

Programming Languages - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Programming Languages
Kod przedmiotu	06.9-WM-ER-IB-40_18
Wydział	Wydział Mechaniczny
Kierunek	WM - oferta ERASMUS
Profil	-
Rodzaj studiów	Program Erasmus
Semestr rozpoczęcia	semestr zimowy 2018/2019

Informacje o przedmiocie	
Semestr	2
Liczba punktów ECTS do zdobycia	5
Typ przedmiotu	obowiązkowy
Język nauczania	angielski
Sylabus opracował	dr hab. inż. Katarzyna Arkusz, prof. UZ

Formy zajęć					
Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Wykład	30	2	-	-	Zaliczenie na ocenę
Laboratorium	30	2	-	-	Zaliczenie na ocenę

Cel przedmiotu

The aim is to acquire the skills and competencies of the structured programming in C language and the basics of programming in C ++

Wymagania wstępne

Basic knowledge of information technology.

Zakres tematyczny

Designing the program. Structured Programming. The algorithms and data structures and their representations in a programming language. Programming in C. The structure of the program, the command syntax. Fixed and variable data types. Operators, expressions. Type conversions. Arithmetic operators and their hierarchy. Instructions inputs and outputs. Conditional statements. Instructions iterative loops: for, while, for. Features: structure, arguments, result, prototype declaration calling. The formal parameters and actual functions. The concept and properties of the stack. Passing parameters by value and address. Returning values from functions. Recursive functions.

Indicators: declaration, initialization, and a reference to the address indicated value. Solid indicators and indices for fixed: properties and application range. Pointers to functions: examples of applications. Formal parameters of the function which is a pointer to a function.

Boards. The declaration, applicable examples. String as an array of characters. Arrays vs pointers. Multi-dimensional arrays. Data structures. Properties. Arrays of structures. Fields. Lift.

Introduction to object-oriented programming. The concept of class as an abstract data type, storage methods, encapsulation. Basics of inheritance. Polymorphism as a mechanism to support object-oriented programming.

Metody kształcenia

Lecture: Lecture in the form of a multimedia presentation

Laboratory: Exercises and calculations

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Has ordered knowledge of the methods and techniques of programming.		zaliczenie - ustne, opisowe, testowe i inne	Wykład
Able to formulating and solving tasks related to biomedical engineering, to see the system aspects, economic, legal and social with the use of computer technology.		- aktywność w trakcie zajęć - wykonanie sprawozdań laboratoryjnych	Laboratorium

Warunki zaliczenia

Lecture - provided credit is to get a positive evaluation of the test.

Laboratory - provided credit is to pass all the laboratory.

The final grade received by the student is the arithmetic mean of the above grades.

Literatura podstawowa

1. Loudon K. Algorytmy w C. Helion 2003.
2. Kerighan, R. Programowanie w języku C. WNT 2000.
3. Kisilewicz J. Język. w środowisku Borland C++. Wydanie IV. Oficyna Wydawnicza Politechniki Wrocławskiej. Wrocław 2003.
4. Stroustrup B. C++ Język programowania. WNT 2001

Literatura uzupełniająca

5. Lippman S.B. *Model w C++*, WNT, Warszawa, 1996.
6. Eckel B. *Thinking in C++*, Hellion, Warszawa, 2002.

Uwagi

Zmodyfikowane przez dr hab. inż. Katarzyna Arkusz, prof. UZ (ostatnia modyfikacja: 19-04-2018 12:03)

Wygenerowano automatycznie z systemu SylabUZ