

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Inżynieria oprogramowanie dla multimedków		Kod 1010841161010843668
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. W ramach przedmiotu materiał obejmuje zagadnienia związane z budowaniem i zarządzaniem projektem z zakresu informatyki i inżynierii oprogramowania. W ramach tych zagadnień poznawane są modele cyklu życia oprogramowania, czynności wykonywane w poszczególnych fazach przedsięwzięcia programistycznego, wykorzystanie narzędzi CASE, właściwą identyfikację i definicję wymagań oraz opis działań związanych z zapewnieniem odpowiedniej jakości powstającego oprogramowania. Omawiane są najbardziej popularne metody analizy i projektowania oprogramowania.		
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. Posiada wiedzę w zakresie tworzenia oprogramowania dedykowanego różnym specjalistycznym platformom sprzętowym. - [K1_W11]		
Umiejętności:		

<p>1. Posiada umiejętność wykorzystywania narzędzi programistycznych, bibliotek specjalizowanych dla zastosowań multimedialnych. - [K1_U14]</p> <p>2. Potrafi wykorzystywać narzędzia do zarządzania wersjami oprogramowania. Potrafi testować oprogramowanie, wykorzystywać narzędzia do debuggowania napisanego oprogramowania. - [K1_U14]</p> <p>3. Posiada umiejętność dostosowania kodu oprogramowania dla różnych platform sprzętowych i uzyskania tego samego efektu w różnych warunkach i przy różnych wymaganiach. - [K1_U14]</p>
<p>Kompetencje społeczne:</p>
<p>1. Jest otwarty na możliwości ciągłego dokształcania się i rozumie konieczność podnoszenia kompetencji zawodowych. - [K1_K01]</p> <p>2. Ma poczucie odpowiedzialności za jakość powstającego oprogramowania. - [k1_k03]</p>

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. Mikroprocesory ARM. Windows Mobile.</p> <p>COM+ Component Object Model -podstawowa architektura programistyczna COM+. Działanie podstawowych mechanizmów COM+ w ramach Windows 2000 i architektury Windows Distributed InterNet Applications (DNA). Użycie Interface Definition Language (IDL) do tworzenia bibliotek typów. Tworzenie klientów i komponentów w językach C++. Wątki i przedziały.</p> <p>C# Przegląd języka C#, środowiska CLR (Common Language Runtime) oraz BCL (Base Class Library), omówienie elementów języka C#, główne typy danych C#, typy bazowe, obiekty, wyliczenia, delegacje, kolekcje, interfejsy. Obsługa wyjątków i zdarzeń. Wątki, i korzystanie z wyrażeń regularnych. Refleksje i własne atrybuty. Współpraca ze starszymi komponentami API Win32 oraz COM.</p> <p>Narzędzia zarządzania złożonymi przedsięwzięciami programistycznymi (UML), systemy zarządzania wersjami oprogramowania, niezawodność oprogramowania. Budowa i zarządzanie projektem informatycznym. Użycie UML zgodnie z zasadami podejścia zorientowanego obiektowo. Narzędzia zarządzania wersjami Microsoft Visual SourceSafe oraz CVS. Organizowanie systemu zarządzania kodem źródłowym. Niezawodność oprogramowania. Wykorzystanie debuggerów.</p> <p>Materiał obejmuje zagadnienia związane z budowaniem i zarządzaniem projektem z zakresu informatyki i inżynierii oprogramowania. W ramach tych zagadnień poznawane są modele cyklu życia oprogramowania, czynności wykonywane w poszczególnych fazach przedsięwzięcia programistycznego, wykorzystanie narzędzi CASE, właściwą identyfikację i definicję wymagań oraz opis działań związanych z zapewnieniem odpowiedniej jakości powstającego oprogramowania. Omawiane są najbardziej popularne metody analizy i projektowania oprogramowania ze szczególnym uwzględnieniem metod obiektowych. Poruszane zagadnienia zilustrowane są kilkoma przykładami.</p>	

<p>Literatura podstawowa:</p> <p>1. Programowanie: COM+ Guy Eddon, Henry Eddon, Wyd. RM (2001)</p> <p>2. Microsoft Windows - programowanie sieciowe Anthony Jones, Jim Ohlund, (2000)</p> <p>3. DirectX w przykładach Adam Ślosarski, Wyd. Mikom, (Warszawa 1999)</p> <p>4. Efektywne programowanie w C++. Dov Bulka, David Mayhew(przekład Jacek Mozdyniewicz), Wyd. Mikom, (Warszawa 2001)</p> <p>5. OpenGL - księga eksperta Richard S. Wright jr, Michael Sweet Tłumaczenie:Marcin Pancewicz, Wyd. Helion, (11/1999)</p>
--

<p>Literatura uzupełniająca:</p> <p>1. Teach Yourself DirectX 7 in 24 hours Dunlop, Shepherd, Martin, SAMS, 2000</p>

Bilans nakładu pracy przeciętnego studenta	
Czynność	Czas (godz.)
1. Zajęcia wymagających indywidualnego kontaktu z nauczycielem akademickim	45
2. Przygotowania do laboratorium i opracowania raportu (sprawozdania)	15
3. Czytania literatury (podręczniki, katalogi)	15
4. Przygotowanie do egzaminu	15
5. Konsultacje z wykładów i projektu z labotarotrium	15
Obciążenie pracą studenta	

Wydział Elektroniki i Telekomunikacji

forma aktywności	godzin	ECTS
Łączny nakład pracy	90	3
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	60	2
Zajęcia o charakterze praktycznym	30	1