Principles of programming - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Principles of programming
Kod przedmiotu	11.3-WE-ELEKTP-PrinProgr-Er
Wydział	Wydział Nauk Inżynieryjno-Technicznych
Kierunek	Elektrotechnika
Profil	ogólnoakademicki
Rodzaj studiów	Program Erasmus pierwszego stopnia
Semestr rozpoczęcia	semestr zimowy 2022/2023

Informacje o przedmiocie

Semestr	3
Liczba punktów ECTS do zdobycia	5
Typ przedmiotu	obowiązkowy
Język nauczania	angielski
Sylabus opracował •	dr hab. inż. Paweł Majdzik, 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

To provide basic knowledge about computer system architecture and programming.

To provide basic knowledge about C program structure and design.

To give basic skills in using C commands and functions to solve programming problems.

Wymagania wstępne

Zakres tematyczny

Computer system structure. Operating system. Program structure and design. Programming languages. Algorithmic languages. C programming. Program structure, commands syntax, identifiers, types, constans, declarations of data.

Arithmetic Operations, Relational and Logical Operations, Bitwise Operators, Assignment Operators, Type Conversions

Instructions: expressional instruction, empty instruction, grouping instruction.

Control instructions: if-else, for loop, switch, while loop. Printout formatting with printf function. Flag, field width, precision, formatting character. Complex instructions, expressional instruction, grouping instruction. Control instructions: if-else, switch. Loops: do, while, for. Functions: prototypes, declaration, definition, benefits for functions, arguments, result, calling out, use of functions, recurrence functions.

Scopes of names: local scope, external scope, modular programming.

Pointers: pointers syntax, declaration, operators, using the address and the pointed value. Use of

pointers to communicate with other elements.

Arrays : declaration, usage, pointers and arrays, strings. Data structures. Features, operation. Arrays of structures. Fields. Unions.

Dynamic memory: standard memory allocation functions, dynamic memory management

Data structures: lists, stacks, binary trees, circular buffers.

Metody kształcenia

Lecture, laboratory exercises.

Efekty uczenia się i metody weryfikacji osiągania efektów uczenia się

Opis efektu

Symbole efektów Metody weryfikacji

Forma zajęć

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Can realize a programming project individually, if necessary with additional self-		 kolokwium 	 Wykład
studying.		• sprawdzian	Laboratorium
Knows and can practically apply principles of C language software design and		• kolokwium	• Wykład
analyze an example program		• sprawdzian	• Laboratorium
Knows and can solve examples of software tasks working individually or in a team		• kolokwium	• Wykład
		• sprawdzian	• Laboratorium

Warunki zaliczenia

Lecture – the passing condition is to obtain a positive mark from the final test. Laboratory – the passing condition is to obtain positive marks from all laboratory exercises to be planned during the semester. Calculation of the final grade: lecture 50% + laboratory 50%

Literatura podstawowa

- 1. Język ANSI C. Programowanie. Wydanie II, Brian W. Kernighan, Dennis M. Ritchie, Wydawnictwo Helion, Gliwice 2010
- 2. Summit S. Programowanie w języku C, Helion, 2003,
- 3. Kisilewicz J. Język C w środowisku Borland C++, Wydanie IV, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 2003
- 4. Stephen Prata, Języ.k C. Szkoła programowania, Robomatic, Wrocław 2001

Literatura uzupełniająca

Uwagi

Zmodyfikowane przez dr hab. inż. Paweł Szcześniak, prof. UZ (ostatnia modyfikacja: 06-04-2022 22:42)

Wygenerowano automatycznie z systemu SylabUZ