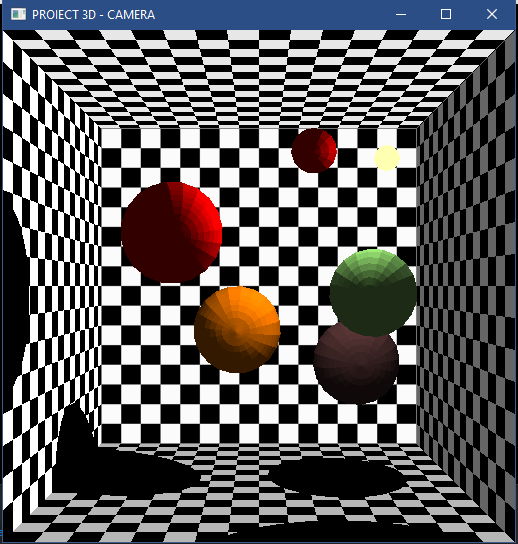
|  |
| --- |
| PROIECT 3D GRAFICĂ |
| TEMA: Room (Camera – 2016) |
| În paginile ce urmează vă voi prezenta conținutul și modul de realizare ale proiectului meu 3D, precum și felul în care am abordat tema aleasă. |
| Laborant: MINCU Radu Ștefan  Titular de curs: Conf.dr. STUPARIU Sorin |

**UNIVERSITATEA BUCUREȘTI, FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ**

**Specializarea CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI**

STUDENTĂ: IANCU Mihaela

Grupa: 252



**CUPRINS**

Introducere.........................................................................2

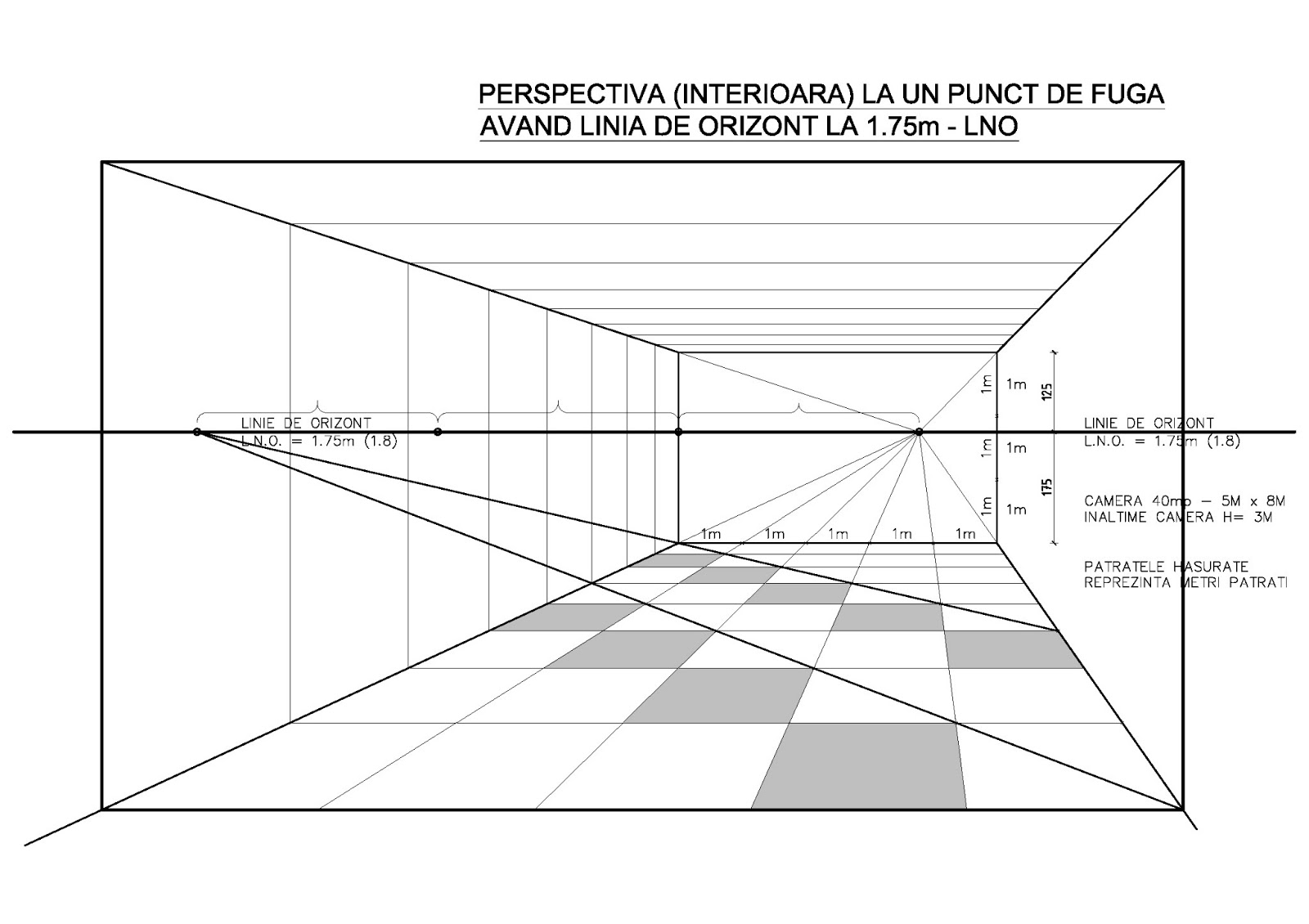
Motivația Alegerii Temei...................................................3

Structura și conținutul proiectului......................................4

Bibliografia.........................................................................8

**Introducere**

Proiectul meu 3D ilustrează o cameră desenată în perspectivă interioară la un punct de fugă (Fig.1), în care se gasesc 5 sfere (6, împreună cu sursa de lumină), care efectuează o mișcare definită de o traiectorie circulară (detalierea se va face in partea de –Structura și conținutul proiectului–). În realizarea proiectului se regăsesc obiecte cuadrice (sfere), texturare, umbre, transformări 3D, teste de adâncime și iluminare, detalierea lor se va face în capitolul –Structura și conținutul proiectului–, de la pagina 4).



**Fig.1**

**Motivația Alegerii Temei**

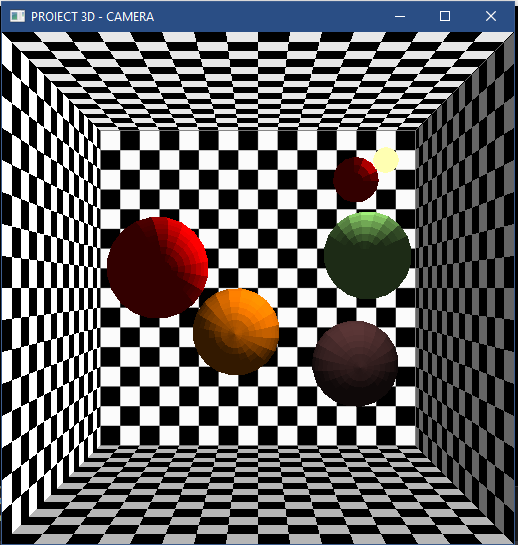
Tema mea se numește *Room* și face referire la filmul din 2016, *Room (Camera)*, ce prezinta povestea unei femei, care este ținută captivă de un anume Old Nick, împreună cu băiețelul ei în vârstă de doar cinci ani într-un șopron din grădină. Totul a început în urmă cu șapte ani când femeia, pe vremea aceea minoră, a fost răpită și adusă în „Cameră” - șopronul din grădina casei lui Old Nick. Cei doi prizonieri din acea Cameră își doresc nespus să scape de acolo.

Astfel, proiectul meu “se învârte” în jurul ideii de libertate, mișcarea circulară a celor trei sfere și restul de două, care se rotesc pe loc ilustrând încercările disperate ale mamei de a își scăpa fiul din ascunzătoare.

Cele două sfere care staționează în spate reprezintă încercările eșuate, iar cele trei în mișcare, determinarea și deznodământul fericit al poveștii (cei doi prizonieri reușesc să scape).

Cele trei sfere în mișcare se explică astfel: cele două care se rotesc în același sens reprezintă pe cei doi prizonieri care scapă, ei având ca scop comun libertatea și lupta pentru câștigarea ei; iar cea de-a treia, cea mică, care se rotește invers față de cele două îl ilustrează pe răpitor, care are o “traiectorie” diferită, viața schimbându-se și pentru el (arestul).

Am ales această temă pentru că, în primul rând, am vrut să fac ceva diferit de o simplă cameră mobilată și, în al doilea rând, pentru că am vrut să îmi testez imaginația și să văd cât de departe pot duce analogia cu filmul, reprezentată cu ajutorul unor forme și funcții grafice.



Sferele care se rotesc pe loc.

**Structura și conținutul proiectului**

Am lucrat și folosit codul 11\_C\_1\_umbra.cpp din folderul cu coduri sursă de pe moodle.

**Aspecte "punctuale" - utilizarea tehnicilor de mai jos:**

* obiecte si transformari 3D,
* iluminare,
* tehnici de amestecare / efect de ceata / teste de adancime,
* umbre,
* texturare,
* obiecte cuadrice,
* curbe / suprafete Bezier.

Obiecte și transformări 3D:

Ca obiecte, am realizat cu comanda glutSolidSphere(15,15,15);, o sferă cu raza de 15 rad.

Aceasta este plasată în planul apropiat al camerei, fiind de culoare roșie și cea mai mică în comparație cu celelalte 4 sfere (fără sursa de lumină).

Restul sferelor au fost create ca obiecte cuadrice folosind funcțiile sphere(), sphere1(), sphere2(), sphere3(), folosind glPushMatrix(), Translatef si glPopMatrix() pentru plasarea sferei în plan și glCallList(SPHERE) pentru crearea formei de sfera, împreună cu glNewList și gluNewQuadric() – voi dezvolta în momentul în care voi vorbi despre obiecte cuadrice.

Am realizat fiecare perete cu GL\_QUADS și glVertex3f: podeaua, tavanul, pereții laterali (stânga, dreapta), peretele din spate.

Folosind comenzile Rotatef și Translatef, am rotit sferele create și respectiv, le-am “plasat” în fereastra de desen. Pentru rotațe, am declarat doi vectori de coordonate statici: static GLfloat theta[]={0.0,0.0,0.0}; static GLfloat beta[]={0.0,0.0,0.0}; și o variabilă statică static GLint axis=2.0; cu ajutorul careia am putut roti sferele cu 360°. Acest lucru l-am realizat în funcția void spinsphere() în felul următor: vectorul beta realizează rotirea în sensul acelor de ceasornic (rotirea se face cu un unghi de 10°, daca se trece de 360°, se întoarce în valoarea 0° și o ia de la început); iar vectorul theta, în sens invers acelor de ceasornic.Am efectuat rotirea pe fiecare axa, X, Y, Z: glRotatef(theta[0],1.0,0.0,0.0); glRotatef(theta[1],0.0,1.0,0.0); glRotatef(theta[2],0.0,0.0,1.0);

Pentru rotirea pe loc, am pus cei trei vectori statici dupa Translatef și înainte de glCallList(SPHERE) în funcția de creare a sferei.

Pentru rotirea pe loc și traiectorie circulară, am pus cei trei vectori statici dupa glPushMatrix() și înainte de Translatef în funcția de creare a sferei.

Iluminare:

Am folosit tehnici de iluminare pentru a contura mai bine aspectul de sferă, împreună cu glShadeModel(GL\_FLAT), obiectul rămâne deschis(luminat) spre sursa de lumină și întunecat(umbrit) în partea opusă celei luminate.

Și pereții sunt luminați/umbriți ținând cont de sursa de lumină. Pentru activarea iluminării, am folosit glEnable(GL\_LIGHTING); glEnable(GL\_LIGHT0);

Sursa de lumină a fost create cu glNewList(LIGHT, GL\_COMPILE); sphere = gluNewQuadric(); gluSphere(sphere, 5.f, 20, 20); fiind un obiect cuadric.

Pentru iluminare, a fost nevoie și de proprietațile de material, reprezentate prin comanda glMaterialfv(GL\_FRONT, GL\_AMBIENT\_AND\_DIFFUSE, sphere\_mat); urmata de apelarea funcției pentru crearea obiectului (de ex. sphere();).

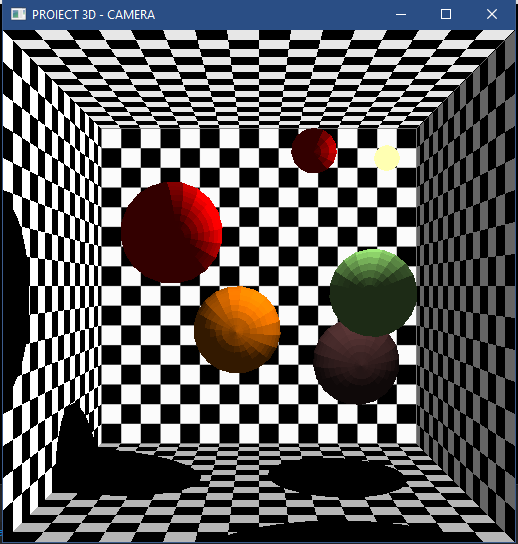
Teste de adâncime:

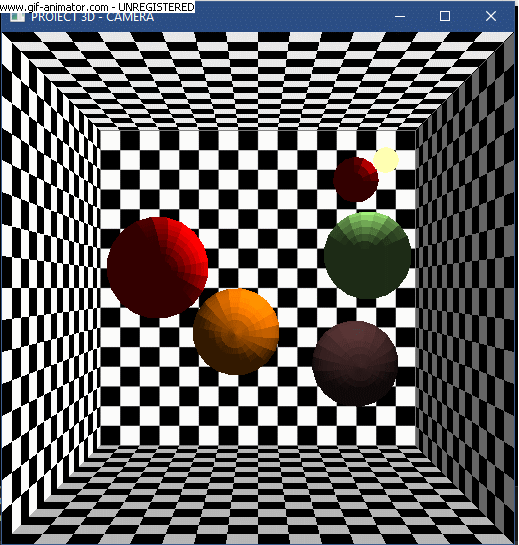
Testele de adâncime sunt folosite pentru delimitarea formei sferelor din spate și umplerea acestora. Activarea testelor se face cu glEnable(GL\_DEPTH\_TEST);, atât pentru tot programul, cât și pentru fiecare umbra proiectată pe podea și peretele din stânga. În particular, această funcție este utilă atunci când scena pe care vrem să o reprezentăm conține obiecte foarte îndepărtate de obiectele din prim plan (în cazul meu, cele două sfere, din fața peretului din spate).

Dar pentru ca testele de adâncime să funcționeze, avem nevoie de un buffer, care este încorporat în sintaxa următoare: glClear(GL\_DEPTH\_BUFFER\_BIT | GL\_COLOR\_BUFFER\_BIT | GL\_STENCIL\_BUFFER\_BIT);

Umbre:

Am creat un meniu cu următoarele opțiuni: FĂRĂ și UMBRE. FĂRĂ reprezintă obiectele reprezentate făra umbre, iar UMBRE reprezintă modul în care obiectele lasă umbra, atât în mișcare, cât și în repaus.





Pentru umbră am folosit o funcție numită shadowmatrix, cu tipul void, care are ca parametrii matricea care calculează umbra (4x4), planul “pământ” (pe care se proiectează umbra) și poziția sursei de lumină.

Matricea care calculează umbra va fi diferită pentru fiecare proiecție a ei: GLfloat leftwallshadow[4][4]; GLfloat rightwallshadow[4][4]; GLfloat floorshadow[4][4]; și se calculează folosind produsul scalar dintre vectorii poziției sursei de lumină și vectorul planului pe care se proiectează umbra(dimensiunea vectorilor fiind 4).

Ecuația planului de proiecție se calculează cu funcția finplane, care are ca parametrii cele trei puncte date (vectori de dimensiune 3) și un vector de coordinate (dimensiune 4).

Texturare:

În partea de texturare, în codul meu se găsește funcția make\_texture, care construiește textura asemănătoare unei table de șah, activându-o pe fiecare perete cu comanda

glEnable(GL\_TEXTURE\_2D); și închizându-o cu glDisable(GL\_TEXTURE\_2D);.

Două reguli sunt luate în considerare: repetarea texturii și filtrare. Ambele sunt legate de faptul că atât structura de texeli cât și cea de pixeli sunt discrete, iar scopul lor este de a indica modul în care se procedează atunci când nu există o corespondență 1:1 între cele două structuri.

Primul parametrul indică tipul de textură, fiind GL TEXTURE 2D. Al doilea și cel de al treilea reprezintă filtrarea, rolul regulii de filtrare fiind cel de a stabili ce texel îi este asociat pixelului considerat, bazându-se și pe distanța dintre cei doi. Iar cu glTexImage2D se stabilesc caracteristicile texturii.

Obiecte cuadrice:

Am folosit obiectele cuadrice la realizarea sferelor prin liste de display SPHERE, apelate cu glCallList(SPHERE). Am creat listele de display cu glNewList(SPHERE, GL\_COMPILE) sau glNewList(LIGHT GL\_COMPILE) – pentru sursa de lumină.

Realizarea sferei: crearea listei de display, crearea obiectului cuadric, apelarea funcției gluSphere (având ca prim parametru obiectul cuadric construit anterior), ștergerea obiectului creat, închiderea liniei de display cu glEndList().

Curbe/Suprafețe Bezier: nu am folosit

Alte aspecte privind proiectul:

Perspectiva a fost realizată cu ajutorul comenzilor glMatrixMode(GL\_PROJECTION); glFrustum(-100., 100., -100., 100., 320., 640.); glMatrixMode(GL\_MODELVIEW);

Am introdus un meniu pentru cele a evidenția cele două stări ale proiectului: cu umbre și fără umbre (glutCreateMenu(menu);).

Cu comanda glShadeModel (GL\_FLAT); am ales să defines mai bine suprafața sferelor pentru a vedea mai bine cum lucrează iluminarea.

**Agumentarea originalitatii si a gradului de elaborare și prezentarea contributiei fiecarui membru**

Proiectul a fost realizat doar de mine, inspirându-mă din codul sursă 11\_C\_1\_umbra.cpp de pe moodle.

Am elaborat proiectul pe baza ideii pe care mi-am făcut-o urmărind filmul. Rămânând profund impresionată de film, am ales să îmi provoc creativitatea și să transpun tema majoră a filmului într-un proiect 3D, având la dispoziție câteva sfere, spatiul unei camere și cunoștințe grafice OpenGL.

În concluzie, tema pe care am ales-o pentru a da viață proiectului a fost interesantă și captivantă, făcându-mă să lucrez cu foarte mare plăcere.

**Bibliografia**

Codul sursa 11\_C\_1\_umbra.cpp

Suportul de curs (moodle)

https://www.khronos.org/opengl/wiki/Main\_Page

https://open.gl/depthstencils

https://www.opengl.org

https://www.google.ro/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwiu4ISeiMjbAhVFEVAKHU1XCL0QjRx6BAgBEAU&url=http%3A%2F%2Fpregatirearhitectura.blogspot.com%2Fp%2Futile-desen.html&psig=AOvVaw14YjMCtRcQgw-0u2tubGUX&ust=1528684647923000 – **Fig1**

http://www.glprogramming.com/red/chapter11.html

https://www.opengl.org/discussion\_boards/showthread.php/167622-Rotate-sphere-and-move-it

https://www.opengl.org/discussion\_boards/showthread.php/132502-Color-tables