THE BUCHAREST ACADEMY OF ECONOMIC STUDIES
Faculty: ECONOMIC CYBERNETICS, STATISTICS AND INFORMATICS
Chair: Economic Informatics

SYLLABUS
Academic year: 2010-2011

Course title	**DATA STRUCTURES**					
				Hours per semester		
Course code	**0100101OS311220L**	Numbers of points	**5**	Total	Lectures (C)	Seminar (S)	Laboratory/ project work (L/P)
				56	**28**	**28**	**-**

Course title	**DATA STRUCTURES**						
				Hours per semester			
Course code	**0100101OS311220L**	Numbers of points	**5**	Total	Lectures (C)	Seminar (S)	Laboratory/ project work (L/P)
				56	**28**	**28**	**-**
Faculty where delivered	**ECONOMIC CYBERNETICS, STATISTICS AND INFORMATICS**				Year of study		**3**
					Semester		**1**
Specialisation	ECONOMIC INFORMATICS						
Course type: F – fundamental, S – specialised, C – complementary							**S**
Course curricular category : C – compulsory, E – elective, F - free, S - special							**C**
Pre-requisites	Compulsory						
	Recommended						
Learning objectives	Initiation in defining and using data structures adequately.						

Course contents (descriptors)	
	• Basic concepts: memory areas, referring address, contextual content, abstract data defining models, data classification criteria, data presentation models analytical, graphic, graph, textual
	• Sparse matrices and arrays: models, properties, operations – definitions, initializations, traversals, updates; sparse matrix: encoding with arrays, lists, aggregated lists, sparse matrix operations; storing arrays into files, generalizations; array operations libraries;
	• Heterogeneous data structures: articles, files, databases, repository; internal encoding, searches, retrieval, traversal; referring expressions, aggregation;
	• Lists, stacks, queues: definition, models: analytical, graphic, textual; operations: creation, traversal, insertion, deletion, concatenation, conversion, aggregation;
	• Trees: definition, models: analytical, graphic, textual; operations: creation, traversal, insertion, deletion, concatenation, conversion, aggregation; B trees, AVL trees;
	• Data structures optimizations;
	• Object oriented data structures in complex applications;
	• Using data structures as resource allocation and leveling process in complex projects for developing economic problems oriented software
	• A project must be completed with the following objective: efficient use of data structures. The project structure is as follows: defining the objective, the necessary resources and the proposed and implemented solution. For the elaboration of the project a bibliography not older than 5 years is needed. The project structure includes: introduction, problem identification, elaborating possible solutions for implementation; defining how the solution will be identified; efficiency; the development cycle of the final product; quality analysis methods; conclusions; bibliography; annexes.
	• The project must be loaded on an online platform for orthogonally analysis in the 14th week of the semester in which the course is studied. A complete and correct approach, theoretically and experimentally, allows students to obtain the 20% assigned for this task. The students have the right of doing the project again for obtaining the respective points in the re-examination session, if they agree and whish this.
	• The students must give a solution to each one of the 5 problems received during the semester, earning so a 10% points.
	• For their activity in elaborating practical tasks for the course Data Structures, the students will obtain 20% points. So, for the activity and their effort in a permanent and rhythmic preparation, thorough and well documented, students get a maximum 50% points and the rest of 50% are assimilated with the final exam.

Type of assessment (**E** – exam, **A** – continuous assessment, **C** – colloquium)		**E**
Assesment percentage	Final exam/assessment paper	50%
	Projects/Essay(s) + attendance	50%
Bibliography	• Cristian BOLOGA - *Algoritmi si structuri de date,* Editura RISOPRINT, Cluj-Napoca, 2005, ISBN 973-651-003-8, 323pg.	
	• I.Smeureanu, I. Ivan, M. Dârdală, *Limbajul C/C++ prin exemple probleme*, Ed.CISON,	

	Bucureşti, 1995
	• Ion IVAN, Cristian IONITA, Cătălin BOJA, Marius POPA, Adrian POCOVNICU, Daniel MILODIN - *Practica dezvoltarii software orientata pe structuri de date*, Editura ASE, Bucuresti, 2005, ISBN 973 - 594-630-0, 223 pg.
	• Mirela Catrinel VOICU – *Aplicaţii cu baze de date şi structuri de date în Java utiliyând mediul de dezvoltare JBuilder*, Ed. Universităţii de Vest, 2007
	• Saumeyendra Sengupta, Carl Phillip Korobkin - *C++ Object Oriented Data Structures*, Springer Verlag, New York, 1994
	• William Ford, William Topp – *Data Structures with* C++, Prentice Hall Inc., New Jersey, 1996, ISBN 0-13-320938-5.

Instructors	Position, title, first name, surname	Signature
	Prof. Ion IVAN, PhD	

Legend: **L** – lecture; **S** – seminar; **L/P** – laboratory/project work.