

Grafikkprogrammering – scenegrafer

Denne uken skal du implementere støtte for flere objekter og lage objekter representert med en scenegraf.

Støtte for flere objekter

Regelen hittil er at hvis et nytt objekt lastes inn av bruker, så skal nåværende objekt slettes, noe som begrenser oss til visualisering av bare et objekt. Implementer nå støtte for flere objekter.

Dette krever forståelse av WebGL sin tilstandsmaskin og i praksis må du primært gjøre kode-refaktorering; oppgaven krever ikke implementasjon av nye WebGL funksjonaliteter eller endringer i shader-kode.

Du ønsker ikke at alle objekter plasseres oppå hverandre. Hvis applikasjonen ikke støtter dette fra tidligere, implementer støtte for enkel forflytning av objekter, som bevegelse i XZ-planet med piltaster. En enkel løsning er at forflytning beveger det siste objektet som har blitt lagt til av bruker og når et nytt objekt legges til, vil posisjonen til nåværende objekt låses. Hvis du ønsker, kan du legge til funksjonalitet for å velge objekt som forflyttes, men dette er ikke så viktig.

Generasjon av kube

Opprett et JavaScript script for håndtering av kube objekter og legg til funksjonalitet for å lage et hard-kodet kube objekt med høyde og vidde til kubens som parameter. Du kan bruke kubens fra tutorialen fra oppgave 1 som mal:

https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API/Tutorial/Getting_started_with_WebGL

I tillegg, skal kube-objektet støtte de affine transformasjonene translasjon, skalering, og rotasjon. Objektet skal ha en Model transformasjonsmatrise som kan ved draw() multipliseres med View og sendes til verteks-shader som ModelView.

Du må nå gjøre eventuell refaktorering som flytter Model-transformasjoner inn i hvert objekt som tegnes. Test kubens din ved å gjennomføre noen enkle transformasjoner på kubens i kode.

Scenegraf

Løsningen over er en enkel scenegraf med et nivå. Implementer nå en scenegraf-representasjon som fører til at scenens din representeres som et tre. En utfordring er å korrekt produsere Model-transformasjonen basert på parent-objektene til hver node.

Lag, i kode, en enkel tre-struktur med kuber (for eksempel kuber i ulike størrelse stablet oppå hverandre) og test scenegraf-implementasjonen med dette.

Effekter i scenegrafen

Hvis du ikke har gjort dette tidligere, assosier shadere med scenegrafnoder/3D-objekter, slik at ulike effekter kan oppnås for ulike objekter. Du kan for eksempel teste løsningen din ved å bruke ulike shading (per-verteks vs. per-piksel) på ulike objekter.

Robot

I kode, lag en robot bygd opp av kuber. Kubene burde støtte ulike farger (dvs. en farge for hver kube).

Webapplikasjonen burde støtte muligheten for å dynamisk legge til roboter slik at flere roboter kan visualiseres.

Bonusoppgave: i kode, opprett funksjonalitet som animerer roboten, som å vinke, rotere hodet, og gå.