

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Zaawansowane programowanie w multimediami		Kod 1010841161010843667
Kierunek studiów Elektronika i Telekomunikacja	Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki	Rok / Semestr 3 / 6
Ścieżka obieralności/specjalność Multimedia i elektronika powszechnego	Przedmiot oferowany w języku: polski	Kurs (obligatoryjny/obieralny) obieralny
Stopień studiów: I stopień	Forma studiów (stacjonarna/niestacjonarna) stacjonarna	
Godziny Wykłady: 1 Ćwiczenia: - Laboratoria: 2 Projekty/seminaria: -		Liczba punktów 3
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) inny		(ogólnouczelniany, z innego kierunku) z danego kierunku
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne nauki techniczne		Podział ECTS (liczba i %) 3 100% 3 100%
Odpowiedzialny za przedmiot / wykładowca: dr inż. Sławomir Maćkowiak email: smack@put.poznan.pl tel. +48 0616653890 Wydział Elektroniki i Telekomunikacji ul. Piotrowo 3A 60-965 Poznań		
Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:		
1	Wiedza:	1. Posiada wiedzę w zakresie programowania w językach C/C++. 2. Posiada wiedzę podstawową z zakresu przetwarzania obrazu.
2	Umiejętności:	1. Potrafi pozyskiwać informacje z literatury i baz danych oraz innych źródeł w języku polskim lub angielskim. 2. Potrafi się posługiwać językami programowania wysokiego poziomu C/C++.
3	Kompetencje społeczne	Zdolny do samodzielnego uczenia się (podręczniki, programy komputerowe) Zachowuje się aktywnie na zajęciach, stawia pytania, świadomie korzysta z kontaktów z prowadzącym (np. w ramach konsultacji).
Cel przedmiotu: Przedmiot wychodzi naprzeciw najnowszym trendom programowania aplikacji dla zastosowań multimedialnych, wykorzystania nowoczesnych technologii multimedialnych w systemach telekomunikacyjnych. Różne języki programowania pozwalają na wykorzystanie różnych stylów programowania zwanych również paradygmatem programowania oraz specyficznych cech danego języka. Wybór konkretnego narzędzia, biblioteki programistycznej może zależeć od indywidualnych upodobań, polityki firmy tworzącej oprogramowanie lub ze względu na zadanie, jakie końcowa aplikacja ma realizować. Najlepszym rozwiązaniem jest wybór języka programowania najbardziej dostosowanego do rozwiązywanego zadania i ewentualnej istniejącej infrastruktury. Najważniejsze kryteria wyboru języka programowania to: paradygmat i rodzaj języka, przenośność i wydajność kompilatorów, dojrzałość oraz dostępność narzędzi i dokumentacji. Ze względu na dostępność kompilatorów dla danej platformy sprzętowej, wynikowy kod aplikacji, czy konieczność utrzymania i integracji z już istniejącą infrastrukturą wybór odpowiednich narzędzi i technik programistycznych ma duże znaczenie. W przypadku większych projektów istotne jest też, jak szybko można znaleźć odpowiednio wykwalifikowane osoby i zespół.		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza:		
1. Ma praktyczną wiedzę na temat dostępnych narzędzi programistycznych, bibliotek specjalizowanych wspomagających pisanie oprogramowanie dla zastosowań multimedialnych. - [K1_W11]		
2. Jest zapoznany z algorytmami tworzenia wrażenia ruchu obiektów, modelowania cieni, światła, przesłaniania się obiektów graficznych. - [K1_W11]		
3. Posiada wiedzę w zakresie tworzenia oprogramowania dedykowanego różnym specjalistycznym platformom sprzętowym. - [K1_W11]		
Umiejętności:		

1. Posiada umiejętność wykorzystywania narzędzi programistycznych, bibliotek specjalizowanych dla zastosowań multimedialnych. - [K1_U14]
2. Potrafi tworzyć grafikę dwuwymiarową i trójwymiarową na ekranie monitora komputerowego, potrafi tworzyć wrażenie ruchu obiektów, uzyskiwać efekty cieni, światła, przesłaniania się obiektów graficznych. - [K1_U14]
3. Posiada umiejętność dostosowania kodu oprogramowania dla różnych platform sprzętowych i uzyskania tego samego efektu w różnych warunkach i przy różnych wymaganiach. - [K1_U14]
Kompetencje społeczne:
1. Jest otwarty na możliwości ciągłego dokształcania się i rozumie konieczność podnoszenia kompetencji zawodowych. - [K1_K01]
2. Ma poczucie odpowiedzialności za jakość powstającego oprogramowania. - [k1_k03]

Sposoby sprawdzenia efektów kształcenia		
1.	Egzaminy pisemny lub ustny lub pytania testowe.	
2.	Raporty (Sprawozdania) z jednolitych tematycznie bloków ćwiczeń laboratoryjnych.	
3.	Sprawdzanie przygotowania do zajęć i aktywności podczas ćwiczeń laboratoryjnych.	
Treści programowe		
<p>Windows Multimedia: Multimedia Audio, Multimedia Input, Video for Windows. MFC (Microsoft Foundation Classes): Obsługa plików wielu formatów. Serializacja własnych obiektów. Inteligentne drukowanie. Edycja w podglądzie wydruku. Dostosowywanie elementów kontrolnych. Niemodalne arkusze właściwości. Własne kreatory AppWizard. Własne procedury DDX i DDV. Rozszerzanie MFC za pomocą bibliotek DLL. ActiveX i programy internetowe. Bazy danych, wielowątkowość. DirectX: Tworzenie obiektu DirectDraw, tworzenie powierzchni DirectDraw. Kopiowanie zawartości powierzchni między innymi powierzchniami. Przelączanie buforów. Bezpośredni dostęp do zawartości powierzchni. Tworzenie i zarządzanie CLIPPER-em. Tworzenie palet kolorów. Animowanie palet. Direct3D: Komunikacja DirectX z urządzeniami graficznymi i systemem Windows. Podstawy dwu- i trójwymiarowej grafiki. Modelowanie, renderowanie, korzystania z macierzy. Operacje przeliczania i transformowania bitmap z użyciem DirectDraw. Wykorzystanie sprajtów, przelączanie buforów oraz inne techniki animacji w DirectDraw. Programowanie interfejsu Direct3D w trybie retained, w celu tworzenia trójwymiarowych animacji. Manipulowanie ramkami, siatkami, teksturami, mipmapami, światłami i cieniami. Ruch z wykorzystaniem morfingu i innych technik animacji. Direct Sound. Open GL: Wprowadzenie do OpenGL. OpenGL z użyciem biblioteki AUX. OpenGL for Windows: OpenGL + Win32 = Wiggly. Błędy i inne komunikaty OpenGL. Rysowanie w trzech wymiarach: linie, punkty i wielokąty. Manipulowanie przestrzenią 3D: transformacje współrzędnych. Kolory i cieniowanie. Oświetlenie i źródła światła. Modelowanie i kompozycja obiektów 3D. Grafika rastrowa. Mapowanie tekstur. Kwadryki: sfery, cylindry i dyski. Maszyna stanu OpenGL. Krzywe i powierzchnie. Mikroporocesyory ARM. Windows Mobile.</p>		
Literatura podstawowa:		
1. Programowanie: COM+ Guy Eddon, Henry Eddon, Wyd. RM (2001)		
2. Microsoft Windows - programowanie sieciowe Anthony Jones, Jim Ohlund, (2000)		
3. DirectX w przykładach Adam Ślosarski, Wyd. Mikom, (Warszawa 1999)		
4. Efektywne programowanie w C++. Dov Bulka, David Mayhew(przekład Jacek Mozdyniewicz), Wyd. Mikom, (Warszawa 2001)		
5. OpenGL - księga eksperta Richard S. Wright jr, Michael Sweet Tłumaczenie:Marcin Pancewicz, Wyd. Helion, (11/1999)		
Literatura uzupełniająca:		
1. Teach Yourself DirectX 7 in 24 hours Dunlop, Shepherd, Martin, SAMS, 2000		
Bilans nakładu pracy przeciętnego studenta		
Czynność	Czas (godz.)	
1. Zajęcia wymagających indywidualnego kontaktu z nauczycielem akademickim (wykład 15h, lab 30h)	45	
2. Przygotowania do laboratorium i opracowania raportu (sprawozdania)	15	
3. Czytania literatury (podręczniki, katalogi)	10	
4. Przygotowanie do zaliczenia	15	
5. Konsultacje z wykładów i projektu z laboratorium	3	
6. Udział w zaliczeniu	2	
Obciążenie pracą studenta		
forma aktywności	godzin	ECTS
Łączny nakład pracy	90	3
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	50	2

Zajęcia o charakterze praktycznym	45	1
-----------------------------------	----	---