

FACULTY of Informatics and Management / DEPARTMENT .....

**SUBJECT CARD****Name in Polish:** Języki i paradygmaty programowania**Name in English:** Programming Languages and Paradigms**Main field of study (if applicable):** Informatics**Specialization (if applicable):** .....**Level and form of studies:** 1st/ ~~2nd~~\* level, full-time / ~~part-time~~\***Kind of subject:** ~~obligatory~~ / optional / ~~university-wide~~\***Subject code** INZ0289Wcl**Group of courses** YES / ~~NO~~\*

	Lecture	Classes	Laboratory	Project	Seminar
Number of hours of organized classes in University (ZZU)	30	15	15		
Number of hours of total student workload (CNPS)	60	60	30		
Form of crediting	Examination / crediting with grade*	Examination / crediting with grade*	Examination / crediting with grade*	Examination / crediting with grade*	Examination / crediting with grade*
For group of courses mark (X) final course	X				
Number of ECTS points	2	2	1		
including number of ECTS points for practical (P) classes	0		2		
including number of ECTS points for direct teacher-student contact (BK) classes	1,2	1,2	0,6		

\*delete as applicable

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

1. Ability to write simple applications in a high-level programming language.
2. Knowledge of basic algorithms and data structures.
3. Basic knowledge of object-oriented programming (preferably Java).

**SUBJECT OBJECTIVES**

C1 Basic understanding of fundamental programming paradigms and programming-language constructs.

C2 Understand the strengths and weaknesses of different programming paradigms.

## SUBJECT EDUCATIONAL EFFECTS

relating to knowledge:

PEK\_W01 Understand the characteristics and usage scenarios of a limited, but representative, set of programming languages (Scala, Java, OCaml).

PEK\_W02 Know common abstractions and the fundamental differences on how those abstractions are supported in programming languages.

relating to skills:

PEK\_U01 Solve small but characteristic problems in the programming languages addressed in the course.

PEK\_U02 Select the programming paradigm that best suits each problem.

PEK\_U03 Use a modern programming environment and programming tools.

PEK\_U04 Know and observe safety rules.

## PROGRAMME CONTENT

Form of classes - lecture		Number of hours
Lec 1	The role of abstraction in programming. Functional programming in interactive environment.	2
Lec 2	Basics of functional programming, cont.	2
Lec 3	Higher-order functions.	2
Lec 4	Defining new types.	2
Lec 5	Eager and lazy evaluation. Lazy lists.	2
Lec 6	Computational effects. Imperative programming.	2
Lec 7	Abstract data types. Modules and functors in OCaml.	2
Lec 8	Written test (1 hour). Object-oriented programming I - Java.	2
Lec 9	Object-oriented programming I - Java, cont.	2
Lec 10	Object-oriented programming II - Scala.	2
Lec 11	Generic classes and variance properties.	2
Lec 12	Most important forms of abstraction in programming languages.	2
Lec 13	Concurrent programming. Actors and message passing.	2
Lec 14	Written test.	2
Lec 15	Concurrent programming. Threads and shared memory.	2
	Total hours	30
Form of classes - class		Number of hours
Cl 1	Administrative class. Grading policy.	1
Cl 2	Basics of functional programming.	2
Cl 3	Higher-order functions. Defining new types.	2
Cl 4	Eager and lazy evaluation. Computational effects.	2
Cl 5	Abstract data types. Object-oriented programming in Java.	2
Cl 6	Object-oriented programming in Scala.	2

Cl 7	Generic classes and variance properties. Forms of abstraction in programming languages.	2
Cl 8	Concurrent programming. Actors and message passing.	2
	Total hours	15
<b>Form of classes - laboratory</b>		<b>Number of hours</b>
Lab 1	Administrative lab. Grading policy. Safety rules. Installation instructions for software used.	1
Lab 2	Basics of functional programming.	2
Lab 3	Higher-order functions. Defining new types.	2
Lab 4	Eager and lazy evaluation. Computational effects.	2
Lab 5	Abstract data types. Object-oriented programming in Java.	2
Lab6	Object-oriented programming in Scala.	2
Lab7	Generic classes and variance properties. Forms of abstraction in programming languages.	2
Lab8	Concurrent programming. Actors and message passing.	2
	Total hours	15
<b>TEACHING TOOLS USED</b>		
N1. Lecture supported by multimedia presentations.		
N2. Modern programming environment and programming tools.		
N3. E-learning system used to publish teaching materials and messages.		

#### EVALUATION OF SUBJECT EDUCATIONAL EFFECTS ACHIEVEMENT

<b>Evaluation</b> (F – forming (during semester), P – concluding (at semester end))	Educational effect number	Way of evaluating educational effect achievement
F1	PEK_W01 PEK_W02 PEK_U01 PEK_U02	Grading homework exercises solved at classes and declared as solved.
F2	PEK_W01 PEK_W02 PEK_U01 PEK_U02 PEK_U03 PEK_U04	Grading programs written on-line during labs.
F3	PEK_W01 PEK_W02	Written tests.
C The overall grade of the course is $(2T+C+L)/4$ rounded off. T is the grade for written tests (lecture), C is the grade for classes and L is the grade for labs.		
<b>PRIMARY AND SECONDARY LITERATURE</b>		

**PRIMARY LITERATURE:**

- [1] Handouts provided by the teacher.
- [2] M. Odersky, L.Spoon, B.Venners, Programming in Scala, Artima 2010.
- [3] J. Hickey, Introduction to Objective Caml, Internet.
- [4] P. Van Roy, S.Haridi, Concepts, Techniques, and Models of Computer Programming, MIT 2004.

**SECONDARY LITERATURE:**

- [1] R. W.Sebesta, Concepts of Programming Languages, Addison-Wesley 2012.
- [2] E. Chailloux, P.Manoury, B.Pagano, Developing Applications with Objective Caml, Internet.

**SUBJECT SUPERVISOR (NAME AND SURNAME, E-MAIL ADDRESS)**

Zdzisław Splawski, [zdzislaw.splawski@pwr.wroc.pl](mailto:zdzislaw.splawski@pwr.wroc.pl)

MATRIX OF CORRELATION BETWEEN EDUCATIONAL EFFECTS FOR  
SUBJECT  
**Programming Languages and Paradigms**  
AND EDUCATIONAL EFFECTS FOR MAIN FIELD OF STUDY  
Informatics  
AND SPECIALIZATION .....

Subject educational effect	Correlation between subject educational effect and educational effects defined for main field of study and specialization (if applicable)**	Subject objectives***	Programme content***	Teaching tool number***
<b>PEK_W01 (knowledge)</b>	K1INF_W05, K1INF_W06	C1, C2	Lec1-15, Cl2-8, Lab2-8	N1, N3
<b>PEK_W02</b>	K1INF_W06	C1, C2	Lec1-15, Cl2-8, Lab2-8	N1, N3
<b>PEK_U01 (skills)</b>	K1INF_U02, K1INF_W05, K1INF_W06	C1, C2	Cl2-8, Lab2-8	N2, N3
<b>PEK_U02</b>	K1INF_U02, K1INF_W05, K1INF_W06	C1, C2	Cl6-8, Lab6-8	N2, N3
<b>PEK_U03</b>	K1INF_U02, K1INF_W05	C1, C2	Lab2-8	N2
<b>PEK_U04</b>	K1INF_U14		Lab1-8	

\*\* - enter symbols for main-field-of-study/specialization educational effects

\*\*\* - from table above