

Wydział Mechaniczny PWR

KARTA PRZEDMIOTU

Nazwa w języku polskim: **Podstawy informatyki**

Nazwa w języku angielskim: **Foundations of Computer Science**

Kierunek studiów (jeśli dotyczy): **Biomechanika Inżynierska**

Poziom i forma studiów: **I stopień, stacjonarne**

Rodzaj przedmiotu: **wybieralny**

Kod przedmiotu: **BIM031101**

Grupa kursów: **nie**

	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium
Liczba godzin zajęć zorganizowanych w Uczelni (ZZU)	15		30		
Liczba godzin całkowitego nakładu pracy studenta (CNPS)	30		60		
Forma zaliczenia	Zaliczenie na ocenę		Zaliczenie na ocenę		
Grupa kursów					
Liczba punktów ECTS	1		2		
w tym liczba punktów odpowiadająca zajęciom o charakterze praktycznym (P)			2		
w tym liczba punktów ECTS odpowiadająca zajęciom wymagającym bezpośredniego kontaktu (BK)	0.6		1.4		

WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I KOMPETENCJI SPOŁECZNYCH

1. Podstawowa wiedza z zakresu analizy matematycznej i logiki

CELE PRZEDMIOTU

C1. Wprowadzenie do algorytmiki

C2. Nauka podstaw nieobiekowego języka programowania C

C3. Nauka i ćwiczenie implementacji algorytmów w języku C

PRZEDMIOTOWE EFEKTY UCZENIA SIĘ

I. Z zakresu wiedzy:

PEK_W01 - Student posiada wiedzę niezbędną do formułowania skutecznych algorytmów

PEK_W02 - Student posiada wiedzę dotyczącą składni nieobiekowego języka programowania C

II. Z zakresu umiejętności:

PEK_U01 - Student potrafi tworzyć algorytmy i zapisywać je w formie schematów blokowych

PEK_U02 - Student potrafi implementować algorytmy w nieobiekowym języku programowania C

III. Z zakresu kompetencji społecznych:

PEK_K01 - Student potrafi pracować samodzielnie i w zespole

TREŚCI PROGRAMOWE

Forma zajęć – Wykład		Liczba godzin
Wy1	Podstawowe pojęcia z zakresu informatyki i algorytmiki	2
Wy2	Wprowadzenie do algorytmiki. Schematy blokowe algorytmów	2
Wy3	Przykłady algorytmów, tworzenie schematów blokowych	2
Wy4	Język C programowania. Podstawowe zasady kodowania. Typy zmiennych. Operatory arytmetyczne, logiczne, bitowe. Konstrukcje warunkowe w języku C	2
Wy5	Pętle, tablice i funkcje w języku C,	2
Wy6	Wskaźniki w języku C. Rekurencja. Przykłady implementacji algorytmów	2
Wy7	Struktury danych w języku C	2
Wy8	Kolokwium	1
		Suma: 15
Forma zajęć – Laboratorium		Liczba godzin
Lab1	Tworzenie schematów blokowych algorytmów - ćwiczenia	2
Lab2	Ćwiczenie w programowaniu w języku C z wykorzystaniem instrukcji warunkowych i operatorów logicznych	2
Lab3	Ćwiczenie w programowaniu w języku C z wykorzystaniem pętli	2
Lab4	Ćwiczenie w programowaniu w języku C z wykorzystaniem funkcji	2
Lab5	Ćwiczenie w programowaniu w języku C z wykorzystaniem wskaźników. Zastosowanie odwołania do zmiennej przez referencję.	2
Lab6	Ćwiczenie w programowaniu w języku C z wykorzystaniem struktur danych	2
Lab7	Implementacja list jednokierunkowych w języku C	2
Lab8	Implementacja list dwukierunkowych w języku C	2
Lab9	Implementacja drzew w języku C	2
Lab10	Implementacja wybranych algorytmów w języku C	2

Lab11	Samodzielny projekt	8
Lab12	Zaliczenie	2
		Suma: 30

STOSOWANE NARZĘDZIA DYDAKTYCZNE

N1. wykład tradycyjny z wykorzystaniem transparencji i slajdów
N2. ćwiczenia problemowe
N3. prezentacja projektu

OCENA OSIĄGNIĘCIA PRZEDMIOTOWYCH EFEKTÓW UCZENIA SIĘ (Wykład)

Oceny (F – formująca (w trakcie semestru), P – podsumowująca (na koniec semestru))	Numer efektu uczenia się	Sposób oceny osiągnięcia efektu uczenia się
F1	PEK_W01, PEK_W02	kolokwium
P = F1		

OCENA OSIĄGNIĘCIA PRZEDMIOTOWYCH EFEKTÓW UCZENIA SIĘ (Laboratorium)

Oceny (F – formująca (w trakcie semestru), P – podsumowująca (na koniec semestru))	Numer efektu uczenia się	Sposób oceny osiągnięcia efektu uczenia się
F1	PEK_U01, PEK_U02	ocena projektu własnego
P = F1		

LITERATURA PODSTAWOWA I UZUPEŁNIAJĄCA

LITERATURA PODSTAWOWA

1. J.Grębosz: Symfonia C++ standard, Edition 2000
2. Wirth Niklaus: Algorithms + Data Structures = Programs. Prentice-Hall (1976)

LITERATURA UZUPEŁNIAJĄCA

OPIEKUN PRZEDMIOTU

dr inż. Ewelina Świątek-Najwer tel.: 71 320-21-93 email: ewelina.swiatek@pwr.edu.pl

SUBJECT CARD

Name in Polish: **Podstawy informatyki**

Name in English: **Foundations of Computer Science**

Main field of study (if applicable): **Engineering Biomechanics**

Level and form of studies: **I level, full-time**

Kind of subject: **optional**

Subject code: **BIM031101**

Group of courses: **no**

	Lecture	Classes	Laboratory	Project	Seminar
Number of hours of organized classes in University (ZZU)	15		30		
Number of hours of total student workload (CNPS)	30		60		
Form of crediting	Crediting with grade		Crediting with grade		
Group of courses					
Number of ECTS points	1		2		
including number of ECTS points for practical (P) classes			2		
including number of ECTS points for direct teacher-student contact (BK) classes	0.6		1.4		

PREREQUISITES RELATING TO KNOWLEDGE, SKILLS AND OTHER COMPETENCES

1. Basic knowledge on mathematical analysis and logics

SUBJECT OBJECTIVES

- C1. Introduction to algorithms
- C2. Teaching of basics on non-object-oriented language of programming C
- C3. Teaching and exercises in implementing algorithms in C.

SUBJECT LEARNING OUTCOMES

I. Relating to knowledge:

PEK_W01 - Student has knowledge necessary to create efficient algorithms

PEK_W02 - Student has knowledge of syntax of non-object-oriented C programming language

II. Relating to skills:

PEK_U01 - Student can create algorithms and describe them in a form of block diagrams

PEK_U02 - Student can implement algorithms in non-object-oriented C programming language

III. Relating to social competences:

PEK_K01 - Student can work independently and in a team

PROGRAM CONTENT

Form of classes – Lecture		Number of hours
Lec1	Basic concepts in informatics and algorithmics	2
Lec2	Introduction to algorithmics. Block diagrams of algorithms	2
Lec3	Examples of algorithms, creation of block diagrams for algorithms	2
Lec4	C language. Basic rules of coding in C. Types of variables. Arithmetic, logical, bit operators. Conditional statements in C language	2
Lec5	Loops, tables and methods in C	2
Lec6	Pointers in C language. Recursion. Examples of algorithms implementations.	2
Lec7	Structures in C.	2
Lec8	Test	1
		Total hours: 15
Form of classes – Laboratory		Number of hours
Lab1	Creation of algorithm block diagrams - exercises	2
Lab2	Exercises in programming in C language using conditional instructions and logical operators	2
Lab3	Exercises in programming in C language using loops	2
Lab4	Exercises in programming in C language using functions	2
Lab5	Exercises in programming in C language using pointers. Application of passing of function argument by reference	2
Lab6	Exercises in programming in C language using data structures	2
Lab7	Implementation of unidirectional list in C language	2
Lab8	Implementation of bidirectional list in C language	2
Lab9	Implementation of tree structures in C language	2

Lab10	Implementation of selected algorithms in C language	2
Lab11	Self-project	8
Lab12	Credit for the course	2
		Total hours: 30

TEACHING TOOLS USED		
N1. traditional lecture with the use of transparencies and slides N2. problem exercises N3. project presentation		

EVALUATION OF SUBJECT LEARNING OUTCOMES ACHIEVEMENT (Lecture)		
Evaluation (F – forming (during semester), P – concluding (at semester end))	Learning outcomes number	Way of evaluating learning outcomes achievement
F1	PEK_W01, PEK_W02	test
P = F1		

EVALUATION OF SUBJECT LEARNING OUTCOMES ACHIEVEMENT (Laboratory)		
Evaluation (F – forming (during semester), P – concluding (at semester end))	Learning outcomes number	Way of evaluating learning outcomes achievement
F1	PEK_U01, PEK_U02	note for self-project
P = F1		

PRIMARY AND SECONDARY LITERATURE		
<u>PRIMARY LITERATURE</u> 1. J.Grębosz: Symfonia C++ standard, Edition 2000. 2. Wirth Niklaus: Algorithms + Data Structures = Programs. Prentice-Hall (1976)		
<u>SECONDARY LITERATURE</u>		

SUBJECT SUPERVISOR

dr inż. Ewelina Świątek-Najwer tel.: 71 320-21-93 email: ewelina.swiatek@pwr.edu.pl