

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Information engineering		Kod 1010331111010338982
Kierunek studiów Automatic Control and Robotics	Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki	Rok / Semestr 1 / 1
Ścieżka obieralności/specjalność Automatic Control and Robotics	Przedmiot oferowany w języku: polski	Kurs (obligatoryjny/obieralny) obligatoryjny
Stopień studiów: I stopień	Forma studiów (stacjonarna/niestacjonarna) stacjonarna	
Godziny Wykłady: 60 Ćwiczenia: - Laboratoria: 30 Projekty/seminaria: -		Liczba punktów 8
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) podstawowy		(ogólnouczelniany, z innego kierunku) ogólnouczelniany
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne nauki techniczne		Podział ECTS (liczba i %) 8 100% 8 100%
Odpowiedzialny za przedmiot / wykładowca: dr inż. Piotr Kaczmarek email: piotr.kaczmarek@put.poznan.pl tel. +48616652886 Wydział Elektryczny ul. Piotrowo 3A 60-965 Poznań		
Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:		
1	Wiedza:	Ma podstawową wiedzę wynikającą z programu szkoły średniej w zakresie matematyki, informatyki i logiki
2	Umiejętności:	Potrafi pozyskiwać informacje z literatury, baz danych i innych źródeł; posiada umiejętności samokształcenia w celu podnoszenia i aktualizacji kompetencji zawodowych. Posługuje się językiem angielskim na poziomie B2 wystarczającym do porozumiewania się, a także czytania ze zrozumieniem kart katalogowych, not aplikacyjnych, instrukcji obsługi urządzeń oraz opisów narzędzi informatycznych.
3	Kompetencje społeczne	Rozumie potrzebę i zna możliwości ciągłego doksztalcania się ? podnoszenia kompetencji zawodowych, osobistych i społecznych, potrafi inspirować i organizować proces uczenia się innych osób.
Cel przedmiotu: Celem przedmiotu jest nauka programowania proceduralnego i obiektowego w języku C i C++, zapoznanie z podstawowymi bibliotekami i narzędziami wspomagającymi programowanie komputerów PC. Podstawy teoretyczne wspomagane są zajęciami laboratoryjnymi. Ponadto wykład obejmuje zapoznanie studentów z architekturą komputerów PC, sieci komputerowych i współczesnych interfejsów komunikacyjnych i trendach w rozwoju systemów informatycznych		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza:		
1. Ma uporządkowaną wiedzę w zakresie wybranych algorytmów i struktur danych oraz metodyki i technik programowania proceduralnego i obiektowego - [K_W10]		
2. Ma uporządkowaną wiedzę w zakresie architektur komputerów, systemów i sieci komputerowych oraz systemów operacyjnych. - [K_W13]		
Umiejętności:		
1. Potrafi skonstruować algorytm rozwiązania prostego zadania inżynierskiego oraz zaimplementować, przetestować i uruchomić go w wybranym środowisku programistycznym na komputerze klasy PC dla wybranych systemów operacyjnych. - [K_U10]		
2. Potrafi pracować indywidualnie i w zespole; umie oszacować czas potrzebny na realizację zleconego zadania; potrafi opracować i zrealizować harmonogram prac zapewniający dotrzymanie terminów - [K_U02]		
Kompetencje społeczne:		

1. Posiada świadomość ważności i rozumie pozatechniczne aspekty i skutki działalności inżynierskiej w tym jej wpływ na środowisko i związaną z tym odpowiedzialność za podejmowane decyzje. - [K_K02]

Sposoby sprawdzenia efektów kształcenia

Wykład: egzamin w formie pisemnej z zakresu zasad programowania proceduralnego i obiektowego, architektury komputerów PC i interfejsów komunikacyjnych

Laboratorium: sprawdzenie praktycznych umiejętności z zakresu programowania proceduralnego i obiektowego w języku C i C++, oceny z kolokwium, pracy na zajęciach oraz zadań domowych

Treści programowe

Wykład: Systemy liczbowe, podstawowe typy danych, pętle i instrukcje warunkowe, funkcje, wskaźniki, struktury i dynamiczne typy danych, obsługa plików, podstawy algorytmów (sortowanie, metody rekurencyjne i iteracyjne), programowanie obiektowe, polimorfizm, dziedziczenie, OpenGL, programowanie aplikacji sieciowych klient-server, tworzenie aplikacji okienkowych, architektura procesorów, współczesne trendy rozwoju procesorów oraz techniki zwiększania wydajności obliczeniowej, metody przechowywania danych, sieci komputerowe i interfejsy komunikacyjne (ethernet, usb, rs232, rs485, firewire, bluetooth), metody realizacji warstwy fizycznej w sieciach komputerowych i interfejsach komunikacyjnych (sieci bezprzewodowe, przewodowe, światłowodowe), karty graficzne i metody przetwarzania równoległego

Laboratorium: programowanie w języku C i C++, obsługa i formatowanie wejścia/wyjścia, nauka stosowania pętli i instrukcji warunkowych, organizowania kodu programu przy pomocy funkcji. Wykorzystanie tablic, wskaźników i dynamicznych struktur danych (listy jedno i dwukierunkowe). Tworzenie i projektowanie prostych obiektów, zastosowanie dziedziczenia i polimorfizmu, wykorzystanie operatorów, wykorzystanie bibliotek wspomagających programowanie (OpenGL, STL, windows sockets)

Aktualizacja 2017: C++14, VisualStudio 2017, nowe przykłady

Literatura podstawowa:

1. Bruce Eckel, Thinking in C++, Volume 2: Practical Programming
2. Bjarne Stroustrup, Programming: Principles and Practice Using C++ (2nd Edition)
3. Irv Englander, The Architecture of Computer Hardware, Systems Software, and Networking: An Information Technology Approach
4. e-learning platform: moodle.put.poznan.pl

Literatura uzupełniająca:

Bilans nakładu pracy przeciętnego studenta

Czynność	Czas (godz.)
1. Wykład	60
2. Laboratorium	30
3. Przygotowanie do egzaminu/zaliczenie wykładu	35
4. Przygotowanie do ćwiczeń i wykonanie sprawozdań	60
5. Egzamin i konsultacje	5

Obciążenie pracą studenta

forma aktywności	godzin	ECTS
Łączny nakład pracy	190	8
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	95	4
Zajęcia o charakterze praktycznym	95	4