
 - a. `Int fontbuffersize = sizeof(char)* filesize`
 - b. `Char * = ...`
4. Leemos el fichero con `fread_s`
5. Bakefont para sacar el alfa
6. Rellenamos nuestro buffer alfa con lo que nos ha devuelto el `bake`
7. Creamos la textura
8. Set pixels
9. Liberamos todo

Pasos a seguir en el método Draw

1. Recorremos con un bucle el texto que se recibe
2. Para cada carácter obtenemos la información de nuestra fuente con `GetBakequad`
3. Dibujamos con `Drawrotsized` y toda la información que nos da.