

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Języki programowania		Kod 1010254451010320531
Kierunek studiów Mechatronika - studia niestacjonarne I stopnia	Profil kształcenia (ogólnoakademicki, praktyczny) (brak)	Rok / Semestr 3 / 5
Ścieżka obieralności/specjalność -	Przedmiot oferowany w języku: polski	Kurs (obligatoryjny/obieralny) obligatoryjny
Stopień studiów: I stopień	Forma studiów (stacjonarna/niestacjonarna) niestacjonarna	
Godziny Wykłady: 10 Ćwiczenia: - Laboratoria: 8 Projekty/seminaria: -		Liczba punktów 2
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) (brak)		(ogólnouczelniany, z innego kierunku) (brak)
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne nauki techniczne		Podział ECTS (liczba i %) 2 100% 2 100%
Odpowiedzialny za przedmiot / wykładowca:		
prof. dr hab. inż. W. Szelağ email: wojciech.szelağ@put.poznan.pl tel. 616652116 Wydział Elektryczny ul. Piotrowo 3, 60-965 Poznań tel.: 061 665 2388		
Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:		
1	Wiedza:	Podstawowa wiedza zakresie informatyki, sprzętu komputerowego, systemów operacyjnych, algorytmów i struktur danych
2	Umiejętności:	Obsługa: komputera, systemu operacyjnego Windows oraz oprogramowania użytkowego
3	Kompetencje społeczne	Rozumie potrzebę uczenia się
Cel przedmiotu:		
Przekazanie podstawowej wiedzy z zakresu języków programowania, opanowanie umiejętności opracowywania algorytmów i podstaw programowania strukturalnego w językach Pascal, Object Pascal, C/C++		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza:		
1. Znajomość podstawowych pojęć z zakresu języków programowania, algorytmów, struktur danych oraz opracowywania programów komputerowych - [K_W04] 2. Znajomość języków programowania Pascal, Object Pascal i C/C++ - [K_W04, 03] 3. Znajomość podstawowych technik programowania i wybranych środowisk programistycznych - [K_W04] 4. Znajomość metod tworzenia algorytmów i programów komputerowych do rozwiązywania wybranych zadań i problemów technicznych - [K_W04, 03]		
Umiejętności:		
1. Pozyskiwanie informacji z Internetu oraz literatury technicznej dotyczącej programowania i języków programowania - [K_U01,08] 2. Umiejętność formułowania i rozwiązywanie zadań oraz problemów technicznych przy wykorzystaniu inżynierskich metod analitycznych i symulacyjnych - [K_U14,28] 3. Umiejętność optymalizacji zadań oraz wybranych problemów inżynierskich i opracowywania odpowiadającym im programów komputerowych w językach Pascal, Object Pascal i C/C++ - [K_U30,28] 4. Umiejętność testowania opracowanych programów oraz oceny poprawności ich funkcjonowania - [K_U05]		
Kompetencje społeczne:		
1. Rozumie potrzebę uczenia się przez całe życie; potrafi inspirować i organizować proces uczenia się innych osób - [K_K01] 2. Potrafi współdziałać i pracować w grupie, przyjmuje w niej różne role - [K_K03] 3. Potrafi określić priorytety służące realizacji określonego zadania - [K_K04]		

Sposoby sprawdzenia efektów kształcenia		
<p>Wykład:</p> <p>? Zaliczenie na podstawie kolokwium składającego się z pytań ogólnych i testowych. Skala ocen 51-60% pkt. ? dst, 61-70% pkt dst+, 71-80% pkt. ? db, 81-90% pkt. ? db+, 91-100% pkt. ? bdb.</p> <p>Laboratorium:</p> <p>? premiowanie praktycznej wiedzy zdobytej w trakcie poprzednich ćwiczeń laboratoryjnych,</p> <p>? praktyczne sprawdzenie umiejętności programowania w języku C++, Pascal i Object Pascal (kolokwium zaliczeniowe),</p> <p>? ocena wiedzy i umiejętności związanych z realizacją indywidualnych i grupowych projektów programistycznych</p> <p>Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:</p> <p>? umiejętność współpracy w ramach zespołu praktycznie realizującego zadanie szczegółowe w laboratorium,</p> <p>? wykorzystanie elementów i technik wykraczających poza materiał z zakresu prowadzonego wykładu i ćwiczeń laboratoryjnych,</p> <p>? staranność estetyczną zrealizowanych projektów.</p>		
Treści programowe		
<p>Podział języków programowania. Kompilacja, interpretacja. Języki niskiego i wysokiego poziomu. Przegląd języków programowania: Asembler, Basic, Fortran, C, C++, C#, Pascal, JAVA, PHP. Środowiska programistyczne. Edytory i kompilatory programów źródłowych, Debugger. Programowanie strukturalne i obiektowe. Podstawy programowania w języku Pascal i Object Pascal, C/C++.</p>		
<p>Literatura podstawowa:</p> <ol style="list-style-type: none"> 1. Jakubczyk K., Turbo Pascal i Borland C++. Przykłady, Wyd. Helion, Gliwice 2002 2. Hyde R.: Asembler. Sztuka programowania Helion, 2004. 3. Boduch A.: Delphi 7 - Kompendium programisty, Helion, 2003. 4. Grębosz J.: Symfonia C++ Standard, Wyd. Edition 2007. 		
<p>Literatura uzupełniająca:</p> <ol style="list-style-type: none"> 1. Boduch A. Wstęp do programowania w języku C#, Wyd. Helion, Gliwice 2006. 2. Stasiewicz A.: C++, Ćwiczenia praktyczne. Wydanie II, Helion 2006. 		
Bilans nakładu pracy przeciętnego studenta		
Czynność	Czas (godz.)	
Obciążenie pracą studenta		
forma aktywności	godzin	ECTS
Łączny nakład pracy	60	2
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	5	0
Zajęcia o charakterze praktycznym	15	0