

Programowanie gier 3D - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Programowanie gier 3D
Kod przedmiotu	11.3-WI-INFP-PGT
Wydział	Wydział Informatyki, Elektrotechniki i Automatyki
Kierunek	Informatyka
Profil	ogólnoakademicki
Rodzaj studiów	pierwszego stopnia z tyt. inżyniera
Semestr rozpoczęcia	semestr zimowy 2018/2019

Informacje o przedmiocie	
Semestr	5
Liczba punktów ECTS do zdobycia	6
Typ przedmiotu	obieralny
Język nauczania	polski
Sylabus opracował	dr hab. inż. Marek Sawerwain, prof. UZ

Formy zajęć					
Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Laboratorium	30	2	18	1,2	Zaliczenie na ocenę
Projekt	15	1	9	0,6	Zaliczenie na ocenę
Wykład	30	2	18	1,2	Egzamin

Cel przedmiotu

- zapoznanie studentów z możliwościami środowisk programowania gier 3D z uwzględnieniem wiodących pakietów aplikacji i środowisk programistycznych przeznaczonych do tego celu,
- ukształtowanie wśród studentów zrozumienia terminologii i podstawowej funkcjonalności systemów gier komputerowych.

Wymagania wstępne

Grafika komputerowa, Programowanie obiektowe

Zakres tematyczny

Krótką prezentacją głównych etapów w historii rozwoju gier komputerowych, podkreślając rozwój możliwości interaktywnej grafiki 3D.

Istniejące systemy i środowiska programowania gier 3D. Przykładowe aplikacje/pakiety wspierające rozwój aplikacji z grafiką 3D.

Interaktywna grafika komputerowa 3D. Modelowanie geometryczne i transformacje w przestrzeni trójwymiarowej, nawigacja w przestrzeni 3D. Wirtualna Rzeczywistość jako interaktywne środowisko 3D. Reprezentacja przestrzeni 3D. Konstrukcja i umieszczanie podstawowych elementów 3D. Metody przekształcania obiektów. Reprezentacja obiektów 3D - cieniowanie i oświetlenie. Konstrukcja siatki. Konstrukcja terenu.

Animacje i interakcje w środowiskach gier 3D. Kluczowanie ruchu, symulacje zachowań oparte na modelach fizycznych. Systemy detekcji kolizji. Animacje pozycji, orientacji i skali. Interakcja z użytkownikiem.

Narzędzia wspomagające projektowanie gier 3D. Problem wydajności aplikacji czasu rzeczywistego. Techniki optymalizacji sceny graficznej takie jak, ukrywanie przesłoniętych obiektów, poziom detalu. Systemy skryptowe w systemach projektowania gier 3D.

Wybrane aspekty programowania sieciowych gier 3D, tworzenie serwera i klienta. Omówienie podstawowych algorytmów synchronizacji pozycji graczy poprzez sieć w środowisku 3D.

Metody kształcenia

Wykład: wykład konwencjonalny/tradycyjny.

Laboratorium: ćwiczenia laboratoryjne, wg listy zadań.

Projekt: praca w grupach, metoda projektu.

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
-------------	-----------------	--------------------	-------------

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Potrafi dokonać ewaluacji aplikacji (krytycznie ocenić jej treść, sposób przygotowania i jakość techniczną)	• K_W20	• sprawdzian z progami punktowymi	• Wykład
Potrafi wykorzystać zaawansowane funkcje/możliwości narzędzi służących do programowania gier 3D	• K_U29	• obserwacja i ocena aktywności na zajęciach • sprawdzian z progami punktowymi	• Laboratorium
Potrafi zaplanować i zrealizować harmonogram wieloetapowych prac projektowych	• K_W20	• sprawdzian z progami punktowymi	• Wykład
Potrafi wykonać przykładową aplikację/prototyp gry komputerowej w wybranym środowisku programistycznym	• K_U29	• projekt	• Projekt

Warunki zaliczenia

Wykład - warunkiem zaliczenia jest uzyskanie pozytywnej oceny z egzaminu przeprowadzonego w formie pisemnej.

Laboratorium - warunkiem zaliczenia jest uzyskanie pozytywnych ocen ze wszystkich sprawdzianów pisemnych z ćwiczeń laboratoryjnych, przewidzianych do realizacji w ramach programu laboratorium.

Projekt - warunkiem zaliczenia jest wykonanie wszystkich zadań projektowych, przewidzianych do realizacji w ramach zajęć projektowych oraz przygotowanie pisemnego raportu ze zrealizowanego projektu.

Składowe oceny końcowej = wykład: 30% + laboratorium: 30% + projekt: 40%

Literatura podstawowa

1. Vince J.: Virtual Reality Systems, Addison Wesley, Cambridge, 1995.
2. Bociek B.: Blender. Podstawy modelowania, Helion, 2007.
3. Goldstone W., Projektowanie gier w środowisku Unity 3.x, Helion, 2012.
4. Geig M., Unity. Przewodnik projektanta gier, Helion, 2015.
5. Ross, E., Ross, .J: Unity i C#. Podstawy programowania gier, Helion 2018.

Literatura uzupełniająca

1. Flemming B., Dobbs D.: Animacja cyfrowych twarzy, Helion, 2002.
2. Creighton R.H, Unity 4.x Game Development by Example: Beginner's Guide, Pack Pub, 2013.
3. Pereira V., Learning Unity 2D Game Development by Example, Pack Pub, 2014.
4. Smith S., Queiroz C., Unity 4.x Cookbook, Pack Pub, 2013.
5. DoppioSlash, C.: Physically Based Shader Development for Unity 2017: Develop Custom Lighting Systems, Apress 2018.

Uwagi

Zmodyfikowane przez dr hab. inż. Marek Sawerwain, prof. UZ (ostatnia modyfikacja: 27-04-2018 13:30)

Wygenerowano automatycznie z systemu SylabUZ