Uniwersytet Gdański

Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

KAPITAŁ LUDZKI NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA EUROPEJSKI FUNDUSZ SPOŁECZNY

Course title	ECTS code
Introduction to declarative programming	11.3.1489

Name of unit administrating study

Faculty of Mathematics, Physics and Informatics

Studies

faculty	field of study	type	first tier studies (BA)
Faculty of Mathematics, Physics and Informatics	Informatics	form	full-time
		specialty	all
		specialization	all

Teaching staff

prof. UG, dr hab. Christoph Schwarzweller

Forms of classes, the realization and number of hours	ECTS credits
Forms of classes	6
Laboratory classes, Lecture	Course of 30 hours of lectures + 30 hours of labaratory + student's own work
The realization of activities	
classroom instruction	
Number of hours	
Lecture: 30 hours, Laboratory classes: 30 hours	

The academic cycle

2021/2022 winter semester

Type of course	Language of instruction
an elective course	polish

Teaching methods	Form and method of assessment and basic criteria for eveluation or examination requirements
- conducting experiments - designing experiments	**Final evaluation**
	- Graded credit
	- Examination
	Assessment methods
	- (mid-term / end-term) test
	- written exam with open questions
	- written exam (long written answer/problem solving)
	The basic criteria for evaluation
	colloquium after the laboratory
	written exam

Method of verifying required learning outcomes

Required courses and introductory requirements

A. Formal requirements
no formal requirements

B. Prerequisites
no starting requirements

Aims of education

Introduction to declarative programming based on the programming languages Scheme and Prolog

Course contents

1. Introduction to declarative programming
2. Scheme: Programming with functions, higher order functions, evaluation model
3. Prolog: Programming with relations, programmeing with terms, cut and negation

Bibliography of literature

Abelson, Sussman; Structure and Interpretation of Computer Programs

Bratko; Prolog --- Programming for Artificial Intelligence

The learning outcomes (for the field of study and specialization)

Student:

K_W02 has in-depth knowledge of the basic programming paradigms; also knows the current trends in programming languages

K_W04 knows formal computation models as well as their properties and importance in practical IT applications, has knowledge of computability barriers and computation difficulties

K_W05 knows the most important programming constructs and data structures

K_W09 is fluent in at least two programming languages and libraries of algorithms and data structures, and has knowledge of the practical conditions for efficient implementation of algorithms

K_W12 knows well the rules of occupational health and safety in the IT profession

K_U02 has the ability to design abstract data structures and their efficient implementation

K_U06 designs, analyzes in terms of computational correctness and complexity, and programs algorithms using various programming techniques

K_U07 can apply known algorithms in specific situations, can effectively select the type and method of implementation of the algorithm depending on the problem posed

Knowledge

Student:

knows paradigm declarative programming

knows model of computing in Scheme

knows programming language Prolog

Skills

Student:

• programs using the declarative paradigm paradigm

• can solve problems using the programming languages Scheme and Prolog

Social competence

Student:

• is able to use english literature

• knows the copyright regulations related to writing programs

Contact

Christoph Schwarzweller <schwarzw@inf.ug.edu.pl>