

Tentamen i DH2640, Grafik- och interaktionsprogrammering, utlagd 5/5 2009

Tentan består av två delar, hemtenta samt munta.

Lösningar till hemtentan skickas via email som attachment i pdf-format till Lars Kjell Dahl <lassekj@csc.kth.se> och Yngve Sundblad <y@kth.se> senast fredag 15 maj.

Vi bekräftar mottagna email.

Spara kopia av din inlämnade hemtenta och ta med den till muntan.

Rättningen påbörjas inte förrän vid deadline och bildar underlag för muntan, som genomförs individuellt 18-20 maj. Under en kvart diskuterar vi dina svar på hemtentan.

Anmälan till muntan görs via en Doodle-länk som finns på kursens webbplats från måndag 11 maj.

Möjliga tider finns 10.15-17.45 den 18 maj och 19 maj, 10.15-14.45 den 20 maj.

Du måste formulera individuella svar med egna ord och fullständiga meningar. Du ska inte svara med direkta referat av litteratur. För varje uppgift ska du ange källorna för dina svar. Se till att ditt namn finns överst på första sidan i dokumentet och att uppgifterna är sorterade i nummerordning.

När du anger referenser, ska du ange författare, tidskrift/konferens och årtal, så långt det går. Tänk på att värdera kvalitén på referenserna. Försök också att leta efter olika källor för dina svar – det är bra att inte enbart använda ”sekundärmaterial” av typ Wikipedia (även om sådana källor naturligtvis också kan ha sin plats). Se till att relevant referens ges tillsammans med den uppgift den hör till.

Använd figurer!

Det är bra med allmänna beskrivningar, gärna kompletterade med konkreta detaljer. Undvik att skriva svepande/diffusa texter, som gör osäkert om du förstått det du skriver om. Tänk på att det är bra med informativa illustrationer/bilder. Begrepp, definitioner och förutsättningar är det bra att ange. Läs frågetexten så att du inte glömmer någonting.

Svaren per fråga bör vara 0.5 - 2 sidor långa. Poängen (2 eller 4) relaterar till förväntad längd på svaret.

Via hemsidan under rubriken "Etik och hederskodex" finns en länk <http://www.nada.kth.se/utbildning/hederskodex/> som är institutionens syn på samarbete.

Poängen från vår preliminära bedömning av hemtentan bildar tillsammans med vår bedömning av muntan underlag för betyget på tentan i skalan A-F.

Lycka till / Lasse och Yngve

VÄND!

(2) Uppgift 1

Ray tracing handlar till stor del om effektivitet. Varför blir ray tracing så tidskrävande om man inte optimerar på något sätt. Beskriv åtminstone två olika sätt att begränsa arbetet i ray tracing (utan att ändra på objekten eller upplösningen på skärmen).

(2) Uppgift 2

Gouraud shading och Phong shading ger ofta ganska olika resultat. Ge ett exempel på en typ av bild där det blir stor skillnad mellan metoderna och ett exempel på där det blir liten skillnad mellan metoderna. Motivera varför det blir skillnad och varför det ibland inte blir så stor skillnad och vad det beror på. Diskutera också skillnaderna i beräkningsarbete mellan de metoderna.

(2) Uppgift 3

I samband med datorspel vill man inte göra onödiga beräkningar och man försöker därför undvika att rita sådant som spelaren ändå inte kommer att uppfatta. Diskutera vilka möjligheter du kan komma på för att "fuska" och på det sättet spara tid. Försök hitta på åtminstone 4 olika sätt och beskriv dem.

(2) Uppgift 4

Beskriv hur radiosity-metoden fungerar (använd figur). Om betraktaren flyttar sig, vad behöver då räknas om för att man ska kunna rita ut en ny bild? Om vi i stället använder ray tracing, vad behöver då räknas om när betraktaren förflyttar sig för att man ska kunna rita ut en ny bild? Förklara den skillnad det blir.

(2) Uppgift 5

Det finns två huvudtyper av algoritmer för borttagning av skymda ytor, de som utgår från pixlar och de som utgår från objekt som lagras med den noggrannhet som datorn använder. Beskriv en algoritm av vardera typen.

(4) Uppgift 6

Antag att man vill göra ett dataspel där det ingår att navigera runt i ett landskap och sedan utföra olika uppgifter på olika platser. Antag att man ska bygga allting från början. Beskriv översiktligt de programvarudelar man behöver för grafikdelen (inkl animation och interaktion).

(2) Uppgift 7

Hur strukturerar man objektorienterade interaktionsprogram enligt MVC (Model-View-Controller)? Vad vinner man på detta? Vilka nackdelar kan det finnas? MVC är ett exempel på ett designmönster, med ursprung i arkitektur. Vad är designmönstrets främsta användning inom programmeringstekniken/datalogin?

(4) Uppgift 8

Beskriv ett antal (minst fem) andra interaktionsformer än tangentbord, mus och skärm och ange någon för- och nackdel med respektive interaktionsform.