NATIONAL UNIVERSITY OF ENGINEERING
COLLEGE OF INDUSTRIAL AND SYSTEMS ENGINEERING

INDUSTRIAL ENGINEERING PROGRAM

SYLLABUS - ST221 ALGORITHMS AND DATA STRUCTURES

I. GENERAL INFORMATION

CODE	**:**	ST221 Algorithms and Data Structure
SEMESTER	**:**	2
CREDITS	**:**	3
HOURS PER WEEK	**:**	4 (Theory – Practice - Laboratory)
PREREQUISITES	**:**	CB121 Differential Calculus
CONDITION	**:**	Compulsory

II. COURSE DESCRIPTION

The course prepares students for the formulation of algorithms to solve different problems in mathematics and engineering. Students understand and apply different types of data structures and develop algorithms for arrays processing based on sorting, ordering, searching using different selective, sequential and repetitive control structures. Algorithms and program optimization criteria are analyzed in terms of processing and resources requirements, simplicity and efficiency. Intensive analysis of algorithms and their computer implementation is promoted throughout the course.

III. COURSE OUTCOMES

At the end of the course, students:

1. Develop algorithms as a set of instructions to solve problems using computing methods.
2. Understand the meaning and scope of different types of data structures and apply them according to the characteristics of the problem being solved.
3. Develop computer programs using multi-dimensional arrays, loops, and sequential, selective and repetitive control structures, recursion.
4. Considers program optimization criteria in terms of processing requirements, execution times, memory and resources requirements.
5. Develop structured and object oriented computer programs.

IV. LEARNING UNITS

1. PROBLEMS, ALGORITHMS AND PROGRAMS

Introduction to problems, algorithms and programs / Methodologies for problem solving / Techniques for algorithms formulation / Algorithms specification: Imperative, functional / Algorithm complexity analysis: efficiency, complexity / Recurrence equations / Non-recursive algorithm complexity / Recursive complexity algorithms.

2. PSEUDO-CODE PROGRAMMING

Variables / Local and global variables / Sequential control structures / Selective control structures: simple, multiple, nested / Repetitive control structures / Fixed number of repetition loops / Loop break conditions / Controlled input loops/ Controlled output loops / Nested loops.

3. ALGORITHMS DESIGN TECHNIQUES

Vectors and matrices / Vector operations / Matrix operations / Chains: creation, use and operations / Chain arrays / Pointers / Algorithms / Brute force method / Recursion / Divide and win / Dynamic programming / Voracious algorithms / Input and output.

4. SEARCH AND ORDERING METHODS

Bubble method / Direct insertion method / Shell sort method / Heap sort method / Quick sort method / Merge sort methods / Applications.

5. PROGRAMMING LANGUAGE FOR DATA STRUCTURE

Modular programming / Object oriented programming / Design patterns / Heritage and polymorphism / Data types in execution time / Interfaces /Execution faults: classification representation and treatment / Fault-tolerant software / Applications.

6. DYNAMIC PROGRAMMING

Pointers: creation, use / Pointer operations / Hierarchical data structures / Multipath hierarchical data structures / Network-type data structures / Graphs / Simple linked lists / Ordered linked lists / Doubled link lists / Trees / Search binary trees / Queues / Applications.

V. LABORATORIES AND PRACTICAL EXPERIENCES:

Laboratories using C++ language:

Laboratory 1: Introduction algorithms	Laboratory 5: Ordering methods
Laboratory 2: Algorithms complexity	Laboratory 6: Object oriented programs
Laboratory 3: Algorithms design techniques	Laboratory 7: Linked lists
Laboratory 4. Algorithms design techniques	Laboratory 8: Double linked lists

VI. METHODOLOGY

This course is carried out in theory, practical and lab sessions. In theory sessions, the instructor introduces concepts, theorems and applications. In practical sessions, different problems are solved and their solutions are analyzed. In laboratory sessions, students implement the algorithms using PSelnet and C languages. At the end of the course, students must submit and defend an integrating project. In all sessions, students' active participation is encouraged.

VII. EVALUATION FORMULA

Evaluation system F.

The average grade PF is calculated as follows:

PF = (EP + EF + 2 PP)/4

EP: Mid-Term Exam EF: Final Exam

PP: Quizzes average (4 and 1 is eliminated)

VIII. BIBLIOGRAPHY

1. **M.A. WEISS**
   Data Structures and Algorithms. Analysis in Java.
   Addison Wesley, 2010.
2. **D. MALIK**
   Data Structures Using C++
   Thomson Learning, 2014.
3. **L. JOYANES and I. ZAHONERO**
   Data Structures in Java.
   McGraw Hill, 2008