UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

SYLLABUS DEL CORSO

Informatica Grafica

2627-3-E3101Q134

Obiettivi

Lo studente acquisirà competenze per la progettazione e l'implementazione di algoritmi per la computer grafica 2D e 3D. In particolare sarà in grado di applicare le competenze acquisite per sviluppare applicazioni basate sulle API di rendering OpenGL e sfruttando le potenzialità di una pipeline di rendering programmabile attraverso l'uso degli shader.

Conoscenza e capacità di comprensione
Conoscenza dei concetti fondamentali della computer grafica 2D e 3D
Conoscenza delle tecniche e algoritmi alla base della rappresentazione e gestione dei dati 2D e 3D

Conoscenza e capacità di comprensione applicate
Capacità di applicare le conoscenze per la creazione di programmi rendering
Capacità di utilizzare le API grafiche per lo sviluppo di applicazioni grafche

Autonomia di giudizio
Capacità di scegliere le tecniche più adatte per il contesto applicativo

Abilità comunicative
Saper presentare e discutere lo sviluppo di progetti di computer grafica giustificando le scelte progettuali

Capacità di apprendere
Capacità di approfondire in modo indipendente aspetti avanzati della computer grafica

Contenuti sintetici

Strumenti di costruzione, trasformazione e presentazione di modelli geometrici per la grafica tridimensionale. Elementi alla base della computer grafica standard 3D. Creazione di applicazioni di computer grafica utilizzando le tecniche di riferimento e gli standard più diffusi.

Programma esteso

Introduzione alle pipeline di rendering 3D

- Fissa
- Programmabile

OpenGL

- Introduzione alle API
- Logica di funzionamento
- Gli shaders
- Linguaggio GLSL (OpenGL Shading Language)

Strumenti di sviluppo

- G++
- GLEW
- FREEGLUT
- GLM
- ASSIMP

Matematica per la computer grafica

Il processo di rendering

- Modellazione 3D
- Trasformazioni geometriche
- Cambi di sistemi di riferimento
- Trasformazione di camera
- Trasformazione di proiezione
- Trasformazione Viewport
- Clipping
- Hidden surface removal
- Depth test

Approssimare la luce

- Modelli di Illuminamento locale
- Algoritmi di Shading

Dare i dettagli

- Texture Mapping
- Bump Mapping
- Shadow Mapping
- Environment Mapping

Cenni di Physical Based Rendering

- Modelli di illuminamento globale
- Ray Tracing

Esempi di software di modellazione 3D

- Blender
- POVray

Prerequisiti

Lo studente dovrà necessariamente avere una buona conoscenza di almeno un linguaggio di programmazione (preferibilmente C++). Necessarie conoscenze di algebra lineare e geometria.

Modalità didattica

L'insegnamento è erogato in lingua Italiana.

L'insegnamento è strutturato nel seguente modo:

40 ore di lezioni frontali in modalità erogativa ed interattiva in presenza
20 ore di esercitazioni in modalità erogativa ed interattiva in presenza
12 ore di laboratorio in modalità erogativa ed interattiva in presenza

Materiale didattico

Steven K. Feiner, Andries van Dam, John F. Hughes, Morgan McGuire, David F. Sklar, James D. Foley, Kurt Akeley, Computer Graphics: Principles and Practice, Third Edition, Addison-Wesley Professional

Graham Sellers, Richard S. Wright Jr., Nicholas Haemel, "OpenGL Superbible: Comprehensive Tutorial and Reference" 7th edition, Addison-Wesley.

Slides e dispense.

Periodo di erogazione dell'insegnamento

III° anno. Secondo Semestre.

Modalità di verifica del profitto e valutazione

L'esame prevede una prova scritta, e un progetto con una prova orale.

Dopo aver superato lo scritto, è possibile presentare il progetto all'orale, nello stesso appello o in un appello successivo, entro il sesto appello dell'anno accademico.

La prova scritta consiste in domande a risposta libera e domande a risposta chiusa sugli argomenti illustrati nel corso. Alcune domande possono essere sostituite da brevi esercizi.

Il progetto di Informatica grafica ha lo scopo di farvi prendere dimestichezza con gli argomenti visti a lezione applicando le conoscenze in nuovi contesti. Il progetto può essere realizzato in una delle seguenti modalità:

1. realizzare dei modelli 3D completi di texture.
2. analizzare tecniche di rendering avanzate implementandole in piccole applicazioni demo.

La successiva discussione consente di verificare l'apprendimento dei concetti spiegati a lezione e la loro corretta applicazione all'interno del progetto sviluppato.

Al momento non sono previste prove in itinere.

Orario di ricevimento

Su appuntamento

Sustainable Development Goals

IMPRESE, INNOVAZIONE E INFRASTRUTTURE