

Wydział Mechaniczny PWR

## KARTA PRZEDMIOTU

Nazwa w języku polskim: **Informatyka III**

Nazwa w języku angielskim: **Software Engineering III**

Kierunek studiów (jeśli dotyczy): **Robotyka i Automatyzacja Procesów**

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

Rodzaj przedmiotu: **obowiązkowy**

Kod przedmiotu: **RAM031049**

Grupa kursów: **nie**

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

### WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I KOMPETENCJI SPOŁECZNYCH

1. Zna semantykę i syntaktykę języka C
2. Potrafi pisać, testować i debugować programy pisane w języku C
3. Posiada wiedzę dotyczącą paradygmatu programowania obiektowego i zapisu UML

### CELE PRZEDMIOTU

C1. Nauczyć umiejętności stosowania obiektowego paradygmatu programowania do rozwiązywania praktycznych zadań

## PRZEDMIOTOWE EFEKTY UCZENIA SIĘ

### I. Z zakresu wiedzy:

### II. Z zakresu umiejętności:

PEK\_U01 - Potrafi zaimplementować w języku C++ program na podstawie zadanej specyfikacji oraz diagramów UML

PEK\_U02 - Potrafi zastosować obiektowy paradygmat programowania, tj. zamodelować w języku UML, a następnie zaimplementować w języku C++ program dla zadanego problemu

PEK\_U03 - Potrafi zastosować poprawny styl programowania, przetestować i zdebugować opracowany program oraz opracować dokumentację kodu

### III. Z zakresu kompetencji społecznych:

PEK\_K01 - Potrafi wyszukiwać oraz krytycznie analizować informacje

PEK\_K02 - Potrafi zaplanować zadanie programistyczne, określić priorytety działań oraz zrealizować zadanie zgodnie z wymaganiami

## TREŚCI PROGRAMOWE

Forma zajęć – Projekt		Liczba godzin
Proj1	Klasa, atrybut, metoda, konstruktor, destruktor, obiekt, przeciążanie metod	2
Proj2	Akcesory, konstruktor przeciążony, konstruktor kopiujący, operatory new i delete	2
Proj3	Dziedziczenie, funkcje zaprzyjaźnione	2
Proj4	Operatory, przeciążanie operatorów, wskaźnik this	2
Proj5	Strumienie, operacje na plikach	2
Proj6	Asocjacja, agregacja, kompozycja	2
Proj7	Polimorfizm	2
Proj8	Symulacja układu regulacji	2
Proj9	Gra MasterMind/Oczko/etc	2
Proj10	Szablony	2
Proj11	Wyjątki	2
Proj12	Modelowanie obiektowe z użyciem UML	2
Proj13	Implementacja zamodelowanego w UML programu	2
Proj14	Testowanie i debugowanie programu, dokumentowanie kodu	2
Proj15	Zaliczenie	2
		Suma: 30

## STOSOWANE NARZĘDZIA DYDAKTYCZNE

N1. praca własna - przygotowanie do projektu  
 N2. Praca własna - implementacja, testowanie i dokumentowanie programów  
 N3. Internetowa baza wiedzy

## OCENA OSIĄGNIĘCIA PRZEDMIOTOWYCH EFEKTÓW UCZENIA SIĘ (Projekt)

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_U03 PEK_K01 - PEK_K02	Odpowiedzi ustne, kartkówki, sprawozdania (program + dokumentacja UML)
P = F1		

## LITERATURA PODSTAWOWA I UZUPEŁNIAJĄCA

### LITERATURA PODSTAWOWA

J. Liberty, C++ dla każdego, Helion 2002  
 B. Stroustrup, Język C++, WNT 2002  
 J. Grębosz, Symfonia C++, Editions 2000 rok 2006  
 B. Eckel, Thinking in C++ Edycja polska, Helion 2002

### LITERATURA UZUPEŁNIAJĄCA

N. M. Josuttis, C++. Programowanie zorientowane obiektowo. Vademecum profesjonalisty, Helion 2003  
 M. Flenov, C++ Elementarz hakera, Helion 2005

## OPIEKUN PRZEDMIOTU

dr hab. inż. Jacek Reiner tel.: 29-81 email: jacek.reiner@pwr.edu.pl

Faculty of Mechanical Engineering

## SUBJECT CARD

Name in Polish: **Informatyka III**

Name in English: **Software Engineering III**

Main field of study (if applicable): **Robotics and Process Automation**

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

Kind of subject: **obligatory**

Subject code: **RAM031049**

Group of courses: **no**

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

## PREREQUISITES RELATING TO KNOWLEDGE, SKILLS AND OTHER COMPETENCES

1. He/she knows the semantics and syntax of C
2. He/she can write, test and debug programs written in C
3. Has knowledge of object-oriented programming paradigm and writing UML

## SUBJECT OBJECTIVES

C1. Learn the skills to apply object-oriented programming paradigm for solving practical tasks

## SUBJECT LEARNING OUTCOMES

### I. Relating to knowledge:

### II. Relating to skills:

PEK\_U01 - Is able to implement a C ++ program from a given specification and UML

PEK\_U02 - Is able to apply object-oriented programming paradigm, ie. Modeled in UML, then implement in C ++ program for a given problem

PEK\_U03 - Can apply the correct style of programming, test, and debug a program developed and compile the code

### III. Relating to social competences:

PEK\_K01 - Can search and critically analyze information

PEK\_K02 - Is able to plan programming task, prioritize and implement activities in accordance with the requirements of the job

## PROGRAM CONTENT

Form of classes – Project		Number of hours
Proj1	Class, attribute, method, constructor, destructor, object, overloading methods	2
Proj2	Accessors overloaded constructor, copy constructor, new and delete operators	2
Proj3	Inheritance, functions befriended	2
Proj4	Operators, operator overloading, this pointer	2
Proj5	Streams, File operations	2
Proj6	Association, aggregation, composition	2
Proj7	polymorphism	2
Proj8	Simulation of the control system	2
Proj9	MasterMind game / Pond / etc	2
Proj10	Templates	2
Proj11	Exceptions	2
Proj12	Object-oriented modeling with UML	2
Proj13	Implementation of modelled program	2
Proj14	Testing and debugging, source code documentation	2
Proj15	Assessment	2
		Total hours: 30

## TEACHING TOOLS USED

N1. self study - preparation for project class  
 N2. Self work - implementation, testing, documentation  
 N3. Online knowledge base

## EVALUATION OF SUBJECT LEARNING OUTCOMES ACHIEVEMENT (Project)

Evaluation (F – forming (during semester), P – concluding (at semester end))	Learning outcomes number	Way of evaluating learning outcomes achievement
F1	PEK_U01 - PEK_U03 PEK_K01 - PEK_K02	Verbal answers, quizzes, reports (source code + UML documentation)

P = F1

## PRIMARY AND SECONDARY LITERATURE

### PRIMARY LITERATURE

J. Liberty, C++ dla każdego, Helion 2002  
 B. Stroustrup, Język C++, WNT 2002  
 J. Grębosz, Symfonia C++, Editions 2000 rok 2006  
 B. Eckel, Thinking in C++ Edycja polska, Helion 2002

### SECONDARY LITERATURE

N. M. Josuttis, C++. Programowanie zorientowane obiektowo. Vademecum profesjonalisty, Helion 2003  
 M. Flenov, C++ Elementarz hakera, Helion 2005

## SUBJECT SUPERVISOR

dr hab. inż. Jacek Reiner tel.: 29-81 email: [jacek.reiner@pwr.edu.pl](mailto:jacek.reiner@pwr.edu.pl)