SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Bachelor of Science
1.6 Program of study/Qualification	Computer science/ Engineer
1.7 Form of education	Full time
1.8 Subject code	25.

2. Data about the subject

2.1 Subject name	*Fundamental Programming Techniques*				
2.2 Course responsible/lecturer	S.L. dr. eng. Cristina Pop cristina.pop@cs.utcluj.ro Prof. dr. eng. Tudor Cioara tudor.cioara@cs.utcluj.ro				
2.3 Teachers in charge of seminars/ laboratory/ project	S.l. dr. ing. Marcel Antal marcel.antal@cs.utcluj.ro S.L. dr. eng. Cristina Pop cristina.pop@cs.utcluj.ro Conf. dr. ing. Viorica Chifu viorica.chifu@cs.utcluj.ro				
2.4 Year of study	II	2.5 Semester	2	2.6 Type of assessment (E - exam, C - colloquium, V - verification)	E
2.7 Subject category	*DF – fundamentală, DD – în domeniu, DS – de specialitate, DC – complementară*				DF
	DI – Impusă, DOp – opțională, DFac – facultativă				DI

3. Estimated total time

3.1 Number of hours per week	4	of which:	Course	2	Seminars	Laboratory	2	Project
3.2 Number of hours per semester	56	of which:	Course	28	Seminars	Laboratory	28	Project

3.3 Individual study:	
(a) Manual, lecture material and notes, bibliography	10
(b) Supplementary study in the library, online and in the field	16
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays	14
(d) Tutoring	
(e) Exams and tests	4
(f) Other activities:	

3.4 Total hours of individual study (suma (3.3(a)…3.3(f)))	44
3.5 Total hours per semester (3.2+3.4)	100
3.6 Number of credit points	4

4. Pre-requisites (where appropriate)

4.1 Curriculum	Fundamentals of Object Oriented Programming, Data Structures and Algorithms
4.2 Competence	Knowledge of Object Oriented Programming

5. Requirements (where appropriate)

5.1. For the course	Blackboard, projector, computer, internet; *Microsoft Teams platform for online teaching; Web site with course materials*
5.2. For the applications	Blackboard, projector, computer, internet; *Microsoft Teams platform for online teaching; Web site with laboratory materials*

6. Specific competence

6.1 Professional competences	**C4 -** Improving the performances of the hardware, software and communication systems **C4.1** - Identifying and describing the defining elements of the performances of the hardware, software and communication systems

	C4.2 - Explaining the interaction of the factors that determine the performances of the hardware, software and communication systems **C4.3 -** Applying the fundamental methods and principles for increasing the performances of the hardware, software and communication systems **C4.**4 - Choosing the criteria and evaluation methods of the performances of the hardware, software and communication systems **C4.5** - Developing professional solutions for hardware, software and communication systems based on performance optimization
6.2 Cross competences	N/A

7. Discipline objective (as results from the *key competences gained*)

7.1 General objective	Knowledge and using of object-oriented programming techniques for the development of professional software applications
7.2 Specific objectives	- to understand the software development process to use programming techniques for designing of classes and interfaces, including contracts and invariants. - to use programming techniques for code reuse by inheritance and polymorphism - to use generic type, streams, and lambda expressions as well as programming techniques for collection processing - to use programming techniques for reflection and event based - to use object-oriented and functional programming in an integrated approach for the development of flexible and efficient programs - to use lambda expressions and to be able to perform processing operations on streams -to use design patterns and frameworks for reusing design solutions -to use programming techniques for performance and software maintenance

8. Contents

8.1 Lectures	Hours	Teaching methods	Notes
Software development process	2	-Using modern multimedia teaching methods and direct access to internet; -Face to face and/or Online lecture presentations and discussions using the Microsoft Teams platform and course web site -Challenging questions during lecturers -Students are invited to collaborate in research projects -Personal assistance hours the semester and before the exam	N/A
Design view: UML Diagrams	2		
OOP Paradigms	2		
SOLID Principles	2		
Compositional Techniques	2		
Class Design Principles	2		
Design by contract	2		
Reflection & Code refactoring	2		
Design Patterns Overview	2		
Software Quality	2		
Generics	2		
Lambda Expressions	2		
Streams Processing	2		
Performance and Optimization	2		

Bibliography

1. Joshua Bloch - Effective Java, 3/e Addison Wesley, 2018
2. Martin Fowler - Refactoring. Improving the design of existing code, 2/e, Addison Wesley 2019
3. Kishori Sharan, Beginning Java 8 Language Features: Lambda Expressions, Inner Classes, Threads, I/O, Collections and Streams, Apress, 2014
4. Urma RG, Fusco M, Mycroft A – Java 8 in Action: Lambdas, streams, and functional style programming, Manning, 2015

| 5. Subramaniam, V. – Functional Programming in Java, The Pragmatic Programmers, 2014 |
| 6. Online course materials provided by the course lecturer |

8.2 Applications – Seminars/Laboratory/Project	Hours	Teaching methods	Notes
Intro to lab resources and requirements	2	Short presentation of the laboratory assignments, discussions about the assignments, assignments implementation on the computer, face-to-face/on-line discussions and evaluations – for on-line activities the Microsoft Teams platform will be used.	N/A
Assignment 1 – Programming techniques with inheritance and polymorphism	4		
Assignment 2 – Programming techniques with threads	6		
Assignment 3 - Programming techniques with databases, design patterns and reflection	4		
Assignment 4 – Programming techniques with design patterns, contracts and invariants, Java Collection Framework, lambda expressions and stream processing	8		
Lab Evaluation	4		

Bibliography
- http://docs.oracle.com/javase/tutorial/index.html
- http://stackoverflow.com/

Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

Fundamental Programming Techniques is a subject of the domain "Computers and Information Technology". It teaches students to apply object-oriented programming techniques in designing and implementing of software applications. The content was developed based on the analysis of similar disciplines from other universities as well as based on the requirements of the IT employees. The content was also evaluated by Romanian governmental agencies CNEAA and ARACIS.

10. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course	How the students are using programming techniques for: (i) designing of classes and interfaces, including contracts and invariants; (ii) promoting code reuse through inheritance and polymorphism; (iii) using generic type, streams and lambda expressions; (iv) using programming techniques for concurrent and multi-threading programming; (v) using object-oriented and functional programming in an integrated approach for the development of flexible and efficient programs; (vi) using design patterns and frameworks	written exam, face to face or online supervised by using the Microsoft Teams platform	50%
Seminar	-	-	-
Laboratory	-Abilities to effectively specify, design, implement and test quality and performance object – oriented programs -Quality of code and assessment deliverables -Activity and presence during lab sessions	-Assessment of programming assignments during the semester face to face and/or online using the Teams platform	50%
Project	-	-	-
Minimum standard of performance:			
-To be able to use object-oriented programming techniques in designing and implementing software applications			
Grade: 50% laboratory + 50% final exam			

Conditions for participating in the final exam: Laboratory ≥ 5
Handing over all laboratory assignments and obtain a minimum grade of 5 on each assignment; At least 11 laboratory presences.
Conditions for promotion: final exam ≥ 5
Handing overdue laboratory assignments: in an overdue session a student can hand over 1 of the unfinished semester laboratory assignments.

Date of filling in:	**Titulari**	**Titlu Prenume NUME**	**Semnătura**
	Course	S.L.dr.eng. Cristina Pop	
		Prof.dr.eng. Tudor Cioara	
	Applications	S.l.dr.ing. Marcel Antal	
		S.l.dr.ing. Cristina Pop	
		Conf. dr. ing. Viorica Chifu	

Date of approval in the department	Head of department Prof.dr.ing. Rodica Potolea
Date of approval in the Faculty Council	Dean Prof.dr.ing. Liviu Miclea