**Nazwa przedmiotu:**

Programowanie gier komputerowych

**Koordynator przedmiotu:**

Tomasz Martyn

**Status przedmiotu:**

Fakultatywny dowolnego wyboru

**Poziom kształcenia:**

Studia II stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Przedmioty techniczne - zaawansowane

**Kod przedmiotu:**

PGK

**Semestr nominalny:**

4 / rok ak. 2012/2013

**Liczba punktów ECTS:**

5

**Liczba godzin pracy studenta związanych z osiągnięciem efektów uczenia się:**

udział w wykładach: 30h
przygotowanie do wykładów (przejrzenie slajdów, notatek, literatury): 12h
przygotowanie do zajęć laboratoryjnych: 12h
udział w zajęciach laboratoryjnych: 6 x 2h = 12h
praca nad projektem: 70 h

Suma: 30 + 12 + 12 + 12 + 70 = 136h

**Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:**

2,6

**Język prowadzenia zajęć:**

polski

**Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:**

3

**Formy zajęć i ich wymiar w semestrze:**

|  |  |
| --- | --- |
| Wykład:  | 30h |
| Ćwiczenia:  | 0h |
| Laboratorium:  | 15h |
| Projekt:  | 15h |
| Lekcje komputerowe:  | 0h |

**Wymagania wstępne:**

umiejętność programowania w języku C++
podstawowa znajomość API graficznych OpenGL i DirectX

**Limit liczby studentów:**

32

**Cel przedmiotu:**

Praktyczne wprowadzenie do problematyki projektowania i implementowania bibliotek programowych, określanych jako silniki gier, przy wykorzystaniu współczesnego sprzętu graficznego.

**Treści kształcenia:**

Przedmiot poświęcony jest problematyce wytwarzania gier komputerowych przy wykorzystaniu współczesnego sprzętu graficznego. Główny nacisk wykładu położony jest na projektowanie i programowanie wieloplatformowych bibliotek programowych określanych jako silniki gier, ze szczególnym uwzględnieniem architektury najważniejszego z modułów tych silników, jakim jest silnik graficzny. Na kolejnych jednostkach wykładowych przedstawiana jest budowa poszczególnych modułów takich silników oraz wykorzystywane w nich algorytmy i techniki programistyczne. Między innymi omawiane są: efektywne techniki zarządzania pamięcią, zarządzanie zasobami gry, architektura i składniki wieloplatformowego renderera oraz różne aspekty i metody renderingu, efektywne techniki zarządzania obiektami sceny przy wykorzystaniu grafów sceny, różnego rodzaju efekty specjalne, problematyka modelowania i przemieszczania kamery po terenie, metody animacji postaci wykorzystywane w grach.

Wykładowi towarzyszą seanse laboratoryjne, w trakcie których uczestnicy przedmiotu realizują zadania programistyczne przy komputerach pod kierunkiem prowadzącego. Zadania dotyczą w głównej mierze różnych metod i technik stosowanych w realistycznej grafice czasu rzeczywistego, ze szczególnym uwzględnieniem wykorzystania w tym celu programowalnych jednostek potoku graficznego za pomocą programów shaderów. Ćwiczenia te mają na celu przede wszystkim sprawdzenie i wykorzystanie w praktyce wiedzy nabytej na wykładach. Jednakże niektóre z ćwiczeń dotyczą również zagadnień technicznych, które nie są poruszane na wykładach, i jako takie ćwiczenia te powinny być traktowane jako swego rodzaju uzupełnienie wykładów.

W skład przedmiotu wchodzi również projekt, w ramach którego uczestnicy mają za zadanie albo zaprojektowanie i zaimplementowanie własnego prostego silnika i następnie wykorzystanie go do utworzenia aplikacji demonstrującej działanie silnika, albo utworzenie prostej gry przy wykorzystaniu silników dostępnych publicznie.

**Metody oceny:**

Ocena określana jest na podstawie projektu oraz liczby punktów uzyskanych podczas realizowania ćwiczeń laboratoryjnych.

**Egzamin:**

nie

**Literatura:**

1. H. Eberly:3D Game Engine Design, 2nd ed., Morgan Kaufmann 2006.
2. F.Luna: Introduction to 3D Game Programming with Direct X 9.0c: A Shader Approach, Wordware Publishing, 2006.
3. T.Moller, E. Haines, T. Akenine-Moller: Real-Time Rendering, 2nd ed., Ak Peters, 2002.
4. J.C. Hart (et. al): Real-Time Shading, AK Peters, (2002).
5. Perełki programowania gier, tomy 1-3, Helion 2003.
6. R. Fernando, M.J.Kilgard: Język Cg. Programowanie grafiki w czasie rzeczywistym, Helion 2003.
7. D.Shreiner, M.Woo, J.Neider i T.Davis: OpenGL Programing Guide: The Official Guide to OpenGL, Version 2, 5th Edition, Addison-Wesley 2005.
8. R.J. Rost: OpenGL Shading Language, 2nd Edition, Addison-Wesley 2006.
9. Cg Toolkit: User's Manual, A Developer's Guide to Programmable Graphics, Release 1.2, NVIDIA Corporation 2004.
10. R. Fernando: GPU Gems: Programming Techniques, Tips, and Tricks for Real-Time Graphics, Addison-Wesley Professional 2004.
11. M.Pharr, R.Fernando: GPU Gems 2: Programming Techniques for High-Performance Graphics and General-Purpose Computation, Addison-Wesley Professional 2005

**Witryna www przedmiotu:**

http://eres.elka.pw.edu.pl/eres/wwersje$.startup?Z\_ID\_PRZEDMIOTU=PGK&Z\_NR\_WERSJI=1&Z\_CHK=23316

**Uwagi:**

## Efekty przedmiotowe

### Profil ogólnoakademicki - wiedza

**Efekt PGK\_W01:**

Ma wiedzę na temat architektury i implementacji silnika gry

Weryfikacja:

projekt

**Powiązane efekty kierunkowe:** K\_W05, K\_W07, K\_W08, K\_W10

**Powiązane efekty obszarowe:** T2A\_W04, T2A\_W05, T2A\_W07, T2A\_W03, T2A\_W04, T2A\_W05, T2A\_W07

**Efekt PGK\_W02:**

Ma wiedzę na temat metod grafiki czasu rzeczywistego

Weryfikacja:

zadania laboratoryjne i projekt

**Powiązane efekty kierunkowe:** K\_W05, K\_W08, K\_W09

**Powiązane efekty obszarowe:** T2A\_W04, T2A\_W07, T2A\_W03

**Efekt PGK\_W03:**

Ma wiedzę na temat funkcjonowania współczesnego sprzętu graficznego

Weryfikacja:

zadania laboratoryjne i projekt

**Powiązane efekty kierunkowe:** K\_W04, K\_W10

**Powiązane efekty obszarowe:** T2A\_W02, T2A\_W03, T2A\_W04, T2A\_W05, T2A\_W07

**Efekt PGK\_W04:**

Ma wiedzę na temat programowania współczesnych kart graficznych

Weryfikacja:

zadania laboratoryjne

**Powiązane efekty kierunkowe:** K\_W07, K\_W10

**Powiązane efekty obszarowe:** T2A\_W05, T2A\_W03, T2A\_W04, T2A\_W05, T2A\_W07

### Profil ogólnoakademicki - umiejętności

**Efekt PGK\_U01:**

Potrafi zaprojektować wieloplatformowy silnik graficzny

Weryfikacja:

projekt

**Powiązane efekty kierunkowe:** K\_U05, K\_U08, K\_U13

**Powiązane efekty obszarowe:** T2A\_U07, T2A\_U09, T2A\_U12, T2A\_U18

**Efekt PGK\_U02:**

Potrafi wykorzystywać współczesne API graficzne

Weryfikacja:

zadania laboratoryjne i projekt

**Powiązane efekty kierunkowe:** K\_U01, K\_U04, K\_U05, K\_U08

**Powiązane efekty obszarowe:** T2A\_U01, T2A\_U05, T2A\_U07, T2A\_U09, T2A\_U12

### Profil ogólnoakademicki - kompetencje społeczne

**Efekt PGK\_K01:**

Potrafi realizować projekty programistyczne w zespole

Weryfikacja:

projekt

**Powiązane efekty kierunkowe:** K\_K01

**Powiązane efekty obszarowe:** T2A\_K06

**Efekt PGK\_K02:**

Potrafi samodzielnie uzyskiwać dodaktowe informacje dotyczące grafiki komputerowej

Weryfikacja:

projekt

**Powiązane efekty kierunkowe:** K\_K01

**Powiązane efekty obszarowe:** T2A\_K06