

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

SPECIALIZAREA CALCULATOARE SI TEHNOLOGIA

INFORMATIEI

Documentatie Proiect OpenGL

Prelucrare Grafica

Student: Mosila Luciana

Data: 17.01.2024

Grupa: 30236

Cuprins

Prezentarea temei.....	3
Scenariul.....	4
Descrierea scenei si a obiectelor.....	4
Functionalitati	5
3.Detalii de implementare	6
3.1. Functii si algoritmi.....	6
3.2 Modelul grafic	7
3.3 Structuri de date	8
3.4 Ierarhia de clase	8
Prezentarea interfetei grafice utilizator	10
Manual de utilizare	12
Exista urmatoarele functionalitati:.....	12
Concluzii si dezvoltari ulterioare	13
Referinte.....	15

Prezentarea temei

Tema Proiectului: Dezvoltarea unei Scene OpenGL

Proiectul propus vizează dezvoltarea unei scene interactive în mediul OpenGL, oferind utilizatorului posibilitatea de a explora un mediu 3D detaliat. Scopul principal este nu doar să ilustreze modul în care un calculator procesează o scenă de obiecte, ci și să familiarizeze utilizatorul cu diverse biblioteci și librării 3D, cum ar fi OpenGL, GLFW și GLM. Scena proiectată include o varietate de obiecte, cum ar fi elemente din mediul natural, padure și efecte speciale. Interacțiunea cu această aplicație se realizează prin intermediul tastaturii, facilitând experiența utilizatorului similară cu cea dintr-un joc pe calculator.

Obiectivele proiectului sunt definite astfel:

1. Crearea unei scene 3D detaliate și oferirea posibilității de interacțiune cu obiectele prezente.
2. Implementarea unui sistem de camere plutitoare, permitând explorarea din diverse unghiuri a scenei.
3. Adăugarea de modele animate pentru a intensifica experiența utilizatorului.
4. Integrarea efectelor vizuale, cum ar fi ceața și lumina, pentru a obține o reprezentare realistă.

Tehnologiile utilizate în cadrul proiectului includ:

- OpenGL pentru randarea 3D.
- C++ pentru implementarea logicii programului.
- Utilizarea shaderilor pentru controlul aspectelor specifice ale rendering-ului.

Scenariul

Descrierea scenei si a obiectelor

Proiectul constituie o scenă grafică în mediul OpenGL, ilustrând un peisaj cu un râu ce serpuiește prin mijlocul unor dealuri împădurite. Elementele principale includ o barcă care navighează pe râu, copaci pe versanți și o casă la marginea apei, cu posibilitatea de a deschide fereastra.

Pe râu, barca este modelată și animată pentru a conferi o mișcare continuă, simbolizând navigarea pe apele liniștite. Înălțimea dealurilor este conturată prin arbori plasați strategic pentru a recrea aspectul unei păduri dens populate. Pentru casa de pe mal, s-a implementat o opțiune de deschidere a ferestrei, evidențiind aspectul interactiv al scenei.

Eficiența iluminării a fost implementată pentru a crea o atmosferă de zi autentică, având în vedere lumina naturală care cade uniform asupra peisajului. De asemenea, efectul de ceață a fost integrat pentru a adăuga profunzime și mister, aplicându-se asupra apei și a zonei de pe maluri.

Aceste aspecte tehnice, împreună cu utilizarea OpenGL și a shaderelor adecvate, contribuie la realizarea unei scene realiste și interactive, unde utilizatorul poate explora diverse elemente ale peisajului și chiar interacționa cu casa.



Functionalitati

În spatele scenei OpenGL, avem o casă frumoasă pe malul unui râu. Ce e tare e că, apăsând o tastă, poți deschide fereastra casei și să vezi tot peisajul afară.

Pe râu, ai o barcă care se plimbă. Cu o tastă specială, devii căpitan și o poți mișca în sus sau în jos pe râu.

Dacă vrei să schimbi vremea, apeși o altă tastă și apare ceață. Ceea ce face scenariul mai interesant și misterios.

În mijlocul scenei, e o lumină care e ca un soare. Cu niște taste, poți să o faci mai luminoasă toata scena.

Cu opțiunea de wireframe, poți să vezi doar contururile obiectelor și a terenului, oferindu-ți o perspectivă mai simplificată a structurii.

Există și modul de vedere punctiformă, unde obiectele sunt reprezentate doar prin puncte. Această opțiune permite analizarea detaliilor geometrice într-un mod mai detaliat.

Aceste caracteristici adaugă o dimensiune suplimentară experienței tale, permițându-ți să explorezi și să înțelegi detaliile tehnice ale scenei într-un mod mai direct și personal.

Detalii de implementare

Functii si algoritmi

În cadrul proiectului, funcțiile ``initUniforms()`` și ``initShaders()`` își asumă un rol crucial. Prin intermediul funcției ``initUniforms()``, sunt stabilite adresele locațiilor variabilelor din shadere, iar aceste variabile sunt inițializate cu valori specifice. Printre aceste valori se regăsesc detalii despre lumină, perspectiva camerei și matricea de normale. Această funcție reprezintă un element esențial pentru funcționarea corectă a programului, asigurându-se că toate variabilele sunt pregătite și accesibile în timpul procesului de randare.

De asemenea, funcția ``initShaders()`` se ocupă de încărcarea conținutului shaderelor. Fiecare obiect în scenă este, în mod obișnuit, desenat prin intermediul unui shader specific. Această funcție contribuie la asigurarea coeziunii și integrității procesului de randare, stabilind legătura esențială dintre geometria obiectelor și modul în care acestea sunt iluminate și afișate.

Pentru a conferi impresia de infinit, în proiect am introdus un skyDome în scenă. Acesta constă într-o semisferă formată din imagini, oferind iluzia unei lumi fără sfârșit. Imaginile pentru skyDome sunt încărcate din sursele specificate în cadrul laboratorului, adăugând un nivel de realism și profunzime în vizualizarea scenei 3D.

Am implementat efectul de ceață în proiect, bazându-mă pe sursele furnizate în laborator. În procesul de programare, am utilizat funcția în shadere, care se asigură că o valoare dată se încadrează într-un interval specificat. De asemenea, am integrat funcția `"mix()"`, care realizează o interpolare liniară între două valori specificate.

Modelul grafic

În procesul de creare a scenei în Blender, am început prin a modela și a textura elementele principale, inclusiv dealurile și apa. Dealurile au fost conturate cu atenție pentru a conferi peisajului o formă naturală, iar texturile au fost aplicate pentru a adăuga detalii și realitate.

Pentru a crea efectul de apă, am folosit instrumentul Sculpt Mode din Blender pentru a modela suprafața apei într-un mod care să ofere o senzație de mișcare și adâncime. Această abordare a avut scopul de a integra elemente naturale și realiste în peisaj.

Generarea skyDome-ului a implicat crearea unei semisfere care să înconjoare întreaga scenă. Texturile corespunzătoare au fost aplicate acestei structuri sferice pentru a recrea aspectul unui cer deschis și infinit.

În ceea ce privește obiectele individuale importate în scenă, precum casa, copacii, tufișurile și barca, acestea au fost integrate pentru a adăuga diversitate și detalii peisajului. Texturile pentru aceste obiecte au fost fie implementate deja, fie au fost mapate în Blender pentru a le oferi aspectul dorit.

Barca care se plimbă pe apă a fost animată pentru a aduce o notă dinamică în scenă, iar texturile și materialele au fost ajustate pentru a se potrivi cu estetica generală a proiectului.

În ansamblu, procesul de creare a scenei în Blender a implicat atenție la detalii, texturare creativă și integrarea de obiecte animate pentru a oferi o experiență vizuală complexă și captivantă.

Structuri de date

În cadrul proiectului, printre principalele structuri de date folosite se numără `Guint`, utilizat pentru stocarea adreselor variabilelor uniform din `shader`. Această structură este esențială pentru gestionarea și accesarea eficientă a datelor în timpul procesului de randare.

De asemenea, am utilizat structuri de date precum `Model` și `Shaders`, care au fost esențiale pentru organizarea și manipularea entităților 3D în proiect. Aceste structuri au permis o gestionare eficientă a obiectelor, texturilor și `shader`elor în cadrul scenei.

Nu în ultimul rând, am lucrat și cu tipuri de date standard din limbajul C++, cum ar fi `bool`, `int` și `char`, pentru a gestiona diverse aspecte ale logicii programului și pentru a asigura funcționarea corectă a acestuia.

Ierarhia de clase

În structura proiectului, fiecare clasă este însoțită de un fișier header corespunzător. Iată exemple de clase și o scurtă descriere a funcționalităților fiecăreia:

`main.cpp`:

Punctul de intrare în program, unde este creată instanța `Window` și se inițializează restul componentelor.

`Camera.cpp` și `Camera.hpp`:

Clasa `Camera` gestionează aspectele legate de perspectivă și poziționarea camerei în spațiu. Include funcționalități pentru mișcare și controlul camerei.

`Mesh.cpp` și `Mesh.hpp`:

Clasa Mesh reprezintă o rețea de triunghiuri care definește geometria unui obiect 3D. În această clasă se pot stoca și manipula vertecșii, normalele și coordonatele texturilor.

Model3D.cpp și Model3D.hpp:

Clasa Model3D gestionează un obiect 3D format din una sau mai multe plase (Mesh). Aici se pot adăuga funcționalități pentru animație, manipulare și afișare a modelelor.

Shader.cpp și Shader.hpp:

Clasa Shader oferă un mecanism pentru gestionarea shaderelor OpenGL, inclusiv încărcarea, compilarea și activarea lor în cadrul programului OpenGL.

stb_image.cpp și stb_image.hpp:

Biblioteca stb_image furnizează funcții pentru încărcarea de imagini. Este adesea folosită pentru încărcarea texturilor în cadrul proiectelor OpenGL.

tiny_obj_loader:

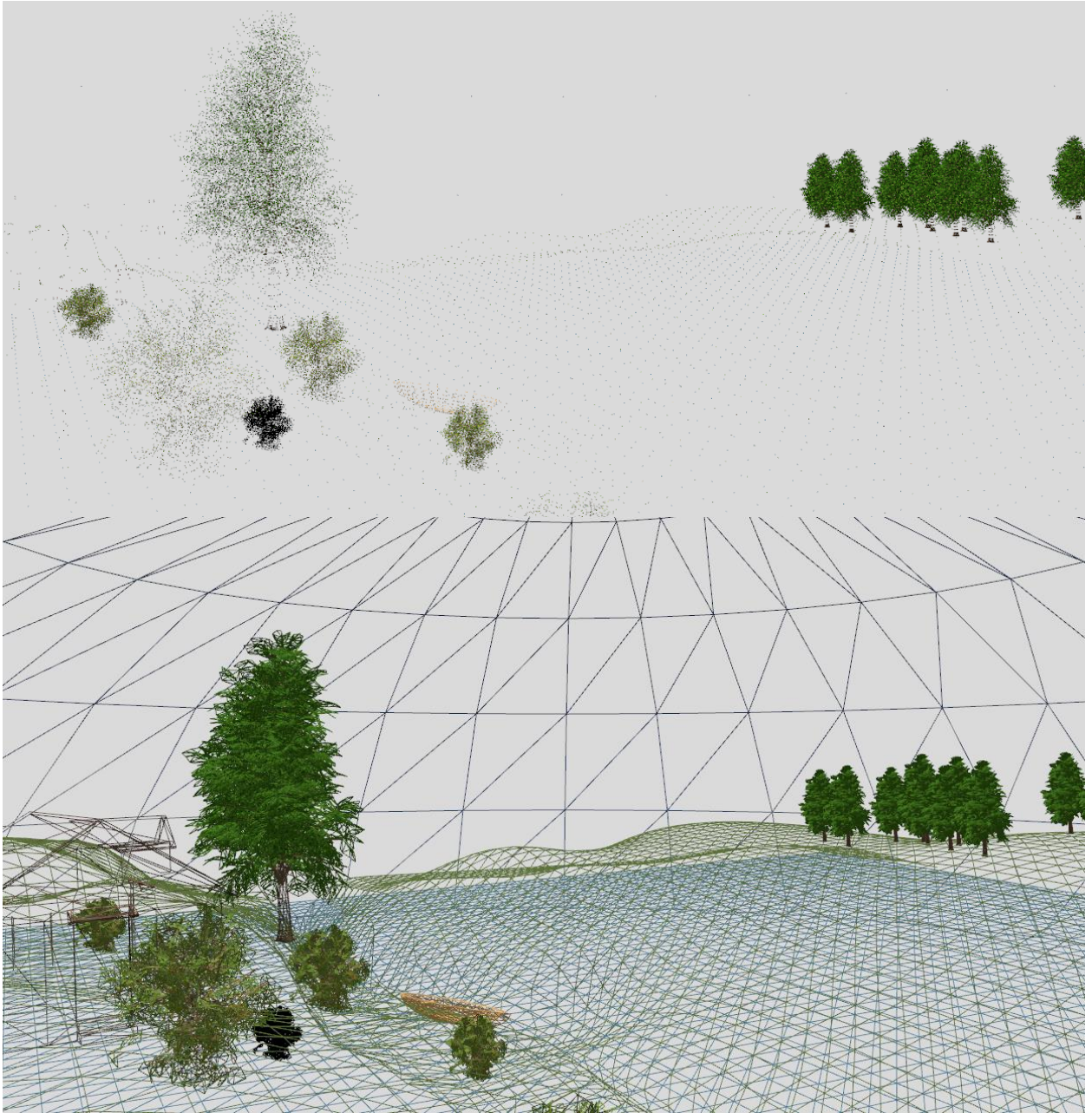
Biblioteca tiny_obj_loader este utilizată pentru încărcarea modelelor în formatul .obj. Oferă funcționalități pentru citirea și interpretarea datelor din fișierele .obj.

Window.cpp și Window.hpp:

Clasa Window reprezintă fereastra aplicației și oferă funcționalități pentru gestionarea evenimentelor de input, inițializarea contextului OpenGL și afișarea graficelor.

Prezentarea interfetei grafice utilizator

Moduri de vizualizare:



Lumina punctiforma:



Ceata:



Manual de utilizare

Exista urmatoarele functionalitati:

Deplasare:

W: Deplasare înainte

S: Deplasare înapoi

A: Deplasare la stânga

D: Deplasare la dreapta

Rotatie:

Q: Rotare spre stânga

E: Rotare spre dreapta

Înălțime:

Z: Mărire în înălțime

X: Micșorare în înălțime

Modele de Vizualizare:

1: Vizualizare punctiformă

2: Vizualizare solidă

3: Vizualizare wireframe

Iluminare:

I: Aprindere lumina punctiformă

J: Stingere lumina punctiformă

Interacțiuni Speciale:

O: Deschidere geam

P: Plimbare cu barca

Concluzii si dezvoltari ulterioare

Acest proiect a servit nu doar ca o platformă pentru aplicarea cunoștințelor în programarea OpenGL, ci și pentru înțelegerea profundă a interacțiunii dintre modelare, animație și programare. Integrarea obiectelor create în Blender a adăugat o dimensiune suplimentară de complexitate, cerând abordări creative și soluții tehnice pentru a le încorpora armonios în cadrul proiectului OpenGL.

Procesul de a da viață fiecărui detaliu al scenei a necesitat nu doar cunoștințe tehnice, ci și abilități de rezolvare a problemelor și gândire creativă. De la gestionarea resurselor la optimizarea performanței și implementarea animațiilor din cod, această experiență a oferit o perspectivă cuprinzătoare asupra dezvoltării unui proiect 3D complex.

Cu fiecare provocare întâlnită și depășită, ai acumulat nu doar competențe tehnice, ci și înțelegerea necesară pentru a aborda provocările specifice în domeniul graficii 3D. Acest proiect nu doar reflectă realizările tale în dezvoltarea software-ului, ci și creșterea ta personală ca programator și creator de experiențe interactive.

Dezvoltări Ulterioare ale Proiectului:

Mișcare Avansată a Bărcii:

Implementarea unui sistem complex de mișcare pentru barcă, permitându-i să se deplaseze pe apă în mai multe direcții. Adăugarea unui personaj în barcă, care interacționează cu mediul și poate controla mișcarea bărcii.

Simularea Vântului pentru Copaci:

Introducerea unui sistem de simulare a vântului pentru copaci, astfel încât aceștia să se miște în mod dinamic în funcție de intensitatea și direcția vântului. Această funcționalitate va adăuga un nivel suplimentar de realism în peisaj.

Interacțiuni cu Casa:

Implementarea unei funcționalități care permite deschiderea și închiderea ușii casei. Astfel, utilizatorul poate interacționa cu casa pentru a explora interiorul sau pentru a adăuga elemente noi.

Elemente Grafice Adiționale:

Integrarea unui set mai larg de elemente grafice, cum ar fi obiecte de mobilier în casă, decoruri suplimentare și alte elemente care să îmbogățească și să diversifice peisajul. Aceste aditii vor contribui la crearea unei scene mai detaliate și captivante.

Iluminare Dinamică și Schimbări de Timp:

Implementarea unui sistem de iluminare dinamică pentru a simula schimbările de timp și atmosfera în scenă. Aceasta poate include cicluri zi-noapte, variații ale luminii solare și alte aspecte care să influențeze aspectul general al scenei.

Aceste dezvoltări ulterioare vor adăuga un nivel avansat de complexitate și realism proiectului, oferind utilizatorului posibilitatea de a explora un mediu 3D în continuă evoluție și interacțiune

Referinte

- <https://free3d.com/3d-models/blender>
- <https://www.blender.org/features/modeling/>
- <https://en.wikipedia.org/wiki/OpenGL>
- https://moodle.cs.utcluj.ro/pluginfile.php/186215/mod_resource/content/2/Laboratory_work_8_RO.pdf
- - <http://www.learnopengl.com/>
- <https://wiki.ogre3d.org/tiki-index.php?page=-Point+Light+Attenuation>