Course name:		
Programming paradigms		
Field of study: **Computer science**	Type of study: **Full-time**	Sourse code:
Course characteristics: **Mandatory within the additional content**	Level: **First**	Year: Semester:
Type of classes: **lectures, laboratories**	Hours per week:	ECTS points amount:

COURSE GUIDE

I. GENERAL INFORMATION OF THE COURSE

AIMS OF THE COURSE

A1. Obtaining knowledge in the basic programming paradigms.

A2. Obtaining knowledge in the selecting programming language to solve posed problem.

PRELIMINARY REQUIREMENTS FOR THE KNOWLEDGE, SKILLS AND OTHER COMPETENCIES

1. Knowledge of mathematics.

2. Basics of computer skills.

3. Basic knowledge in some high level programming language.

4. Ability to correctly interpret and present their own actions.

II. EFFECTS OF EDUCATION

EK 1 – able to characterize basic programming paradigms.

EK 2 – able to characterize different data type and use them in own programs.

EK 3 – able to define variable life time and correctly use variables in own programs.

EK 4 – able to solve basic problems by developing program in some functional programming language

EK 5 – able to solve basic problems by developing program in some logical programming language

PROGRAM OF EDUCATION

Lectures		Hours
Lect. 1	Introduction to Programming Paradigms	**1**
Lect. 2	Data Types in programming languages	**1**
Lect. 3	Variables in programming languages	**1**
Lect. 4	Names and their bindings	**1**
Lect. 5	Programming abstractions	**1**
Lect. 6	Lambda calculus	**1**
Lect. 7	Introduction to functional programming	**2**
Lect. 8	Functional programming in some high level functional language	**2**
Lect. 9	Introduction to logic programming	**1**
Lect. 10	Functional programming in Prolog	**2**
Lect. 11	Prototype object-oriented programming	**2**
Laboratories		**Hours**
Lab. 1	Introduction to Compilers, Coding, and Programs Execution	**2**

Lab. 2	Imperative programming I	2
Lab. 3	Imperative programming II	2
Lab. 4	Imperative programming III	2
Lab. 5	Introduction to functional language programming environment	2
Lab. 6	Lambda calculus	2
Lab. 7	Functional programming I	2
Lab. 8	Functional programming II	2
Lab. 9	Functional programming II in C# language	2
Lab. 10	Logic programming in Prolog I	2
Lab. 11	Logic programming in Prolog II	2
Lab. 12	Logic programming in Prolog III	2
Lab. 13	Programming in prototype base object oriented programming language	2
Lab. 14	Programming in prototype base object oriented programming language	2
Lab. 15	Exam	2

DIDACTIC TOOLS

1. – multimedial presentations for lectures
2. – instructions for laboratories
3. – wide range of algorithm and programming tools
4. – workplaces for students equipped with workstations

BASIC AND ADDITIONAL LITERATURE

W.F. Clocksin, C.S. Mellish "Programming in Prolog: Using the ISO Standard", Springer 2003
C. Smith "Programming F# 3.0", OReilly 2012
T. Liu "F# for C# Developers", Microsoft Press 2013
D. Watt „Programming Language Design Concepts", Wiley 2004
M. Scott "Programming Language Pragmatics", Morgan Kaufmann 2009

TEACHERS

dr inż. Łukasz Bartczuk lukasz.bartczuk@iisi.pcz.pl