



University/Academy: Arab Academy for Science and Technology & Maritime Transport

Faculty/Institute: College of Computing and Information Technology

Program: Computer Science/ Information Systems/ Software Engineering

**Form No. (12)
Course Specification**

1- Course Data

Course Code: CS143	Course Title: Introduction to Problem solving	Academic Year/Level: Year 1 / Semester 2
Specialization: Computer Science	No. of Instructional Units: 2 hrs lecture 2 hrs lab 2hrs section	Lecture:

2- Course Aim	This course will present problem solving heuristics and structured programming techniques, present language independent data types, operations, programming constructs and statements, introduce arrays, and implement fundamental programs using an appropriate programming language.
----------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

3- Intended Learning Outcome:

a- Knowledge and Understanding	<p>Students will be able to demonstrate knowledge of:</p> <p>K1. Essential facts, concepts, principles and theories relating to computing and information and computer applications as appropriate to the program of study.</p> <p>K2. Modeling and design of computer-based systems bearing in mind the trade-offs.</p> <p>K3. Tools, practices and methodologies used in the specification, design, implementation and evaluation of computer software systems.</p> <p>K4. Criteria and specifications appropriate to specific problems, and plan strategies for their solution.</p> <ul style="list-style-type: none">• Explain the steps for problem solving.• Know the different types of problems.• Know the different problem solving techniques.• Understand what is pseudo-code.• Identify different flowchart symbols and their uses.• List the different data types and operators.• Understand the importance of modular programming.• Understand how functions work and know what are function parameters.• Understand the difference between value and reference parameters.• Define different selection control structures.
---------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

	<ul style="list-style-type: none"> • Define the different looping structures. • Define arrays and their uses. • Explain different searching techniques. • Explain different sorting techniques. • Understand and define recursive problems. • Define structures and their uses. • Understand the representation of strings.
b- Intellectual Skills	<p><u>By the end of the course, the student acquires high skills and an ability to understand:</u></p> <p>I1. Analyze computing problems and provide solutions related to the design and construction of computing systems.</p> <p>I3. Identify criteria to measure and interpret the appropriateness of a computer system for its current deployment and future evolution.</p> <p>I10. Define traditional and nontraditional information systems problems, set goals towards solving them, and. observe results</p> <p>I11. Perform comparisons between (methods, techniques...etc).</p> <ul style="list-style-type: none"> • Analyze a given problem and determine the requirements from the problem statement. • Examine a written pseudo-code and determine its function. • Examine a flowchart and determine its function. • Design solutions to problems using functions. • Design solutions to problems using selection. • Design solutions to problems using loops. • Design solutions to problems using arrays. • Evaluate different searching techniques. • Evaluate different sorting techniques. • Design solutions to problems using recursion. • Design solutions to problems using structures. <p>Design solutions to problems using strings.</p>

c- Professional Skills	<p><u>By the end of the course the student will have the ability to:</u></p> <p>P2. Implement comprehensive computing knowledge and skills in projects and in deployment of computers to solve position practical problems.</p> <p>P3. Deploy the equipment and tools used for the construction, maintenance and documentation of computer applications.</p> <ul style="list-style-type: none"> • Apply problem-solving steps. • Use a programming language interface. • Write pseudo code. • Draw flowcharts for a given problem. • Implement a simple program that illustrates the use of different data types. • Implement programs that evaluate arithmetic and logical expressions. • Implement solutions that use functions with both value parameters and reference parameters. • Implement programs that use the selection control structures. • Implement programs that use looping structures. • Implement programs that use arrays. • Implement searching techniques. • Implement sorting techniques. • Implement recursive functions.
-------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

	<ul style="list-style-type: none"> • Implement programs that use structures. • Implement programs that use strings.
d- General Skills	<p>Students will be able to:</p> <p>G1. Demonstrate the ability to make use of a range of learning resources and to manage one's own learning</p> <p>G2. Demonstrate skills in group working, team management, time management and organizational skills.</p> <p>G7. Show the use of general computing facilities.</p>
4- Course Content	<ol style="list-style-type: none"> 1. Overview of Computers and Programming. 2. Pseudo-code and Flowcharts. 3. Overview of Data Types, Expressions, and Statements. 4. Top-Down Design with Functions. 5. Selection Control Structures. 6. Looping Control Structures. 7. Problem Solving with Arrays. 8. Searching Techniques. 9. Sorting Techniques. 10. Recursive Problems. 11. Structures. 12. Strings.
5- Teaching and Learning Methods	Lectures, Labs, Projects, Individual study & self-learning.
6- Teaching and Learning Methods for Students with Special Needs	<ul style="list-style-type: none"> • Students with special needs are requested to contact the college representative for special needs (currently Dr Hoda Mamdouh in room C504) • Consulting with lecturer during office hours. • Consulting with teaching assistant during office hours. • Private Sessions for redelivering the lecture contents. • For handicapped accessibility, please refer to program specification.
7- Student Assessment:	
a- Procedures used:	Exams and Individual Projects
b- Schedule:	<p>7th week exam 30%</p> <p>Project 20%</p> <p>Lab 10%</p> <p>Final exam 40%</p>

c- Weighing of Assessment:	<hr/> Week 7 Grades – 30% Week 12 -Grades – 20% Lab 10% <hr/> Week 16 - Final Exam – 40%
8- List of References:	
a- Course Notes	From the Moodle on www.aast.edu
b- Required Books (Textbooks)	Jeri Hanly, Elliot Koffman, <i>Problem Solving And Program Design In C</i> , Addison Wesley, Sixth Ed
c- Recommended Books	Walter J. Savitch, <i>Problem Solving with C++: The Object of Programming</i> , Addison Wesley.
d- Periodicals, Web Sites, ..., etc.	

Course Instructor: Dr Mostafa AbdelAzeem Head of Department: Dr Samah Senbel

Sign

Sign