

# Laborator 03

## Cuprins

1. Obiective.....	2
2. Primitive <i>OpenTK</i> .....	2
2.1 Introducere.....	2
2.2 Primitive OpenTK.....	2
2.3 Utilizarea culorilor în OpenGL .....	2
3. Probleme de rezolvat .....	3

## 1. Obiective



- Lucrul cu API-ul OpenGL
- Lucrul cu biblioteci externe (OpenTK)
- Asimilarea elementelor de bază pentru lucrul în medii grafice 3D
- Generarea primitivelor 2D fundamentale (punct, linie, triunghi)
- Lucrul cu culori în OpenGL

## 2. Primitive *OpenTK*

### 2.1 Introducere

Framework-ul OpenTK vă expune capacitățile de anti-aliasing ale bibliotecii OpenGL. Identificați în programul-exemplu pus la dispoziție care sunt elementele care permit activarea și utilizarea acestui mod de lucru.

Pentru toate operațiunile implementate la nivelul programului, logați activitățile atât la consolă cât și într-un fișier text. Creați o metodă dedicată care va realiza acest lucru. Fișierul text de log va fi resetat la pornirea programului.

### 2.2 Primitive OpenTK

Urmăriți comportamentul aplicației la desenarea următoarelor elemente:

- puncte;
- linii;
- linii folosind directiva *LineLoop*;
- linii, folosind directiva *LineStrip*;
- triunghiuri;
- triunghiuri, folosind directiva *TriangleFan*;
- triunghiuri, folosind directiva *TriangleStrip*;

### 2.3 Utilizarea culorilor în OpenGL

OpenGL este o bibliotecă grafică bazată pe stări. Acest lucru înseamnă că setarea unei culori curente la un moment dat va determina utilizarea acesteia până la momentul invocării unei noi culori pentru desenarea pixelilor. Setarea culorii de desenare se face folosind directiva **GL.Color3()** sau **GL.Color4()**.

OpenGL acceptă două moduri de specificare a culorii: modul de specificare al culorilor (în care valoarea culorii este încărcată dintr-o listă de constate pre-specificată care este prefixată cu cuvântul-cheie „Color” - *Color.<culoarecurentă>*) și modul RGBA (care permite specificarea intensității culorilor pentru fiecare canal roșu, verde și albastru, cu specificarea opțională a valorii canalului alfa – canalul de transparență – **GL.Color4()**). În modul RGBA, valoarea intensității este transmisă printr-o valoare de 0.0 dacă culoarea

lipsește pe un anumit canal și 1.0 pentru intensitatea maximă a respectivei culori (culoare saturată). Astfel, pentru negru vom avea tupla (0.0, 0.0, 0.0) iar pentru alb (1.0, 1.0, 1.0).

Spațiul de culori depinde de numărul de octeți rezervat de placa video pentru desenarea pixelilor. Uzual acesta este de 24 sau 32 biți (8 biți per canal de culoare, cu opțiune pentru transparență<sup>1</sup>).

Pentru a afișa scena randată, cel puțin o culoare este necesară a fi utilizată ca și culoare curentă (metoda **Color3()** va fi chemată cel puțin o dată în interiorul buclei de randare a scenei). În caz contrar scena nu va fi vizibilă.

Dacă vertexurile care compun o primitivă grafică (exceptând punctul) au culori diferite, se va realiza un gradient de culoare la nivelul vertexurilor fundamentale (informația de culoare pentru fiecare canal va varia între valorile setate pentru fiecare vertex specificat, ca și pentru punctele desenate intermediare între vertexuri).

### 3. Probleme de rezolvat



1.	Care este ordinea de desenare a vertexurilor pentru aceste metode (orar sau anti-orar)? Desenați axele de coordonate din aplicația-template folosind un singur apel <b>GL.Begin()</b> .
2.	Ce este <i>anti-aliasing</i> ? Prezentați această tehnică pe scurt.
3.	Care este efectul rulării comenzii <b>GL.LineWidth(float)</b> ? Dar pentru <b>GL.PointSize(float)</b> ? Funcționează în interiorul unei zone <b>GL.Begin()</b> ?
4.	Răspundeți la următoarele întrebări (utilizați ca referință eventual și tutorii OpenGL Nate Robbins): <ul style="list-style-type: none"> <li>• Care este efectul utilizării directivei <i>LineLoop</i> atunci când desenate segmente de dreaptă multiple în OpenGL?</li> <li>• Care este efectul utilizării directivei <i>LineStrip</i> atunci când desenate segmente de dreaptă multiple în OpenGL?</li> <li>• Care este efectul utilizării directivei <i>TriangleFan</i> atunci când desenate segmente de dreaptă multiple în OpenGL?</li> <li>• Care este efectul utilizării directivei <i>TriangleStrip</i> atunci când desenate segmente de dreaptă multiple în OpenGL?</li> </ul>
5.	Creați un proiect elementar. Urmăriți exemplul furnizat cu titlu de demonstrație - <b>OpenGL_conn_ImmediateMode</b> . Atenție la setarea <i>viewport</i> -ului.
6.	Urmăriți aplicația „ <i>shapes.exe</i> ” din tutorii OpenGL Nate Robbins. De ce este importantă utilizarea de culori diferite (în gradient sau culori selectate per suprafață) în desenarea obiectelor 3D? Care este avantajul?
7.	Ce reprezintă un gradient de culoare? Cum se obține acesta în OpenGL?

<sup>1</sup> Dacă transparența nu este utilizată, modul pe 32 de biți va fi completat cu 8 biți de *padding*.

8.	Creați o aplicație care la apăsarea unui set de taste va modifica culoarea unui triunghi (coordonatele acestuia vor fi încărcate dintr-un fișier text) între valorile minime și maxime, pentru fiecare canal de culoare. Ce efect va apare la utilizarea canalului de transparență? Aplicația va permite modificarea unghiului camerei cu ajutorul mouse-ului. Folosiți documentația disponibilă la <a href="https://opentk.net/api/OpenTK.Input.Mouse.html">https://opentk.net/api/OpenTK.Input.Mouse.html</a>
9.	Modificați aplicația pentru a manipula valorile RGB pentru fiecare vertex ce definește un triunghi. Afișați valorile RGB în consolă.
10.	<b>Ce efect are utilizarea unei culori diferite pentru fiecare vertex atunci când desenați o linie sau un triunghi în modul <i>strip</i>?</b>