



# Prelucrare Grafică

Proiect:

Redarea în OpenGL a unei scene de obiecte

**Facultatea:** Automatică și Calculatoare

**Specializarea:** CTI

**An universitar:** 2021 – 2022

**Student:** Bozdog Raluca-Delia

**Grupa:** 30233

**Profesor laborator:** Nandra Constantin

# Cuprins

1. Prezentarea temei .....	3
2. Scenariul .....	3
2.1 Descrierea scenei și obiectelor .....	3
2.2 Funcționalități .....	4
3. Detalii de implementare .....	4
3.1 Funcții si algoritmi .....	4
3.1.1 Soluții posibile .....	7
3.1.2 Motivarea abordării alese .....	7
3.2 Modelul grafic .....	8
3.3 Structuri de date .....	8
3.4 Ierarhia de clase .....	8
4. Prezentarea interfeței grafice utilizator/manual de utilizare .....	9
5. Concluzii si dezvoltări ulterioare .....	10
6. Referințe .....	10

## **1. Prezentarea temei**

Tema proiectului este realizarea unei scene tridimensionale de obiecte în OpenGL, care să dea sentimentul de realitate, având cât mai mult funcționalități și efecte care să îi dea culoare. Scena aleasă în acest proiect este un peisaj de iarnă, cu munți, brazi ninși și un lac în care se reflectă cerul. În scenă se pot întâlni numeroase animale care îmbogățesc imaginea, cum ar fi vulpe, cerb, lupi, câine, și chiar o pasăre care își mișcă aripile. Scena este completată și de un foc animat și o lampă aprinsă, ambele fiind surse de lumină punctiforme. Peisajul poate fi observat atât pe timp de zi, cât și pe timp de noapte, cu ceață sau cu ninsoare. Pe lângă modul în care sunt vizibile texturile, utilizatorul poate alege să vadă muchiile sau vârfurile care compun scena. De asemenea, la dispoziția utilizatorului este pusă și posibilitatea de a face un tur al scenei, fără să controleze el poziția în scenă.

Scopul proiectului este familiarizarea cu librăria și conceptele OpenGL și utilizarea acestora în practică folosind programare în Visual Studio, făcând apel la librăriile auxiliare GLFW și GLM, în tandem cu învățarea modelării 3D în Blender.

## **2. Scenariul**

### **2.1. Descrierea scenei și a obiectelor**



Construirea scenei s-a realizat cu ajutorul software-ului Blender, folosind modele .obj și .fbx preluate din mediul online, unele cu modificări aduse pentru integrarea în scenă sau pentru reducerea numărului de vârfuri, deoarece dimensiunea scenei era inițial extrem de mare.

S-a încercat aducerea scenei cât mai aproape de realitate, motiv pentru care scena este relativ complexă, cu numeroase zone de interes: lacul, focul, pasărea animată, pădurea lupilor, omul de zăpadă, fiecare dintre acestea aducând un plus de viață scenei.

## **2.2. Funcționalități**

Scena permite mai mult posibilități de navigare prin intermediul tastaturii și a mouse-ului. Pentru a crește fotorealismul scenei au fost adăugate efecte de umbră, lumini punctiforme, skybox pentru încadrarea scenei într-un cadru inteligibil, efect de ceață, ninsoare, foc și reflexie. Se oferă posibilitatea de a parcurge un tur predefinit al scenei. Se poate vizualiza scena doar ca muchii sau vârfuri de obiecte.

## **3. Detalii de implementare**

### **3.1. Funcții și algoritmi**

Pe lângă funcțiile inițiale din componența laboratorului s-au mai creat funcții pentru inducerea efectului de lumină punctiformă, pentru crearea efectului de ninsoare și pentru definirea turului virtual.

### **Ceața**



Pentru implementarea efectului de ceață s-a ales abordarea prezentată în laborator pentru ceața cu dependență pătratică. La pornirea efectului se stabilește o densitate de ceață implicită care dă valoare parametrului `fogFactor` ce se calculează în funcția `computeFog()` din fragment shader-ul principal al proiectului. În funcție de `fogFactor` se calculează culoarea fiecărui fragment ca o medie ponderată a culorii din textura obiectului și a unei nuanțe de gri care induce ceață. Densitatea ceții crește cu îndepărtarea de obiecte și scade odată cu apropierea.

### **Ninsoarea**

Pentru ninsoare s-a generat în Blender o sferă de mici dimensiuni care este importată în proiectul OpenGL. Se rețin coordonatele de poziție, scalare și o viteză generate aleatoriu pentru un număr de 400 de fulgi de zăpadă. Ei se generează la o înălțime prestabilită și coboară înspre pământ. La atingerea pe sol dispar și se poziționează din nou în zona de sus a scenei, gata pentru un nou ciclu de cădere.



### **Reflexia**

Reflexia a fost implementată pentru un obiect importat independent în proiect, ca un obiect separat, cu shader dedicat. Pe suprafața lacului se reflectă imaginea cerului din skybox în funcție de poziția observatorului. Imaginea de pe skybox primește o pondere de 90% din culoarea finală a fiecărui pixel de pe lac, 10% fiind o culoare de albastru.

### **Efectul de foc – lumini punctiforme**

Pe lângă sursa de lumină globală, în scenă sunt prezente și două surse de lumină punctiforme: felinarul și focul. Felinarul este o lumină punctiformă cu parametri de atenuare constanți, motiv pentru care este o lumină stabilă, care nu variază în timp. Pentru a induce efectul de foc a fost animată lumina punctiformă corespunzătoare lui prin modificare parametrilor liniar și pătratic în funcție de ciclul de randare a scenei. Fiecare dintre aceste lumini are o funcție de calcul al aportului său pentru iluminarea scenei în fragment shader-ul principal.



### **Animația de componentă – pasărea care își mișcă aripile**

Pentru a realiza animația de componentă pentru pasăre s-au decupat aripile acesteia ca obiecte diferite în Blender. S-au poziționat cele 3 obiecte nou create (aripa stânga, aripa dreapta și corpul) la poziția dorită în scena de obiecte. Pentru fiecare aripă s-au aflat din Blender coordonatele a două puncte de pe axul cu care acestea se prind de corpul păsării, pentru a afla vectorul în jurul căruia trebuie să se facă rotația pentru mișcarea aripilor. Aripile se rotesc până la un unghi în sus, după care coboară, într-un ciclu continuu. Ambele aripi se rotesc cu același unghi în modul, însă cu semn diferit una față de cealaltă.



### **Turul virtual**

Pentru turul virtual s-a realizat mai întâi o scriere într-un fișier a atributelor camerei pentru o parcurgere manuală a peisajului. Apoi, de fiecare dată când este selectată opțiunea de tur virtual se citesc datele din acest fișier și camera este modificată pentru a corespunde exact deplasării dorite.

#### **3.1.1 Soluții posibile**

#### **3.1.2 Motivarea abordării alese**

Pentru realizarea turului virtual s-ar fi putut implementa și alte abordări, dar cea aleasă a fost considerată cea mai facilă și necesită doar cunoștințe deja deținute din cadrul academic.

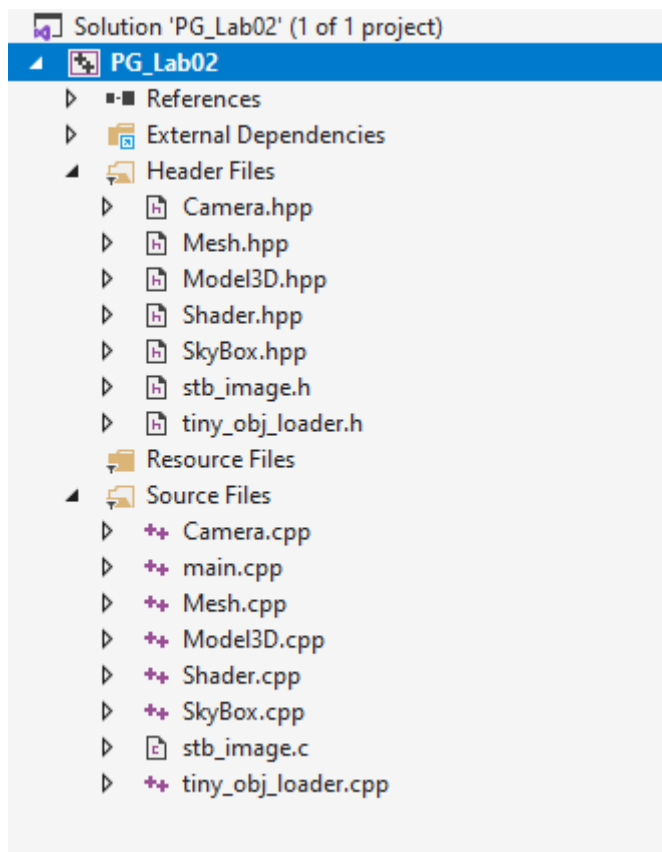
### **3.2.Modelul grafic**

Modelul grafic a fost în totalitate inițializat în Blender, fiind mai apoi exportat pentru a fi folosit în codul din Visual Studio. Aripile păsării au fost importate ca obiecte separate de scenă și diferite, pentru a putea fi animate.

### **3.3.Structuri de date**

Structura de date specifică folosită este array-ul de valori float ce reține caracteristicile generării de ninsăare. Acesta a fost alocat pe heap pentru a eficientiza utilizarea memoriei.

### **3.4.Ierarhia de clase**





#### 4. Manual de utilizare

J, L – rotație sursă de lumină direcțională

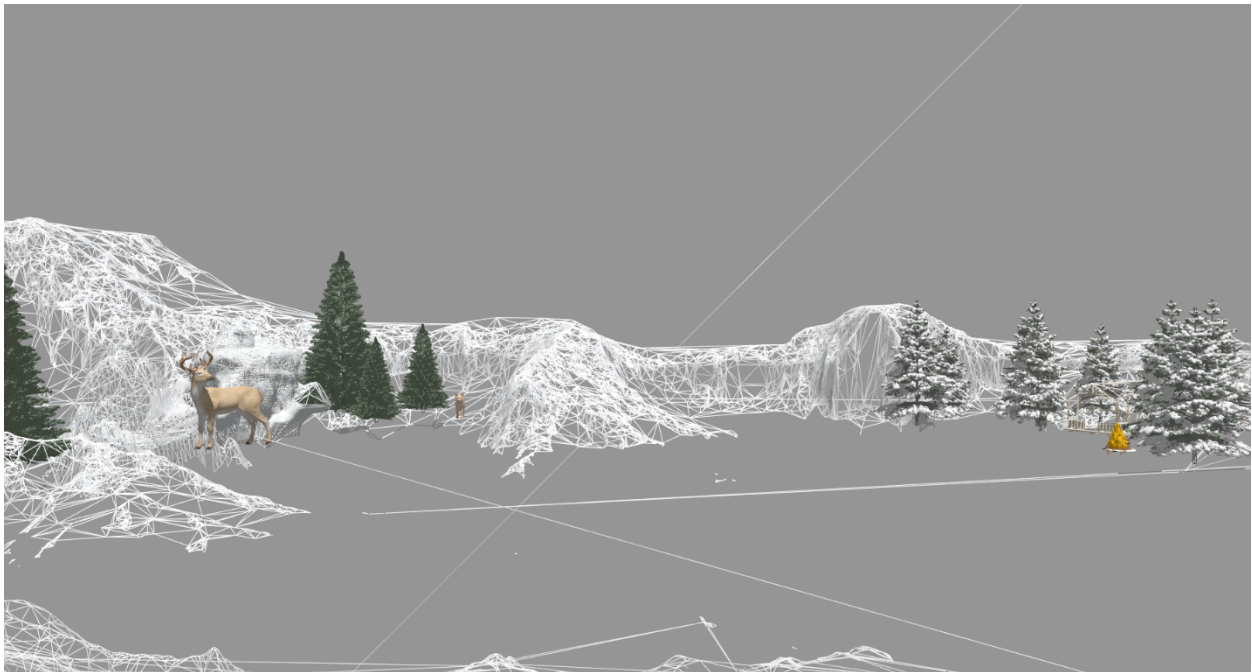
W,S – deplasare cameră față spate

A,D – deplasare cameră stânga dreapta

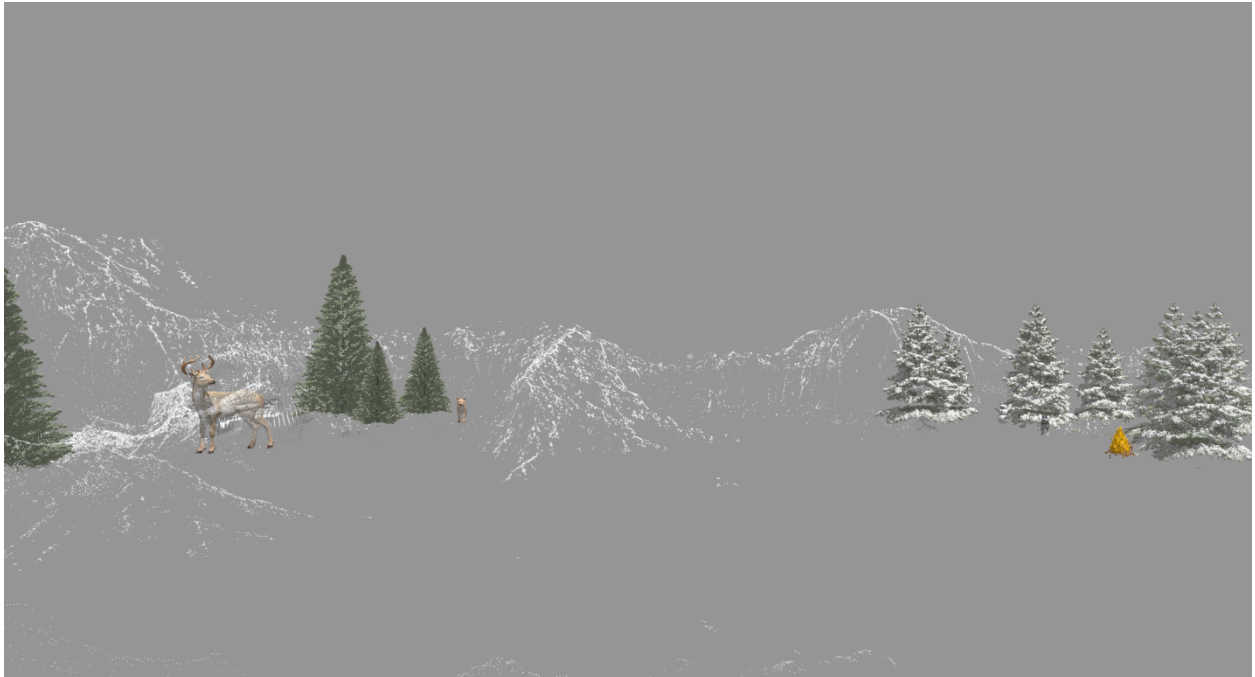
Z,X – rotația camerei în plan vertical

C,V – rotația camerei în plan orizontal

R – vizualizare wireframe



T – vizualizare vertex



Y – activare/dezactivare ninsoare

U – activare/dezactivare ceata

I – comutare mod noapte

O – pornire/întrerupere tur virtual

## **5. Concluzii și dezvoltări ulterioare**

Proiectul a necesitat un volum mare de muncă de editare a scenei, dar și cunoștințe considerabile de matematică și, desigur, de grafică.

Dezvoltările ulterioare ar putea include: introducerea efectului aurora boreală, animații mai realiste ale focului, animația animalelor.

## **6. Referințe**

- <https://learnopengl.com/Introduction>
- <https://www.turbosquid.com/Search/3D-Models/free>
- <https://free3d.com/>
- <https://www.cgtrader.com/free-3d-models>
- <https://www.blender.org/>