

Lösningförslag till tentamen TNM061 2012-06-01

För vissa av uppgifterna ges ingen fullständig lösning, utan endast en kortfattad förklaring av vad som ligger till grund för bedömningen av svaren.

Uppgift 1 (8 p)

a) *Ekvationen skall anges fullständigt och korrekt, med en figur som förklarar vad de ingående vektorerna betecknar. (3 p) En skriftlig förklaring skall ges av övriga parametrars roll, gärna med en figur. (2 p)*

b) Allmänljustermen är helt orealistisk – den modelleras med ett ”ambient”-ljus som är konstant överallt i scenen och kommer från alla håll. Det är grovt otillräckligt för de flesta tillämpningar.

Den diffusa termen är faktiskt riktigt noggrann. Många diffusa ytor har en reflexion som beter sig nästan exakt som den enkla cosinusfunktionen som skalärprodukten ger.

Den spekulära termen tar bara hänsyn till reflexioner av direkta ljuskällor, men det är en sidoeffekt av att det är en lokal reflexionsmodell. En annan defekt är att alla ljuskällor antas vara punkter, och att reflexionen inte varierar med ljusets infallsvinkel mot ytan. Phongs spekulära termen funkar hyfsat som en visuell simulering för enkla ändamål, men den har egentligen inget med verkligheten att göra.

Uppgift 2 (10 p)

a) Lokala modeller beräknar endast belysning från direkta ljuskällor. Globala modeller tar hänsyn även till interaktioner mellan objekt, exempelvis skuggor och ljusreflexioner mellan ytor. (1 p)

b) *Av figur och/eller förklaring skall följande framgå: Strålarna utgår från betraktaren, inte från ljuskällan. Spegling och brytning hanteras med rekursion på i princip samma sätt. Rekursionen kan bli djup och förgrenad, vilket kan göra renderingen väldigt tidskrävande. Diffusa reflexioner beräknas med en lokal modell, t ex Phong. Den slutliga strålens färg beräknas genom att summera alla förgrenade strålars bidrag. En förklaring av hur skuggor hanteras bedöms positivt, men krävs inte för full poäng. (5 p)*

c) Realtidsrendering är helt fokuserad på lokala ljusmodeller och renderar scenen en polygon i taget, strängt taget en pixel i taget. En global modell som raytracing kräver att hela scenen med kanske hundratusentals objekt renderas som en helhet, och dagens grafikhårdvara är inte lämpad för mer generell sökning och sortering i stora datamängder. Strålar kommer dessutom att kunna ta helt olika vägar genom scenen även om de utgår från två pixels alldeles intill varandra, och det blir inte alls lika lätt att parallellisera beräkningarna. (3 p)

Uppgift 3 (4 p) Bildbaserade texturer lagras som en samplad digital bild. Procedurella texturer lagras inte, utan beräknas med en matematisk funktion i de punkter som behövs. (1 p)

Bildbaserade texturer är enkla och snabba att använda men kräver minne, har en begränsad upplösning och är krångliga att ändra och att animera. Procedurella texturer De tar mer beräkningskraft, men de kan beskrivas mycket kompakt, är lätta att ändra och att animera eftersom de beräknas på nytt varje gång de används, och kan beräknas i godtycklig upplösning. Det är också möjligt att beskriva procedurella texturer som en funktion av tre eller fler dimensioner, något som är svårt eller rentav omöjligt med bildbaserad texturer. (3 p, någon av dessa punkter kan få saknas.)

Uppgift 4 (8 p)

Det går utmärkt. Kontrollpunkterna skall sitta på en rät linje på samma avstånd från varandra, alltså med p_0 i startpunkten, p_1 på $1/3$ av vägen, p_2 på $2/3$ av vägen och p_3 i slutpunkten.

Hastigheten är tidsderivatan av positionen som funktion av tiden. Om parametern t direkt utgör tidsvariabel så är alltså hastigheten $p'(t)$. Vi räknar endast med x-koordinaten – övriga koordinater antas vara konstanta. Expansion av uttrycket för $x(t)$ ger:

$$x(t) = (1-t)^3 x_0 + 3t(1-t)^2 x_1 + 3t^2(1-t)x_2 + t^3 x_3$$

$$x(t) = (1-3t+3t^2-t^3)x_0 + (3t-6t^2+3t^3)x_1 + (3t^2-3t^3)x_2 + t^3 x_3$$

Omgruppering av termerna efter gradtal i t ger:

$$x(t) = x_0 + t(-3x_0 + 3x_1) + t^2(3x_0 - 6x_1 + 3x_2) + t^3(-x_0 + 3x_1 - 3x_2 + x_3)$$

En rörelse från $x=0$ till $x=1$ samtidigt som parametern går från $t=0$ till $t=1$ innebär att $x(0)=0$ och $x(1)=1$, vilket insatt i uttrycket ovan direkt ger $x_0=0$ och $x_3=1$.

De resterande två kontrollpunkterna kan beräknas genom att ansätta att rörelsen skall vara linjär, alltså att uttrycket för rörelsen skall vara på formen $x(t)=A+Bt$. Skriv uttrycket som

$$x(t) = A + Bt + Ct^2 + Dt^3, \text{ identifiera termerna och sätt } C=D=0 :$$

$$A = x_0 = 0$$

$$B = -3x_0 + 3x_1 = 3x_1 \text{ (insatt } x_0=0 \text{)}$$

$$C = 3x_0 - 6x_1 + 3x_2 = -6x_1 + 3x_2 = 0 \text{ (insatt } x_0=0 \text{)} \text{ vilket ger } x_2 = 2x_1$$

$$D = -x_0 + 3x_1 - 3x_2 + x_3 = 3x_1 - 3x_2 + 1 = 0, \text{ sätt in } x_2 = 2x_1 \text{ vilket ger } x_1 = 1/3, x_2 = 2/3.$$

Ett något enklare sätt (som inte resulterar i ett ekvationssystem) kan vara att derivera uttrycket:

$$x'(t) = -3x_0 + 3x_1 + 2t(3x_0 - 6x_1 + 3x_2) + 3t^2(-x_0 + 3x_1 - 3x_2 + x_3)$$

och därefter inse att en linjär rörelse av sträckan 1 på tiden 1 måste ske med en hastighet som är konstant 1. Speciellt måste hastigheten i början och i slutet av rörelsen vara 1:

$$x'(0) = 3(x_1 - x_0) = 1 \text{ samt } x_0 = 0 \text{ enligt ovan ger } x_1 = 1/3.$$

$$x'(1) = 3(x_3 - x_2) = 1 \text{ samt } x_3 = 1 \text{ enligt ovan ger } x_2 = 2/3.$$

För att dessutom visa att rörelsen är linjär sätter vi in våra värden $x_i = i/3$ i uttrycket:

$$x'(t) = -0 + 3/3 + 2t(0 - 6/3 + 6/3) + 3t^2(-0 + 3/3 - 6/3 + 1) = 1$$

Endast en konstant 1 blir kvar. Hastigheten är alltså 1 under hela rörelsen, vilket skulle visas.

Uppgift 5 (8 p)

a) Scengrafen skall innehålla ett rotobjekt (själva spelaren) och tre animerade delar: skivtallriken samt två sammanlänkade delar för tonarmen och pickupen. Skivtallriken skall vara barn till skivspelaren, flyttas till sitt rotationscentrum med en fast translation och rotera med en animerad rotation. Tonarmen skall också vara barn till skivspelaren. Pickupen skall vara barn till tonarmen och likaså flyttas till sitt rotationscentrum med en fast translation och animeras med en rörlig rotation. De tre rörliga delobjekten skall illustreras tydligt, med lokala koordinatsystem där origo placerats i rotationscentrum och rotationen sker kring en specificerad lokal koordinataxel.

b) Ett av flera möjliga sätt att beskriva detta i OpenGL är:

```
glPushMatrix(); // Spara den ursprungliga transformationen
glTranslate(x0,y0,z0); // Flytta till önskad position för skivspelaren
renderObject(Skivspelare);
glPushMatrix(); // Spara transformationen
glTranslate(x1,y1,z1); // Flytta till rotationscentrum för skivtallriken
glRotate(0,0,1,angle1); // Snurra skivtallriken
glPopMatrix(); // Återställ transformationen
glTranslate(x2,y2,z2); // Flytta till rotationscentrum för tonarmen
glRotate(0,0,1,angle1); // Roterar tonarmen
renderObject(Tonarm);
glTranslate(x3,y3,z3); // Flytta till rotationscentrum för pickupen
glRotate(1,0,0,angle3); // Roterar pickupen
renderObject(Pickup);
glPopMatrix(); // Återställ den ursprungliga transformationen
```

Uppgift 6

a) Förklara att man kan använda en textur för att styra transparensen (alpha) hos en polygon, och rendera tomrummet mellan spjälorna som en helt genomskinlig del av ytan.

b) Beskriv bump mapping eller normal mapping, med en förklarande figur.

Uppgift 7

a) Motion capture är lämpligt, eftersom det är en komplicerad och uttrycksfull rörelse utförd av en figur med mänskliga former och proportioner. Keyframing är också ett möjligt alternativ.

b) Större saker som studsar mot omgivningen är lämpliga att animera med fysikalisk simulering.

c) Animering längs en förutbestämd bana görs lämpligen med "scripting".

d) Små och många objekt som yr omkring innebär att partikelsystem ären lämplig metod.

e) Det som beskrivs är en interaktiv rörelse som påverkar ett ledat objekt genom att man rör den sista länken i en beroende kedja (kropp-arm-hand, och det är handen som flyttas). Invers kinematik behövs för att lösa problemet på ett korrekt sätt.