

Tentamen TNM061, 3D-grafik och animering för MT2

Tisdag 28/5 2013 kl 8-12, TP51, TP53, TP55, TP56

Inga hjälpmedel

Tentamen innehåller 7 uppgifter, vilka tillsammans kan ge maximalt 50 poäng.

För betyg G (registreras som sifferbetyg 3) erfordras minst 24 poäng.

För betyg VG (registreras som sifferbetyg 5) erfordras minst 36 poäng.

Deluppgifterna bygger inte alltid direkt på varandra, så läs t ex b) även om du inte svarar på a).

Svara kortfattat men rimligt fullständigt på uppgifterna. Förutsätt inte att läsaren redan kan svaret, utan visa med ditt svar att *du* kan ämnet och att du förmår förklara det för andra. Förklara införda beteckningar och ekvationer, och redovisa eventuella beräkningar. Rita figurer där det gör svaret tydligare. Glöm inte heller bort att förklara dina figurer i ord eller i ekvationer.

För uppgifter som ger många poäng vill jag se rimligt utförliga svar för full poäng, men för uppgifter med enstaka poäng räcker det med korta svar. Även korta svar skall vara tydliga.

Jag kommer att besöka tentamenslokalen flera gånger under tentamenstiden för att svara på eventuella frågor. Kontakta tentamensvakten om du behöver få tag i mig mellan dessa tillfällen.

Lycka till!

Stefan Gustavson

Uppgift 1 (10 p). Phongs klassiska lokala reflexionsmodell med tre termer för allmänljus, diffus reflexion och spekulär reflexion brukar traditionellt presenteras med följande förenklade ekvation:

$$I = I_a k_a + I_d k_d (\hat{N} \cdot \hat{L}) + I_s k_s (\hat{R} \cdot \hat{V})^n$$

Om man verkligen ska implementera modellen i programkod måste man däremot tänka på flera saker som inte framgår av den förenklade ekvationen:

- Ekvationen beskriver reflektionen som en skalär intensitet, I . Hur hanterar man färgat ljus och färgade ytor? Visa tydligt med en förklarad ekvation för åtminstone en av de tre termerna. (2 p)
- Vinkelberoendet för den diffusa termen beräknas med en skalärprodukt. Vad händer om vinkeln mellan de båda vektorerna är större än 90 grader? Hur hanterar man det? Hur väl motsvarar det vad som händer i verkligheten? (2 p)
- Ekvationen beskriver inverkan av en enda ljuskälla. Hur hanterar man flera ljuskällor? Svara med en modifierad ekvation. (2 p)
- Nästan samtliga ingående parametrar beror på olika sätt av vilken punkt $\vec{p} = (p_x, p_y, p_z)$ på ytan av ett visst objekt som man räknar på, men det framgår inte av ekvationen. För den diffusa termen, förklara varför var och en av I_d , k_d , \hat{N} och \hat{L} bör kunna bero av \vec{p} för att få god realism! (4 p)

Uppgift 2 (8 p). En central operation i raytracing är att beräkna riktningen för en ljusstråle på dess väg genom scenen.

- Visa (härled) hur man kan beräkna den reflekterade strålens riktning \hat{R} om man känner till den infallande strålens riktning \hat{I} och ytans normalriktning \hat{N} . (2 p)
- Vid passage av en genomskinlig yta kommer strålen vanligtvis att brytas, ändra riktning. Snells brytningslag anger sambandet mellan infallsvinkeln och utgångsvinkeln enligt $\eta_1 \sin \phi_1 = \eta_2 \sin \phi_2$, där ϕ_1 och ϕ_2 är infallsvinkeln respektive utgångsvinkeln mot ytnormalen, och η_1 och η_2 är brytningsindex för mediet strålen går ut ifrån respektive för mediet det går in i.

Visa (härled) hur man med hjälp av detta samband kan beräkna den transmitterade strålens riktning \hat{T} om man endast känner till den infallande strålens riktning \hat{I} och ytans normalriktning \hat{N} samt η_1 och η_2 . (4 p)

- För vissa fall ger Snells brytningslag ingen lösbar ekvation. Man får ett villkor $\sin \phi_2 = a$ där $a > 1$. Vad händer då? (2 p)

Uppgift 3 (6 p). Videospelet "Lego Lord of the Rings" släpptes i slutet av 2012 och upptog en ganska ansenlig del av er examinatorns fritid kring årsskiftet. Se bilder på nästa sida.

- Karaktärerna i spelet är enkla Lego-figurer, men för animeringens skull är det tusentals polygoner i varje modell, varav en stor del sitter i ansiktet. Trots att ansiktet på de flesta av figurerna är en slät och stel cylinder så behöver ögon, mun och andra detaljer i ansiktet kunna deformeras och flyttas mjukt över ytan, och då är det lämpligast att göra dem som geometriska objekt med polygoner. De mönster på figuren som inte behöver animeras är däremot gjorda med vanliga bildbaserade texturer. Det blir mer uppenbart när man ser det animeras, men även i en stillbild tagen på nära håll i hög upplösning syns det tydligt att det rörliga ansiktet är gjort med polygoner och att det stillastående mönstret på kroppen är gjort som en bildbaserad textur. Hur ser man det? (3 p)
- Härskarringen som Frodo bär omkring på igenom hela historien har en egenhet: den reflekterar inte sin omgivning, utan det man ser speglas i ringens blanka guldyta är Mordor, den ondskefulla Saurons rike. (Ja, er examinator har en tendens att titta väldigt nördigt noga på detaljer.) Den effekten är enkel att uppnå med så kallad "environment mapping", även kallat "reflection mapping". Förklara hur metoden fungerar! (3 p)



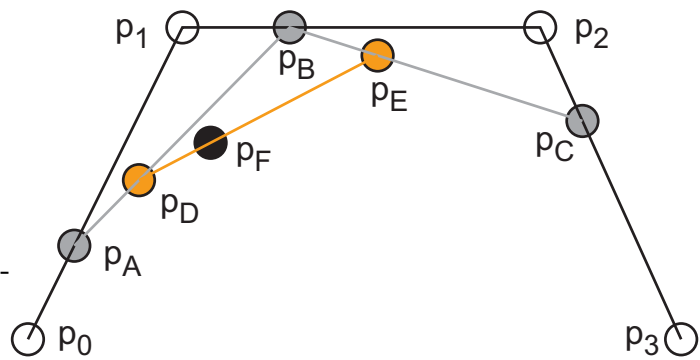
Till vänster: En typisk karaktär ur spelet *LEGO Lord of the Rings*, i det här fallet Smeagol med sin upphittade "älssskade" härskarring medan han fortfarande är ung, mer än 500 år innan huvudhandlingen börjar.

Nedan: Omslagsbilden för spelet. Barnsligt kul.



(Ovanstående två bilder © WB Interactive Entertainment, TT Games och The Lego Group.)

Uppgift 4 (6 p). Ett sätt att rita Béziérkurvor som förr användes för att rita dem för hand, men som också används än idag för att rita dem med datorhjälp, är att räkna ut punkten $p(t)$ för positionen längs ett kurvsegment som funktion av parametern t genom att göra en serie linjära interpolationer. Metoden går att använda för Béziérkurvor av godtyckligt gradtal, men för det vanligaste fallet, ett kubiskt kurvsegment med fyra kontrollpunkter p_i , gör man som följer (se figuren härintill):



Först drar man linjer mellan p_0 och p_1 , mellan p_1 och p_2 och mellan p_2 och p_3 (svarta linjer i figuren), och väljer ut interpolerade punkter på vardera linjen (grå cirklar i figuren) så att punkterna rör sig linjärt mellan linjernas första och andra ändpunkt när t går från 0 till 1. Sedan drar man två linjer mellan de tre således erhållna punkterna, p_A till p_B och p_B till p_C (grå linjer i figuren), och väljer interpolerade punkter p_D och p_E längs de linjerna på samma sätt som för de tre första linjerna (orange cirklar i figuren). Slutligen drar man en linje mellan p_D och p_E (orange linje i figuren) och väljer på samma sätt längs den en interpolerad punkt p_F (svart cirkel i figuren). Visa att den så erhållna punkten p_F uppfyller den kubiska Béziérkurvas ekvation:

$$p(t) = (1-t)^3 p_0 + 3t(1-t)^2 p_1 + 3t^2(1-t) p_2 + t^3 p_3, \quad 0 \leq t \leq 1$$

Uppgift 5 (6 p). Phongs reflexionsmodell i uppgift 1 är en så kallad lokal ljusmodell (local illumination model).

- Vad innebär begreppet "lokal" i detta fall? (1 p)
- För var och en av termerna i Phongs modell, kommentera hur väl den motsvarar vad som sker i verkligheten, och säg något om hur bristerna visar sig visuellt i det renderade resultatet. (3 p)
- För bättre realism kan man använda en global ljusmodell (global illumination). Vad är det? (2 p)

Uppgift 6 (10 p). Nedanstående programkod ritas en enkel scen med OpenGL. Funktionen `renderObject(X)` består av många olika OpenGL-kommandon som tillsammans ritas upp objektet som beskrivs av datastrukturen `X`. Arrayen `float pose[]` beskriver figurens position.

```
gl Translatef(10.0f, 5.0f, 1.0f);
gl Rotatef(pose[0], 0.0f, 0.0f, 1.0f);
gl Rotatef(pose[1], 0.0f, 1.0f, 0.0f);
gl Rotatef(pose[2], 1.0f, 0.0f, 0.0f);
drawObject(Frodobody);
gl PushMatrix();
gl Translatef(0.0f, 0.0f, 1.1f);
gl Rotatef(pose[3], 0.0f, 0.0f, 1.0f);
drawObject(Frodohead);
gl PopMatrix();
gl PushMatrix();
gl Translatef(0.5f, 0.0f, 1.0f);
gl Rotatef(pose[4], 1.0f, 0.0f, 0.0f);
drawObject(Frodol eftarm);
gl Translatef(1.0f, 0.0f, 0.0f);
gl Rotatef(pose[5], 1.0f, 0.0f, 0.0f);
drawObject(Frodol efthand);
gl PopMatrix();
gl PushMatrix();
gl Translatef(-0.5f, 0.0f, 1.0f);
gl Rotatef(pose[6], 1.0f, 0.0f, 0.0f);
drawObject(Frodori ghtarm);
gl Translatef(0.5f, 0.0f, 0.0f);
gl Rotatef(pose[7], 1.0f, 0.0f, 0.0f);
drawObject(Frodori ghthand);
gl PopMatrix();
```

a) Rita en scengraf för att beskriva samma scen i ett scengrafsbaserat API, exempelvis Java3D. (5 p)

b) Transformationerna för objekten består av flera matriser som läggs samman. Samtliga delobjekt i exemplet har både en translation och en rotation. OpenGL gör exakt som man säger i programkoden i exakt den ordning man anger det, men i Java3D så kommer sammansatta transformationer längs samma gren av effektivitetsskäl att slås ihop till en enda matris när man kompilerar scenrafen. En sådan hopslagning är däremot olämplig om någon av de ingående matriserna skall kunna ändras under programkörningen. Hur löser man det problemet i Java3D? (2 p)

c) Med ledning av namnen på objekten som ritas upp, och den mycket begränsade animeringen mellan delobjekten, kan man sluta sig till att det som ritas upp är en modell av den ganska stela verkliga Lego-versionen av figuren Frodo. Antag att man vill placera en ring på Frodos vänstra hand. Ringens position och orientering måste kunna anpassas efter hur den är modellerad, men den får antas ha rätt storlek. Ringen är ett separat objekt som beskrivs av datastrukturen `Theoneri ng`. Vilka förändringar skulle behöva göras i OpenGL-koden ovan för att få med ringen? (2 p)

d) Rita in det nya objektet (ringen) även i din scengraf! (1 p)

Uppgift 7 (4 p). Animeringarna i spelet "Lego Lord of the Rings" är av många olika slag, och har gjorts med många olika metoder, exempelvis keyframing, fysikalisk simulering, motion capture, partikelsystem och animering längs parametriska kurvor. Föreslå en lämplig metod för att animera var och en av följande händelser i spelet, och motivera varför den är lämplig!

a) Frodo (som styrs av spelaren) kastar en sten mot Gollum, men spelaren siktar illa, så stenen missar och studsar mot en klippvägg, faller ner på den ojämna marken och studsar och rullar en bit innan den stannar så att spelaren kan plocka upp den igen och försöka sikta bättre nästa gång.

b) Ett brinnande eldklot kommer flygande och landar på marken på en förutbestämd punkt.

c) En fiende "dör" i en svärdsduell med spelaren. Detta visualiseras på ett barnvänligt och Legomässigt sätt med att figuren faller i bitar, som bara behöver studsas hyfsat korrekt mot omgivningen.

d) Gandalf gör en bekymrad min, lägger huvudet på sned, kliar sig i skägget och funderar.