



**KAPITAŁ LUDZKI**  
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez  
Unię Europejską w ramach  
Europejskiego Funduszu  
Społecznego

**UNIA EUROPEJSKA**  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



<b>Nazwa przedmiotu</b>		<b>Kod ECTS</b>	
Programowanie grafiki 3D w OpenGL NS		11.3.1709	
<b>Nazwa jednostki prowadzącej przedmiot</b>			
Instytut Informatyki			
<b>Studia</b>			
<b>wydział</b>	<b>kierunek</b>	<b>poziom</b>	<b>drugiego stopnia</b>
Wydział Matematyki, Fizyki i Informatyki	Informatyka	<b>forma</b>	niestacjonarne (zaoczne)
		<b>moduł</b>	wszystkie
		<b>specjalnościowy</b>	wszystkie
		<b>specjalizacja</b>	wszystkie
<b>Nazwisko osoby prowadzącej (osób prowadzących)</b>			
dr Maciej Dziemiańczuk			
<b>Formy zajęć, sposób ich realizacji i przypisana im liczba godzin</b>		<b>Liczba punktów ECTS</b>	
<b>Formy zajęć</b>		7 20 godz wykł + 20 godz ćwiczeń + praca własna studenta	
Wykład, Ćw. laboratoryjne			
<b>Sposób realizacji zajęć</b>			
zajęcia w sali dydaktycznej			
<b>Liczba godzin</b>			
Wykład: 20 godz., Ćw. laboratoryjne: 20 godz.			
<b>Termin realizacji przedmiotu</b>			
2021/2022 letni			
<b>Status przedmiotu</b>		<b>Język wykładowy</b>	
fakultatywny (do wyboru)		polski	
<b>Metody dydaktyczne</b>		<b>Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne</b>	
<ul style="list-style-type: none"> <li>- Metoda projektów (projekt badawczy, wdrożeniowy, praktyczny)</li> <li>- Projektowanie doświadczeń</li> <li>- Wykonywanie doświadczeń</li> <li>- Wykład z prezentacją multimedialną</li> </ul>		<b>Sposób zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- Zaliczenie na ocenę</li> <li>- Egzamin</li> </ul>	
		<b>Formy zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- wykonanie pracy zaliczeniowej - projekt lub prezentacja</li> <li>- egzamin pisemny z pytaniami (zadaniami) otwartymi</li> <li>- ustalenie oceny zaliczeniowej na podstawie ocen częściowych otrzymywanych w trakcie trwania semestru</li> </ul>	
		<b>Podstawowe kryteria oceny</b>	
		Forma zaliczenia laboratorium: kilka prostych projektów oddawanych co drugie, co trzecie laboratoria na punkty. Forma zaliczenia wykładu: projekt na ocenę.	
<b>Sposób weryfikacji założonych efektów uczenia się</b>			

zakładany efekt kształcenia	egzamin	projekt	sprawdzian	referat	obserwacja postawy studenta
Wiedza					
K_W03	x		x	x	
P_W1	x		x	x	
P_W2	x		x	x	
P_W3	x		x	x	
Umiejętności					
K_U05		x			
P_U1		x			
P_U2		x			
Kompetencje					
K_K01					x
K_K02					x
P_K1					x
P_K2					x

### Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi

#### A. Wymagania formalne

Brak

#### B. Wymagania wstępne

Umiejętność programowania w języku, który implementuje API OpenGLa, między innymi C/C++/C#/Java. Znajomość podstawowych zagadnień z zakresu tworzenia grafiki 3D (np. w środowisku Blender) będzie bardzo pomocna, ale nie jest wymagana.

### Cele kształcenia

Celem przedmiotu jest nauczenie studentów tworzenia aplikacji wykorzystujących bibliotekę OpenGL do generowania oraz wyświetlania grafiki trójwymiarowej.

### Treści programowe

Wykład będzie omawiał podstawy programowania w OpenGLu oraz podstawy tworzenia silników prostych gier 3D. Istnieje kilka implementacji bibliotek OpenGLa m.in. dla języków C/C++/C#/Java - wykład będzie oparty na implementacji w języku C/C++.

Tematy poruszane na wykładzie:

- Wprowadzenie do biblioteki OpenGL – historia i przykłady.
- Podstawy generowania obiektów trójwymiarowych na ekranie za pomocą API OpenGL.
- Przekształcenia obiektów trójwymiarowych – transformacje i obroty obiektów w przestrzeni 3D.
- Kolory, materiały, tekstury i oświetlenie obiektów trójwymiarowych.
- Bufory w OpenGL – antialiasing, selekcja i inne zastosowania.
- Metody optymalizacji wyświetlania dużej ilości danych geometrycznych.
- Importowanie obiektów wymodelowanych w Blenderze lub w innym środowisku graficznym.
- Kilka wybranych zagadnień z zakresu tworzenia prostej gry 3D – detekcja kolizji, prosta fizyka w grach, automatyczne generowanie sceny, animacje, sprity.

Na laboratorium będziemy tworzyć proste aplikacje, gry i symulacje oddawane w formie małych projektów co drugie/trzecie laboratorium. Kolejne projekty będą prowadzone w ten sposób, aby rozwijać jeden większy projekt, który można oddać jako zaliczenie wykładu. Może być nim prosta gra lub inna aplikacja generująca grafikę trójwymiarową. Wybór języka, w którym można oddawać projekty jest dowolny, ograniczając się do tych, które wspierają omawiane zagadnienia.

### Wykaz literatury

- Janusz Ganczarski, OpenGL. Podstawy programowania grafiki 3D, Helion.
- Graham Sellers, Richard S. Wright Jr., Nicholas Haemel, OpenGL. Księga eksperta, Wydanie VII, Helion.
- Kevin Hawkins, Dave Astle, OpenGL. Programowanie gier, Helion.
- OpenGL Reference Manual, Silicon Graphics, Inc.
- Luke Benstead, Beginning OpenGL Game Programming, Cengage Learning PTR.

### Kierunkowe efekty uczenia się

K\_W03 : ma pogłębioną wiedzę na temat paradygmatów programowania oraz zaawansowanych konstrukcji programistycznych; zna aktualne trendy w językach

### Wiedza

Student:

- P\_W1: rozumie w jaki sposób obiekty trójwymiarowe są przetwarzane przez karty graficzne oraz wyświetlane na ekranie komputera (K\_W03)

<p>programowania</p> <p>K_U05 : potrafi zastosować znane algorytmy w konkretnych sytuacjach, potrafi efektywnie dobrać rodzaj algorytmu w zależności od postawionego problemu</p> <p>K_K01 : zna ograniczenia własnej wiedzy i rozumie potrzebę dalszego uczenia się</p> <p>K_K03 : potrafi i jest gotów formułować opinie na temat podstawowych zagadnień informatycznych</p>	<ul style="list-style-type: none"> <li>• P_W2: rozumie jaki wpływ mają na wygląd sceny trójwymiarowej rodzaj kamery, światło oraz materiały obiektów (K_W03)</li> <li>• P_W3: zna podstawowe techniki manipulacji grafiką trójwymiarową wykorzystywane przy tworzeniu gier trójwymiarowych (K_W03)</li> </ul>
	<p><b>Umiejętności</b></p> <p>Student:</p> <ul style="list-style-type: none"> <li>• P_U1: umie napisać program, który wykorzystuje API bibliotek OpenGL do wygenerowania grafiki trójwymiarowej na ekranie (K_U05)</li> <li>• P_U2: potrafi zaadaptować gotowy kod wykorzystujący OpenGL do swoich potrzeb (K_U05)</li> </ul>
	<p><b>Kompetencje społeczne (postawy)</b></p> <p>Student:</p> <ul style="list-style-type: none"> <li>• P_K1: zna ograniczenia wiedzy z zakresu generowania grafiki w porównaniu do pełnych możliwości dzisiejszych kart graficznych i rozumie potrzebę dalszego rozwoju (K_K01)</li> <li>• P_K2: potrafi i jest gotów formułować opinie na temat podstawowych technik generowania grafiki komputerowej (K_K03)</li> </ul>
<p><b>Kontakt</b></p> <p>mdziemiam@inf.ug.edu.pl</p>	