

SYLABUS PRZEDMIOTU

1.	Nazwa przedmiotu w języku polskim oraz języku angielskim Programowanie gier komputerowych Computer games programming
2.	Dyscyplina naukowa informatyka technika i telekomunikacja (2 ECTS) informatyka (2 ECTS)
3.	Język wykładowy polski
4.	Jednostka prowadząca przedmiot Wydział Fizyki i Astronomii
5.	Rodzaj przedmiotu (<i>obowiązkowy, do wyboru</i>) do wyboru
6.	Kierunek studiów Informatyka stosowana i systemy pomiarowe
7.	Poziom studiów I stopień
8.	Rok studiów IV rok
9.	Semestr (<i>zimowy lub letni</i>) zimowy
10.	Forma zajęć i liczba godzin laboratorium komputerowe – 30 godzin
11.	Wymagania wstępne w zakresie wiedzy, umiejętności i kompetencji społecznych dla przedmiotu Podstawy programowania Programowanie C++ Podstawy grafiki/animacji komputerowej
12.	Cele kształcenia dla przedmiotu Celem zajęć jest opanowanie języka C++ na poziomie zaawansowanym, odpowiadającym typowym wymaganiom na rozmowach kwalifikacyjnych przy zatrudnieniu jako programista C++.
13.	Treści programowe

	Studenci poznają środowisko developerów gier komputerowych. Prowadzą prosty projekt wg wskazówek prowadzącego stosując zasady i konstrukcje używane w praktycznym programowaniu gier (np. układ klas, logika gry, odpowiedni poziom abstrakcji kodu itp.). Wynikiem kursu jest gra komputerowa (lub kilka prostych mechanik gry) stworzona własnoręcznie w trakcie zajęć.	
14.	Zakładane efekty uczenia się Student: – zna zasady programowania strukturalnego oraz obiektowego, wybrane języki programowania (skryptowe, kompilowane, graficzne), a także odpowiednie środowiska programistyczne i narzędzia do tworzenia, kontroli wersji, testowania i dystrybucji gier; – zna podstawowe zasady dotyczące działania gier komputerowych. Zna sposób konstruowania klas, ich zależności i budowy podstawowego silnika do gier; – rozumie istotę i znaczenie tzw. mechaniki gry, rozumie istotę designu w tworzeniu tego typu oprogramowania; – posiada wiedzę o sposobach dystrybucji gotowych gier; – zna źródła materiałów (darmowych i komercyjnych), których można użyć w grach komputerowych; – umie zaprogramować prostą grę opartą o własny, prosty silnik (czyli układ klas C++ służących do wyświetlania modeli, poruszania, prostej kolizji); – umie wprowadzać elementy podnoszące tzw. grywalność do swojego projektu; – potrafi samodzielnie wyszukać i wykorzystać źródła materiałów do gier znajdujące się w Internecie; – potrafi aktywnie uczestniczyć w dyskusji dotyczącej własnego projektu lub projektów gier realizowanych przez innych uczestników kursu;	Symbole odpowiednich kierunkowych efektów uczenia się I1_W04 I1_U17 I1_K01, I1_K05
15.	Literatura obowiązkowa i zalecana Literatura zalecana: Perełki programowania gier. Vademecum profesjonalisty. Tom 1 – 6 Game Engine Architecture, Gregory Jason Game Programming Patterns, Robert Nystrom Skuteczny nowoczesny C++. 42 sposoby lepszego posługiwania się językami C++11 i C++14, Scott Meyers C++. 50 efektywnych sposobów na udoskonalenie Twoich programów, Scott Meyers	

16.	Metody weryfikacji zakładanych efektów uczenia się: ciągła kontrola postępów w zakresie tematyki zajęć poprzez weryfikację postępów realizacji projektów gier, dyskusja na temat przyjętych rozwiązań oraz rezultatów projektu, werbalne pytania prowadzącego dotyczące tematów związanych z charakterystycznymi aspektami programowania gier komputerowych	
17.	Warunki i forma zaliczenia poszczególnych komponentów przedmiotu: ocena projektów wykonanych w ramach laboratorium	
	Nakład pracy studenta wyrażony w godzinach zajęć oraz punktach ECTS	liczba godzin przeznaczona na zrealizowanie danego rodzaju zajęć
	zajęcia (wg planu studiów) z prowadzącym:	
	– laboratorium:	30
	praca własna studenta (w tym udział w pracach grupowych):	
	– przygotowanie do zajęć:	40
	– czytanie wskazanej literatury:	15
	– opracowanie dokumentacji:	15
	Łączna liczba godzin zajęć	100
	Liczba punktów ECTS (<i>jeśli jest wymagana</i>)	4