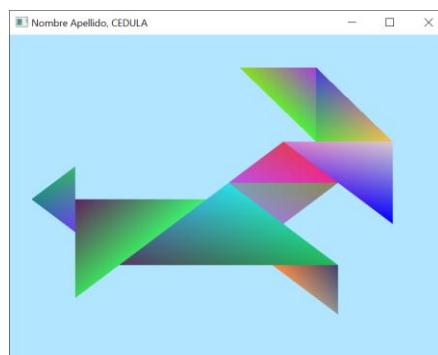


## COMPUTACIÓN GRÁFICA

### TAREA B2T1

#### INDICACIONES:

- Se evaluará tanto el **modelo obtenido** como la **calidad del software**. Se tomará en cuenta la coherencia del código y la optimización de recursos: menor número de VAOs, VBOs, EBOs, shaders y transacciones entre CPU/GPU. Recuerde utilizar shaders (GPU) para optimizar los cálculos de posición y color.
- El nombre de la ventana de su aplicación debe incluir su nombre, apellido y cédula.
- Utilizar el modelo asignado en la Práctica anterior (**Vertex Colors, Fragment Interpolation y Shader Class**). El uso de un modelo diferente al asignado invalidará toda la Tarea. Ejemplo:



## TAREAS

### TAREA 1 (4P)

- Escoger dos archivos de imagen de su autoría (*texture1* y *texture2*) y utilizarlo como archivos de textura.

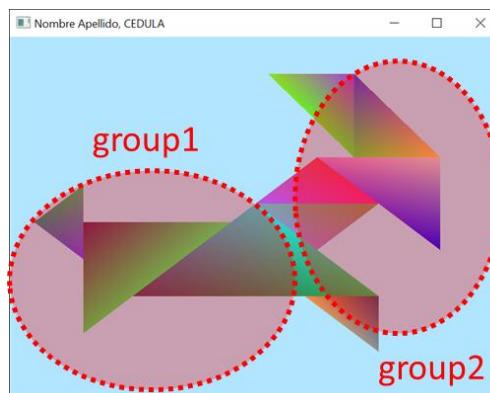
**texture1.jpg**



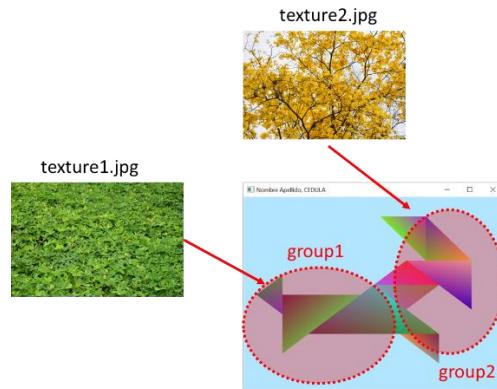
**texture2.jpg**



- Dividir el número de triángulos que conforman su modelo en 2 grupos.



Utilizar los grupos de triángulos y rellenarlos con las texturas utilizando **texture coordinates**. Recuerde que los archivos de textura deben ser cargados una sola vez en OpenGL y no pueden ser modificados/recortados/alterados.

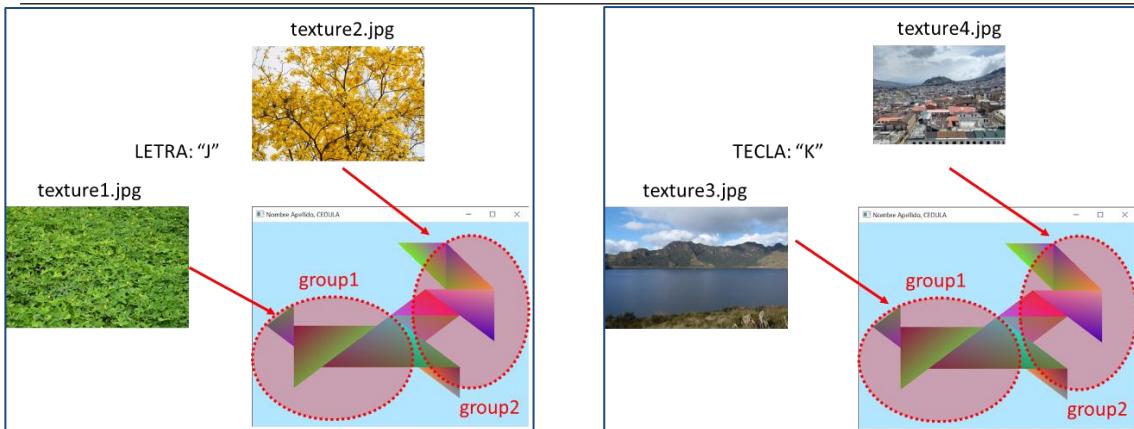


Las texturas deben llenar los triángulos del grupo sin presentar distorsiones en la escena. Un ejemplo del resultado de la Tarea se puede observar en la siguiente figura:



### TAREA 2 (3P)

Escoger 2 texturas adicionales (*texture3* y *texture4*), modificar su código para que al presionar entradas del teclado (ejemplo "J" y "K") se intercambien las texturas en la escena (*texture1*→*texture3* y *texture2*→*texture4*), tal como se explica en la siguiente figura.



### TAREA 3 (3P)

Modificar su código para que el objeto se mueva a lo largo del tiempo de arriba hacia abajo (el objeto puede salir de la zona visible de la ventana de la aplicación).

#### ENTREGA:

Utilizar la siguiente estructura y nombres de archivo para la entrega de su trabajo.

- Archivos imágenes texturas (**texture1, texture2, texture3, texture4**)
- Última versión del código fuente de su aplicación OpenGL (**Tarea\_B2T1.cpp**).
- Última versión del código fuente shaders (**vertexshader.vs, fragmentshader.fs**).
- Video demostrativo de una breve explicación, compilación y ejecución de la última versión de su aplicación (**Tarea\_B2T1.mp4**).

#### NOTA:

- La práctica es **individual**.
- La entrega se realiza únicamente por el aula virtual. No se aceptarán entregas fuera del plazo establecido o por correo electrónico.
- En caso de que el estudiante utilice un modelo diferente al asignado, no se adjunten todos los archivos, el código presente errores de compilación o el video demo no incluya la ventana de ejecución, **la práctica no será calificada**.
- En caso de detectar copia del código con trabajos de sus compañeros (o anteriores semestres), el estudiante perderá la asignatura y el evento será notificado a las autoridades académicas.