

Projektni zadatak 3.9 – Bojno polje

Modelovanje statičke 3D scene (prva faza):

1. Uključiti testiranje dubine i sakrivanje nevidljivih površina. Definirati projekciju u perspektivi ($fov=45$, $near=1$, a vrednost far zadati po potrebi) i *viewport*-om preko celog prozora unutar *Resize* metode.
2. Koristeći *AssimpNet* biblioteku i klasu *AssimpScene*, importovati sledeće modele:
 - a. tenk, i
 - b. brod.

Ukoliko je model podeljen u nekoliko fajlova, potrebno ih je sve učitati i iscrtati. Skalirati modele, ukoliko je neophodno, tako da u celosti budu vidljivi.
3. Modelovati sledeće objekte:
 - a. kopno koristeći *GL_QUADS* primitivu,
 - b. more (pored kopna), koristeći *GL_QUADS* primitivu,
 - c. zid koji se nalazi na ivici kopnenog dela scene, korišćenjem instance *Box* klase, i
 - d. projekte koji se ispaljuju iz jednog na drugo vozilo, korišćenjem instanci *Box* klase.
4. Ispisati vektorski tekst sivom bojom u donjem desnom uglu prozora (redefinirati projekciju korišćenjem *gluOrtho2D* metode). Font je *Verdana*, *14pt*, *bold*. Tekst treba biti oblika:

Predmet: Racunarska grafika
 Sk.god: 2014/15.
 Ime: <ime_studenta>
 Prezime: <prezime_studenta>
 Sifra zad: <sifra_zadatka>

Predmetni projekat - faza 1 sačuvati pod nazivom: *PF1S3.9*. Obrisati poddirektorijume **bin** i **obj**. Zadaci se **brane na vežbama**, pred asistentima.

Vreme za izradu predmetnog projekta – faze 1 su **dve nedelje**. Predmetni projekat – faza 1 vredi **15 bodova**. Način bodovanja je prikazan u tabeli.

Šifra kriterijuma	Bodovi	Opis
CVP	3	Kreiran prozor. Uključeno testiranje dubine i sakrivanje nevidljivih površina. Projekcija, kliping volumen i viewport podešeni.
M	9	Adekvatno učitani ili modelovani pa zatim prikazani mesh modeli.
I	0.5	Napravljen interfejs oko OpenGL prozora.
T	2.5	Ispisan tekst adekvatnim fontom, bojom, i na adekvatnoj poziciji.

Definisanje materijala, osvetljenja, tekstura, interakcije i kamere u 3D sceni (druga faza):

1. Uključiti *color tracking* mehanizam i podesiti da se pozivom metode *glColor* definiše ambijentalna i difuzna komponenta materijala.
2. Definirati tačkasti svetlosni izvor bele boje i pozicionirati ga gore-desno u odnosu na centar scene (na pozitivnom delu vertikalne i horizontalne ose). Svetlosni izvor treba da bude stacionaran (tj. transformacije nad modelom ne utiču na njega). Definirati normale za podlogu, projektile i zid. Uključiti normalizaciju.
3. Za teksture podesiti *wrapping* da bude *GL_REPEAT* po obema osama. Podesiti filtere za teksture da budu linearno filtriranje. Način stapanja teksture sa materijalom postaviti da bude *GL_DECAL*.
4. Zidovima pridružiti teksturu zida od cigle. Projektilima pridružiti teksturu zarđalog metala. Definirati koordinate tekstura.
5. Kopnu pridružiti teksturu peska (slika koja se koristi je jedan segment peska). Moru pridružiti teksturu vodene površine (slika koja se koristi je jedan segment vodene površine). Pritom obavezno skalirati teksture (shodno potrebi). Skalirati teksture korišćenjem *Texture* matrice. Definirati koordinate teksture.
6. Pozicionirati kameru, tako da gleda na scenu sa desne strane i odgore (ne previše izdignuta od podloge). Koristiti *gluLookAt()* metodu.
7. Pomoću ugrađenih WFA kontrola, omogućiti sledeće:
 - a. rotiranje tenka oko vertikalne ose za zadati ugao,
 - b. izbor boje ambijentalne komponente reflektorskog svetlosnog izvora, i
 - c. faktor uniformnog skaliranja broda.
8. Omogućiti interakciju korisnika preko tastature: sa *F5* se izlazi iz aplikacije, sa tasterima *T/G* vrši se rotacija za 5 stepeni oko horizontalne ose, sa tasterima *F/H* vrši se rotacija za 5 stepeni oko vertikalne ose, a sa tasterima *+/-* približavanje i udaljavanje od centra scene. Ograničiti rotaciju tako da se nikada ne vidi donja strana horizontalne podloge i da scena nikada ne bude prikazana naopako.
9. Definirati reflektorski svetlosni izvor (*cut-off=20°*) bele boje iznad broda.
10. Način stapanja teksture sa materijalom za modele tenka, broda, zida i projektila postaviti na *GL_ADD*.
11. Kreirati animaciju ispaljivanja projektila. Animacija treba da sadrži sledeće:
 - Brod ispaljuje projektil na tenk. Nakon što biva pogođen, tenk se prikaže kao uništen, tako što se pomeri malo unazad, a zatim prevrne naopako.

U toku animacije, onemogućiti interakciju sa korisnikom (pomoću kontrola korisničkog interfejsa i tastera). Animacija se može izvršiti proizvoljan broj puta i pokreće se pritiskom na taster C.

Neophodne texture pronaći na internetu. Predmetni projekat - faza 2 sačuvati pod nazivom: *PF2S3.9*. Obrisati poddirektorijume **bin** i **obj**. Zадaci se **brane na vežbama**, pred asistentima.

Vreme za izradu predmetnog projekta – faze 2 su **četiri nedelje**. Predmetni projekat – faza 2 vredi **35 bodova**. Način bodovanja je prikazan u tabeli.

Šifra kriterijuma	Bodovi	Opis
M	2	Podešeni materijali u skladu sa zahtevima zadatka.
S	8	Definisani svetlosni izvori, u skladu sa zahtevima zadatka.
T	8	Učitane, dodeljene, podešene, i mapirane texture, u skladu sa zahtevima zadatka.
K	2	Definisana kamera.
I	7	Omogućena interakcija, u skladu sa zadatkom.
A	8	Realizovana animacija, u skladu sa zadatkom.

Dodatna funkcionalnost za timove

Dvočlani tim

Prva faza:

1. Omogućiti parsiranje i učitavanje 1 formata fajla u kom se čuvaju 3d modeli (npr. .obj+.mtl, .3ds, .dae i sl.), bez upotrebe bilo kakve eksterne biblioteke (poput AssimpNet). Pre implementacije, pažljivo odabrati 3d model i uveriti se da su za njega definisani materijali, normale i texture (u svrhu testiranja moguće koristiti AssimpScene klasu, odnosno AssimpNet biblioteku). Učitati temena i indekse temena modela i modele iscrtati bez primenjenih materijala i tekstura.
2. Dodatno importovati model vojnog objekta (šator, zgrada i sl.)
3. Iscrtavanje učitanih modela realizovati pomoću Vertex Buffer Object mehanizma (sa upotrebom indeksiranja jedinstvenih temena).
4. Kopno (umesto kao ravnu podlogu) modelovati kao dva brega, korišćenjem GL_TRIANGLE_FAN primitiva.

Druga faza:

5. Proširiti parsiranje 3d modela, tako da se materijali, normale i texture učitaju iz fajla i korektno primene na učitane modele.
6. Omogućiti pomeranje pogleda kamere gore-dole i levo-desno, pomoću miša, izmenom parametara gluLookAt metode.
7. Za kopno definisati normale u svakom temenu.
8. Pomoću odgovarajućih WFA kontrola, omogućiti izmenu visine jednog od dva brega na delu scene gde je kopno.

Tročlani tim

Prva faza:

1. Omogućiti parsiranje i učitavanje 2 različita formata fajlova u kom se čuvaju 3d modeli (npr. .obj+.mtl, .3ds, .dae i sl.), bez upotrebe bilo kakve eksterne biblioteke (poput AssimpNet). Pre implementacije, pažljivo odabrati 3d modele i uveriti se da su za njih definisani materijali, normale i teksture (u svrhu testiranja moguće koristiti AssimpScene klasu, odnosno AssimpNet biblioteku). Za prvu fazu je potrebno učitati temena i indese temena modela i model kao takav iscrtati u jednoj (ili nekoliko) boja.
2. Dodatno importovati model vojnog objekta (šator, zgrada i sl.). Poželjno je da učitani modeli budu različitih formata, kako bi se demonstrirali parseri za rad sa oba tipa podržanih formata.
3. Iscrtavanje učitanih modela realizovati pomoću Vertex Buffer Object mehanizma za iscrtavanje (sa upotrebom indeksiranja jedinstvenih temena).
4. Kopno (umesto kao ravnu podlogu) modelovati kao dva brega, korišćenjem GL_TRIANGLE_FAN primitiva.

Druga faza:

5. Implementirati zumiranje u odnosu na kursor miša (tzv. point zoom), pomoću točkića miša.
6. Simulirati jednostavno i uprošćeno talasanje vode (npr. naizmenično iscrtavanje 3d objekata ili deljenjem vodene površine na nekoliko objekata, pa manipulisanjem njihovih normala, tako da se dobije efekat ispupčenog/udubljenog objekta i sl.).
7. Proširiti parsiranje 3d modela, tako da se materijali, normale i teksture učitaju iz fajla i korektno primene na učitane modele.
8. Za kopno definisati normale u svakom temenu.
9. Primeni skybox mehanizam na scenu, koristeći teksturu neba u toku zalaska sunca.