

Mappeoppgave Visualisering og simulering 2023

5/10/23

Sammendrag

Dette er den foreløpige teksten til mappeoppgave høsten 2023 i Visualisering og simulering. Det kan bli små justeringer i teksten til den endelige versjonen som legges ut i Inspira etter skriftlig eksamen. Vi kan gå gjennom oppgaveteksten den 20/10, etter timene i spillmotorarkitektur.

1 Introduksjon

Oppgaven bygger på obligatorisk oppgave 1-3. Dere kan velge en plattform fritt for denne oppgaven, for eksempel mappeoppgaven fra 3D-programmering med Qt/OpenGL/C++, Unity eller Unreal. De som eventuelt velger en annen plattform skal gjøre dette etter avtale med faglærer.

Valg av plattform har ikke noe å si for karakter.

Hvis dere finner det hensiktsmessig, kan dere gjøre denne mappeoppgaven i samme prosjekt som mappeoppgave i spillmotorarkitektur. Punktene i denne oppgavene, som gis i Visualisering og simulering, vurderes da selvsagt ved mappeinnlevering i Visualisering og simulering. Tilsvarende gjelder for spillmotorarkitektur.

Oppgaven er individuell, den skal gjøres individuelt, leveres individuelt og dere får individuell karakter. Men det er selvsagt lov å diskutere problemstillinger i oppgaven underveis. Det er viktig å huske referanser til kode man har funnet på nett, diskusjoner/samtaler etc.

Det forventes at det er tydelig hvordan man skal vise/teste de enkelte oppgavene. Det er meningen at man skal implementere fysikken (Newtons 2. lov, kollisjonsdetektering/kollisjonsbehandling) og innlesing av terrengdatafil selv. Det er viktig å dokumentere implementeringen.

Denne oppgaven er av en slik type at hvert spørsmål krever litt tolkning og oppdeling i mindre deloppgaver. Diskuter, og gjør egne valg for implementering.

2 Konstruksjon og visualisering av 3d terreng

2.1

Last ned terrengdata fra Statens Kartverk for et selvvalgt område med noe kupert terreng, og valgfritt filformat SOSI eller LAS (Les om filformatene LAS og SOSI). Konverter høydedata fra valgt fil til eget format (x y z linjevis). Lag en tekstfil med ett punkt på hver linje (xyz koordinater adskilt slik at de enkelt kan leses inn). Øverst i tekstfila skal antall punkter stå på egen linje. Bruk gjerne tilgjengelig programvare til dette.

2.2

Tilpass koordinatsystemet i rendrevinduet og kameraposisjonen til datagrunnlaget. Test at de innleste punktene vises ved å tegne dem som punkter/punktsky.

2.3

Lag en regulær triangulering for datasettet, som gjennomgått i kapittel 10.6 i forelesningsnotater. Angi nabotrekanten som i kapittel 6.3.8. Indeksering av vertekser/trekanten kan gjøres ved å bruke samme ide som i 2.5-2.6 (figur 2.1).

2.4

Visualiser terrenget. Velg en passende kameraposisjon og -orientering.

3 Simulering av nedbør, og vassdrag

I alle punktene nedenfor forventes dokumentasjon av fysiske lover, modeller, utregninger etc i rapport og kildekode.

3.1

Lag en simulering av en ball (eller flere) som beveger på dette terrenget (avsnitt 2.4) i henhold til Newtons 2. lov. Det skal være mulig å velge startpunkt for en ball interaktivt. Velg et startpunkt slik at ballen legger seg i ro i en forsenkning eller lavt punkt i terrenget.

3.1.1 Kollisjonsdetektering

Det er ikke noe krav om at ballene skal rulle, men det er et pluss hvis du implementerer rulling.

Hvis du har flere baller, er det et pluss hvis du implementerer kollisjonsdetektering mellom ballene.

3.2

Simulering av nedbør. Bruk mange random genererte små partikler til å simulere regndråper. Eksperimenter med antallet og med visualiseringen og dokumenter dette.

Når regndråpene når bakken, kan de visualiseres som små baller og de skal bevege seg etter Newtons 2. lov. Kollisjon mellom disse regndråpene kan fint forekomme (uten kollisjonsdetektering/kollisjonsrespons).

3.3

Simulering av vassdrag. Les av regndråpenes posisjon fra de treffer bakken og deretter for hvert tidssteg dt . Dette tidssteget bestemmes eksperimentelt, og du må velge hvor mange posisjoner du skal registrere.

Bruk de avleste posisjonene (i planet) som kontrollpunkter for en kvadratisk splinekurve (i planet). Bruk dette til å lage og visualisere kvadratiske B-spline kurver som viser regndråpenes bevegelse langs bakken. Lag kurvene i planet, og bruk koordinatene i planet til å finne høyden på overflaten/terrenget. På denne måten løftes kurvene opp på overflaten.

3.4 Simulering av ekstremvær-effekt

Ved mye nedbør skal ballen som har lagt seg til ro i oppgave 3.1 flyte sammen med vannet som dannes. Implementer denne effekten. Forklar/dokumenter hvordan du løser dette, og ta med viktigste beregninger.

Det teller positivt hvis man legger til andre relevante objekter/effekter/funksjonalitet i tillegg til oppgavene som er gitt. Eksempel: bikvadratisk B-spline tensorproduktflate.

4 Levering

4.1

Levering i Inspira.

4.2

Frist: se Canvas.

4.3

Dere skal levere individuelt:

1. Lenke til Git repository med ferdig prosjekt
2. I Inspira: Rapport 6-10 sider. Rapporten skal følge IMRAD struktur, med referanser til ligninger/kapitler i både forelesningsnotater og andre kilder, og den skal samsvare med kildekode.
3. I Inspira: Videopresentasjon (5-6 minutter) hvor all funksjonalitet og viktigste kode vises og forklares.