

## KARTA PRZEDMIOTU

Nazwa w języku polskim: **Programowanie w C++**

Nazwa w języku angielskim: **C++ Programming**

Kierunek studiów (jeśli dotyczy): **Mechatronika**

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

Rodzaj przedmiotu: **wybieralny**

Kod przedmiotu: **MCM034104**

Grupa kursów: **nie**

	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium
Liczba godzin zajęć zorganizowanych w Uczelni (ZZU)			30		
Liczba godzin całkowitego nakładu pracy studenta (CNPS)			90		
Forma zaliczenia			Zaliczenie na ocenę		
Grupa kursów					
Liczba punktów ECTS			3		
w tym liczba punktów odpowiadająca zajęciom o charakterze praktycznym (P)			3		
w tym liczba punktów ECTS odpowiadająca zajęciom wymagającym bezpośredniego kontaktu (BK)			2.1		

### 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

## TREŚCI PROGRAMOWE

Forma zajęć – Laboratorium		Liczba godzin
Lab1	Klasa, atrybut, metoda, konstruktor, destruktor, obiekt, przeciążanie metod	2
Lab2	Akcesory, konstruktor przeciążony, konstruktor kopiujący, operatory new i delete	2
Lab3	Dziedziczenie, funkcje zaprzyjaźnione	2
Lab4	Operatory, przeciążanie operatorów, wskaźnik this	2
Lab5	Strumienie, operacje na plikach	2
Lab6	Asocjacja, agregacja, kompozycja	2
Lab7	Polimorfizm	2
Lab8	Symulacja układu regulacji	2
Lab9	Implementacja gry MasterMind/Oczko/etc.	2
Lab10	Szablony	2
Lab11	Wyjątki	2
Lab12	Modelowanie obiektowe z użyciem UML	2
Lab13	Implementacja zamodelowanego w UML programu	2
Lab14	Testowanie i debugowanie programu, dokumentowanie kodu	2
Lab15	Zaliczenie	2
		Suma: 30

## STOSOWANE NARZĘDZIA DYDAKTYCZNE

N1. praca własna – przygotowanie do laboratorium  
 N2. Praca własna - pisanie, testowanie i dokumentowanie programów  
 N3. konsultacje  
 N4. Internetowa baza wiedzy

## 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_U03 PEK_K01	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

## SUBJECT CARD

Name in Polish: **Programowanie w C++**

Name in English: **C++ Programming**

Main field of study (if applicable): **Mechatronics**

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

Kind of subject: **optional**

Subject code: **MCM034104**

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)			90		
Form of crediting			Crediting with grade		
Group of courses					
Number of ECTS points			3		
including number of ECTS points for practical (P) classes			3		
including number of ECTS points for direct teacher-student contact (BK) classes			2.1		

### PREREQUISITES RELATING TO KNOWLEDGE, SKILLS AND OTHER COMPETENCES

1. He knows the semantics and syntax of C
2. He 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 - Able to implement a C ++ program from a given specification and UML

PEK\_U02 - 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

## PROGRAM CONTENT

Form of classes – Laboratory		Number of hours
Lab1	Class, attribute, method, constructor, destructor, object, overloading methods	2
Lab2	Accessors overloaded constructor, copy constructor, new and delete operators	2
Lab3	Inheritance, functions befriended	2
Lab4	Operators, operator overloading, this indicator	2
Lab5	Streams, File operations	2
Lab6	Association, aggregation, composition	2
Lab7	Polymorphism	2
Lab8	Simulation of the control system	2
Lab9	Implementation game MasterMind / Pond / etc.	2
Lab10	Templates	2
Lab11	Exceptions	2
Lab12	Object Modeling with UML	2
Lab13	Implementation of the program modeled in UML	2
Lab14	Testing and debugging the program, documenting code	2
Lab15	Final assesment	2
		Total hours: 30

### TEACHING TOOLS USED

- N1. self study - preparation for laboratory class
- N2. Own work - implementation, testing and documenting programs
- N3. tutorials
- N4. Online knowledge base

### 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_U03 PEK_K01	Oral answers, quizzes, reports (implemented program + documentation UML)
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