MINISTRY OF NATIONAL EDUCATION

TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA

SYLLABUS

1. Study Program

1.1	Higher Education Institute	Technical University of Cluj-Napoca
1.2	Faculty	Electronics, Telecommunications and Information Technology
1.3	Department	Communications
1.4	Study domain	Electronics and Telecommunications Engineering
1.5	Study level	License
1.6	Study program/ Qualification	Telecommunications Technologies and Systems, Applied Electronics
1.7	Type of education	IF (Full-time learning)
1.8	Discipline code	TST-E12.00, EA-E12.00

2. Discipline

2.1	Discipline name	Computers Programming: Algorithms
2.2	Subject area	Electronics and Telecommunications Engineering
2.3	Responsible	Professor: Mircea-Florin Vaida, PhD Mircea.Vaida@com.utcluj.ro
2.4	Titular	Professor: Mircea-Florin Vaida, PhD. Collaborator: Cosmin Striletchi, PhD.

2.5	Year of study	I	2.6	Semester	2	2.7	Evaluation	Verif.	2.8	Type of discipline	DF/ DOB

3. Total estimated time

Year/ Sem	Discipline name	No. of weeks	Course	Applications			Course	Applications			Indiv. study	TOTAL	ECTS
			[hours/week]				[hours/week]						
			C	S	L	P		S	L	P			
I/2	Computers Programming: Algorithms	14	2	0	2	0	28	0	28	0	74	130	5

3.1	Number of hours per week	4	3.2	Course	2	3.3	applications	2
3.4	Total hours per curriculum	56	3.5	course	28	3.6	applications	28
Individual study								Hours
Study based on manuals, course materials, references and notes								56
Supplementary documentation in libraries, electronic platforms and on field								8
Preparation of seminars/laboratories, homework's, essays, portfolios								4
Tutorial work								2
Assessments								3
Other activities								1

3.7	Total hours of individual study	74
3.8	Total hours per semester	130
3.9	ECTS	5

4. Prerequisites (if necessary)

4.1	Curriculum	Basic knowledge from: - Computer programming – Languages course
4.2	Competences	Basic knowledge of algorithms

5. Requisites (if necessary)

5.1	Course	Video-projector, screen, whiteboard, blackboard
5.2	Applications	PCs with Internet access

6. Specific competences acquired

Professional competences	Theoretical knowledge (What do the student should know)	- Basic concepts about algorithms and programming techniques - Basic elements about OOP in C/C++ - Different programming abilities for sequential and linked data in C/C++
	Acquired skills (What the student is able to do)	To develop: -Algorithms and programming techniques: -recursive alg. -searching and sorting alg. -backtracking and divide et impera methods -Object Oriented Programming-OOP: -classes, objects -overloading methods and operators -inheritance, virtual methods and classes -objectual I/O in C++, files in C++ -Different programming abilities for sequential and linked data in C/C++ -linked lists and trees including stack and queue
	Acquired abilities (what equipment/ instruments/ software the student is able to handle)	After studying this discipline, the students will be able to: - Know main facilities of an OOP IDE, VC++IDE - To execute, test and debug OO applications with dedicated algorithms in C/C++.
Transversal competences		CT3 Adapting to new technologies, professional and personal development through continuing education using electronic documentation and printed sources, in Romanian and in at least one international language (English). Competencies for analysis and synthesis and optimization systems thinking. Flexibility in thinking and ability to work with interdisciplinary concepts and tools.

7. Discipline objectives (based on the grid of specific competences acquired)

7.1	General objective	Development of competences in basic algorithms and C++ OO programming language
7.2	Specific objectives	1. Theoretical knowledge's about basic OO programming in C++ language. 2. Practical abilities to use Visual Studio C++ IDE for OO and algorithms applications.

8. Contents

8.1. Course (titles)		Teaching methods	Observations
1	Recursive programming in C/C++. Stack management. Recursive and non-recursive programming methods. Backtracking.	Presentations, discussions	Videoprojector
2	Recursive and non-recursive programming methods. Variants of Backtracking method. Divide et impera method. Sorting and searching algorithms. Simple sorting: selection, insertion, interchange.		
3	Advanced sorting: merge sort, quick-sort. Introduction in Object Oriented Programming, OOP.		

4	Classes, Objects, members of a class. Constructors, destructors, methods calling in C++. Copy constructor, arrays of objects, visibility domain.		
5	Friend class and functions in C++. Static members. Struct and union in C++. Overloading methods.		
6	Overloading operators in C++.		
7	Inheritance in C++. Simple and multiple inheritances.		
8	Virtual classes and methods. Abstract classes.		
9	I/O operations in C++. iostream library, I/O with format, I/O state, manipulators functions		
10	ostream, istream si fstream classes. Overriding I/O operators. C++ files.		
11	Stack, queue, sequential lists.		
12	Linked lists: SLL, DLL		
13	Trees: definitions, properties. Binary trees, operations		
14	Theoretical evaluation		
8.2. Applications (laboratory work)		Teaching methods	Obser-vations
1	Macro functions. Inline functions. Functions with implicit parameters. Functions with a variable number of parameters. Overloading functions	Experiments, tests using PC's	Network PC's
2	Recursive functions.		
3	Recursive and non-recursive programming methods: Backtracking, divide et impera: searching techniques.		
4	Sorting techniques.		
5	Classes, objects, class members.		
6	The access to a class's members		
7	Constructors. Destructors. Object arrays		
8	Friend functions and classes. Static members.		
9	Operators overloading.		
10	Simple and multiple inheritances		
11	Virtual methods and classes. Abstract classes.		
12	Input/output in C++. Redefining the I/O operators.		
13	Files in C++. Homework evaluation		
14	Final practical test and evaluation.		

References:

In TUC-N library

1. Vaida M., Bazele dezvoltarii aplicatiilor software in electronica si telecomunicatii, curs, litografia UTC-N, 1997
2. Mircea-Florin Vaida, Lenuţa Alboaie, Petre Gavril Pop, Cosmin Strileţchi, Ligia-Domnica Chiorean, Programare orientata pe obiecte si programare web, Editura: Casa Cărţii de Ştiinţă, Cluj-Napoca, pp. 245, 2011
3. Ligia Chiorean, Mircea-Florin Vaida, Petre G. Pop, Cosmin Striletchi, , Elemente de bază şi obiectuale privind dezvoltarea aplicaţiilor în limbajul de programare C/C++, pp. 380, UTPress, 2007/2008

Additional materials

- course notes at http://helios.utcluj.ro/lab/index.php
- laboratory materials available on the website http://helios.utcluj.ro/lab/index.php

In other libraries

1. Striletchi C., Vaida M.F., Pop G.P., Chiorean Ligia, Benta K. Iulian- Tehnologii obiectuale si algoritmi de baza privind dezvoltarea aplicatiilor in limbajul C/C++, Editura Casa Cartii de Stiinta, Cluj-Napoca, 2007
2. Ligia-Domnica Chiorean, Kuderna-Iulian Benţa, Mircea-Florin Vaida, Petre Gavril Pop, Cosmin Strileţchi, Elemente practice de bază pentru programarea în limbajul C/C++, Casa Cartii de Stiinta, Cluj-Napoca, 2012/2013.

9. Discipline content corroborated with the expectations of the epistemic community representatives, associations, professional and related program employers

Acquired skills will be needed in the following possible COR occupations: electronics engineer, telecommunications engineer, system and computer design engineer, or new occupations proposed to be included in COR (sales support engineer, developer of multimedia applications, network operating engineer, test engineer, project manager, traffic engineer, communications system consultant.

10. Assessment

Type of activity	10.1	Evaluation criteria	10.2	Evaluation method	10.3	The weight of the final grade
Course		Theoretical written and oral test with questions/code		Written/oral test (T=33%)		T = 33%
Application		Solving a problem P on a computer (1 hour). The laboratory L will also be evaluated		Lab. evaluations and computer test (P=34%, L=33%)		P+L = 67%
10.4 Minimum performance standard						
The final grade (N) is calculated as average of marks obtained in the evaluation of ongoing activities and application type: N = (T + L + P) / 3.0. The condition for obtaining the ECTS credits is that N and all components of the final grade to be higher than or equal to 5 (five).						

Date
28.01.2015

Titular
Professor
Mircea-Florin Vaida, Ph.D.

Responsible
Professor
Mircea-Florin Vaida, Ph.D.

Date of approval
28.01.2015

Head of department
Professor Virgil Dobrota, Ph.D.