ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
Faculty of Electrical and Computer Engineering
COURSE SYLLABUS
Data Structures

1. CODE AND NUMBER OF CREDITS

CODE	FIEC03012	
NUMBER OF CREDITS: 5	**Theoretical:4**	**Practical:1**

2. COURSE DESCRIPTION

This course covers the definition, design and implementation of abstract data types with emphasis on the application of abstraction, modularity and encapsulation techniques. Basic abstract data types such as lists, stacks, queues, graphs, trees and sets, and their possible applications are reviewed. Finally, we present medium complexity problem for students, and in teams, they must design and implement a solution, using for this a programming language and a suitable coding style.

3. PRE-REQUISITES AND CO-REQUISITES

PRE-REQUISITES	FIEC04341 PROGRAMMING FUNDAMENTALS
CO-REQUISITES	

4. CORE TEXT AND OTHER REQUIRED REFERENCES FOR THE TEACHING OF THE COURSE

CORE TEXT	1.This course is self-contained
REFERENCES	1. Estructura de Datos, Luis Joyanes Aguilar, Ignacio Zahonero,3ra Edicion,1999, McGrawHill 2. Estructuras de datos y algoritmos, Aho, Hoptcroft, Ullman. 1ra Edicion,1987, Addison-Wesley 3. Estructura de Datos en C, Aaron M. Tenembaum, Yedidyah Langsam, Moshe A. Augenstein, 2da Edicion,1999,Addison-Wesley. 4. Data Structures and Their Algorithms, Harry R. Lewis,1ra Edicion,1991, Larry Denenberg, Prentice Hall

5. COURSE LEARNING OUTCOMES

At the end of the course, the student will be able to:

1. Apply in an effective way abstraction, modularity and encapsulation concepts for solving programming problems
2. Design and implement new Abstract data types
3. Identify, understand and use the definition and behavior of basic abstract data types: lists, queue, stack, graph, tree, maps
4. Evaluate, in group or individually, the beast way to design and implement a the solution of a problem using abstract data types
5. Design and implement, in group or individually, the best way to design and implement a the solution of a problem using basic and new abstract data types.
6. Use a programming language for coding and exploiting a coding style to improve code writing and maintenance

6. COURSE PROGRAM

I. COURSE POLICIES AND METHODOLOGIES (1 SESSION – 2.5 HOURS).
II. DATA TYPES (3 SESSIONS – 7.5 HOURS).

- Simple Data Type
- Pointers
- Chains
- Arrays
- Enumerations
- Structures
- Lab

III. ABSTRACT DATA TYPES (ADTS) (3 sessions – 7.5 hours).
- TDA concept and advantages
- BNF representation
- Generic TDA
- Course programing tools
- Lab

IV. LISTS (3 sessions – 7.5 hours).
- Definition
- Advantages, disadvantages and applications
- Simple Linked lists: static and dynamic implementation
- Lists variants
- Lab

V. STACK (2 sessions - 5 hours).
- Definition
- Advantages and disadvantages
- Dynamic Implementation
- Applications
- Lab

VI. QUEUE (2 sessions - 5 hours).
- Definition
- Advantages and disadvantages
- Dynamic Implementation
- Applications
- Lab

VII. MAPS (2 session - 5 hours).
- Definition
- Advantages and disadvantages
- Dynamic Implementation
- Applications
- Lab

VIII GRAPHS (6 sessions - 15 hours).
- Project Review
- Definition
- Advantages and disadvantages
- Dynamic Implementation
- Applications
- Lab
- Algorithms
- Lab

IX TREES (6 sessions - 15 hours).
- Definition
- Terminology
- ADT binary tree
- Lab
- ADT binary search tree
- Lab
- ADT Heap
- Multipath trees
- Lab

7. WORKLOAD: THEORY/PRACTICE

2 sessions a week, 2.5 hours each

8. CONTRIBUTION OF THE COURSE TO THE EDUCATION OF THE STUDENT

Student outcomes addressed by this course. Students review basic concepts of computer science and the definition and implementation of abstract data types such as lists, stacks, queues, graphs, trees and sets. Students review the data structures that implement ADTs lists, stacks, queues, graphs, trees and sets, as well as the main algorithms associated with each. New ADTs are also implemented, according to the programming language Students analyze, design and implement software solutions to real problems by applying the theoretical concepts reviewed in the course. Students use a high-level programming language to implement their software solution, and they solve problems of medium complexity.

BASIC TRAINING	PROFESSIONAL TRAINING	SOCIAL SKILLS DEVELOPMENT
	X	

9. THE RELATIONSHIP BETWEEN THE LEARNING OUTCOMES OF THE COURSE AND THE LEARNING OUTCOMES OF THE DEGREE PROGRAM

LEARNING OUTCOMES OF THE DEGREE PROGRAM*	CONTRIBUTION (High, Medium, Low)	LEARNING OUTCOMES OF THE COURSE**	THE STUDENT MUST:
a) An ability to apply knowledge of computing and mathematics appropriate to the discipline	High	1,2,3,5	Apply abstraction, modularity, encapsulation concepts and demonstrate the good use of use ADT
b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution	Medium	5	Study and identify the best way to design and implement a software problem solution based on ADT.
c) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs	High	2,5	Design and implement a software problem solution based on basic and new ADT.
d) An ability to function effectively on teams to accomplish a common goal	Medium	4,5	Design and implement a software problem solution as part of a team.
e) An understanding of professional, ethical, legal, security and social issues and responsibilities	---		
f) An ability to communicate effectively with a range of audiences	---		
g) An ability to analyze the local and global impact of computing on individuals, organizations, and society	---		
h) Recognition of the need for and an ability to engage in continuing professional development	---		
i) An ability to use current techniques, skills, and tools necessary for computing practice.	Medium	6	Use a programing language and tools for coding software programs using a good coding style.
j) Ability to lead projects	---		

10. EVALUATION IN THE COURSE

Evaluation activities	
Exams	X
Tests	X
Homework/tasks	X
Projects	X
Laboratory/Experiments	X

Class participation	
Visits	
Other	

11. PERSON RESPONSIBLE FOR THE CREATION OF THE SYLLABUS AND THE DATE OF ITS CREATION

Created by	Daniel Ochoa
Date	26 Feb 2013

1. APPROVAL

ACADEMIC SECRETARY OF THE ACADEMIC DEPARTMENT	**DIRECTOR OF TECHNICAL ACADEMIC SECRETARY**
NAME: Sra. Leonor Caicedo G.	NAME: Ing. Marcos Mendoza V.
SIGNATURE:	SIGNATURE: ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL Ing. Marcos Mendoza V. DIRECTOR DE LA SECRETARIA TÉCNICA ACADÉMICA
Date of approval by the Directive Council: 2013-334 2013-08-12	

2. VALIDITY OF THE SYLLABUS

RESOLUTION OF THE POLYTECHNIC BOARD:	13-10-269
DATE:	2013-10-17