

<b>KARTA OPISU MODUŁU KSZTAŁCENIA</b>		
Nazwa modułu/przedmiotu <b>Programowanie wieloparadygmatowe</b>		Kod <b>1010334491010337136</b>
Kierunek studiów <b>Informatyka</b>	Profil kształcenia (ogólnoakademicki, praktyczny) <b>(brak)</b>	Rok / Semestr <b>5 / 9</b>
Ścieżka obieralności/specjalność <b>Technologie informatyczne</b>	Przedmiot oferowany w języku: <b>polski</b>	Kurs (obligatoryjny/obieralny) <b>obligatoryjny</b>
Stopień studiów: <b>I stopień</b>	Forma studiów (stacjonarna/niestacjonarna) <b>niestacjonarna</b>	
Godziny Wykłady: <b>8</b> Ćwiczenia: <b>-</b> Laboratoria: <b>8</b> Projekty/seminaria: <b>-</b>		Liczba punktów <b>3</b>
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) <b>(brak)</b>		(ogólnouczelniany, z innego kierunku) <b>(brak)</b>
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki <b>nauki techniczne</b>		Podział ECTS (liczba i %) <b>3 100%</b>
<b>Odpowiedzialny za przedmiot / wykładowca:</b>		
dr inż. Grażyna Brzykcy email: Grażyna.Brzykcy@put.poznan.pl tel. 61 665 37 24 Wydział Elektryczny ul. Piotrowo 3A 60-965 Poznań		dr inż. Adam Meissner email: Adam.Meissner@put.poznan.pl tel. 61 665 37 24 Wydział Elektryczny ul. Piotrowo 3A 60-965 Poznań
<b>Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:</b>		
1	<b>Wiedza:</b>	Student ma podstawowe wiadomości z zakresu logiki matematycznej, teorii funkcji rekurencyjnych, programowania imperatywnego i deklaratywnego, programowania zorientowanego obiektowo, baz danych, systemów operacyjnych oraz sieci komputerowych.
2	<b>Umiejętności:</b>	Student potrafi pozyskiwać informacje z literatury, baz danych i innych źródeł; potrafi integrować uzyskane informacje, dokonywać ich interpretacji, a także wyciągać wnioski oraz formułować opinie; posługuje się językiem angielskim w stopniu wystarczającym do porozumiewania się, a także do czytania ze zrozumieniem wskazanej literatury przedmiotowej.
3	<b>Kompetencje społeczne</b>	Student rozumie potrzebę i zna możliwości ciągłego doksztalcania się, podnoszenia kompetencji językowych, zawodowych, osobistych i społecznych; ma świadomość odpowiedzialności za pracę własną oraz gotowość podporządkowania się zasadom pracy w zespole i ponoszenia odpowiedzialności za wspólnie realizowane zadania.
<b>Cel przedmiotu:</b> przeгляд paradygmatów obliczeniowych wykorzystywanych w informatyce ze szczególnym uwzględnieniem podstawowych pojęć, technik i abstrakcji programowych; wypracowanie umiejętności doboru modelu obliczeniowego do rozwiązywanego problemu; nabycie umiejętności programowania w wieloparadygmatowym środowisku obliczeniowym.		
<b>Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia</b>		
<b>Wiedza:</b> 1. Student ma uporządkowaną i podbudowaną teoretycznie wiedzę w zakresie podst. konstrukcji programistycznych, implementacji algorytmów, paradygmatów i stylów programowania, metod weryfikacji poprawności programów, języków formalnych, kompilatorów, platform - [K_W05]		
<b>Umiejętności:</b> 1. Student potrafi posłużyć się środowiskami i platformami programistycznymi do pisania, wykonywania i testowania prostych programów kodowanych w językach programowania imperatywnego, obiektowego i deklaratywnego - [K_U10]		
<b>Kompetencje społeczne:</b> 1. Student ma świadomość ważności dokładnego wykonania projektu, zachowania standardów notacyjnych, przestrzegania poprawności językowej i terminowego oddania prac - [K_K07]		
<b>Sposoby sprawdzenia efektów kształcenia</b>		
Wykład. Pisemne sprawdzenie wiadomości obejmujące pytania teoretyczne oraz proste zadania. Laboratorium. Ustna lub pisemna ocena przygotowania studentów do zajęć. Ocena aktywności studentów w zakresie realizacji bieżących ćwiczeń. Warunki zaliczenia wykładu i laboratorium: należy uzyskać co najmniej 50,1% łącznej liczby punktów.		

<b>Treści programowe</b>		
<p>Wykład. Deklaratywny paradygmat obliczeniowy. Charakterystyczne pojęcia i techniki programowania funkcyjnego i deterministycznego programowanie w logice. Programowanie iteracyjne, rekurencyjne, metaprogramowanie, abstrakcyjne typy danych. Deklaratywna równoległość, czyli sterowany danymi model obliczeń równoległych. Model równoległości ze współdzielonym stanem. Relacyjny paradygmat obliczeniowy i bazy danych. Programowanie rozproszone i programowanie z więzami jako elementy metodyki programowania wieloparadygmatowego.</p> <p>Laboratorium. Ćwiczenia obejmujące techniki programowania wieloparadygmatowego w języku Oz z wykorzystaniem środowiska Mozart.</p>		
<p><b>Literatura podstawowa:</b></p> <p>1. Van Roy P., Haridi S.: Programowanie. Koncepcje, techniki i modele, Helion, Gliwice 2005. 2. Mozart Consortium: The Mozart programming system, <a href="http://www.mozart-oz.org">http://www.mozart-oz.org</a>, 2006.</p>		
<p><b>Literatura uzupełniająca:</b></p> <p>1. Brzykcy G., Meissner. A.: Programowanie w Prologu i programowanie funkcyjne. Materiały do ćwiczeń, Wyd. PP., 1999. 2. Kowalski R.: Logika w rozwiązywaniu zadań, WNT, Warszawa, 1989.</p>		
<b>Bilans nakładu pracy przeciętnego studenta</b>		
Czynność	Czas (godz.)	
1. Wykłady	8	
2. Laboratoria	8	
3. Przygotowanie do zajęć	48	
<b>Obciążenie pracą studenta</b>		
forma aktywności	godzin	ECTS
Łączny nakład pracy	64	3
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	16	1
Zajęcia o charakterze praktycznym	48	2