جامعة الأزهر - غزة
Al Azhar University - Gaza

Planning and Quality Assurance Affairs

Form (A)

Course Specifications

General Information

Course name	Introduction to Computing
Course number	ITCS1401
Faculty	
Department	
Course type	College Needs
Course level	1
Credit hours (theoretical)	3
Credit hours (practical)	1
Course Prerequisites	

Course Objectives

1 - Increase familiarity with computers, their components and their operations.

2 - Explain how computers store and manipulate information.

3 - Identifying the steps involved in creating a program and itemize the elements of a typical program.

4 - Understanding the essential of the concepts of Programming Techniques.

5 - Understanding the nature and function of a high level language’s constructs and syntax(C++)

6 - The concept of control constructs

7 - The concept of Looping techniques

8 - key concepts of simple and dynamic data structures

9 - Describe how functions are constructed and implemented.

10 - Give a brief account of library functions and user-defined functions.

11 - Write a complete C++ program using the different statements of the C++ language.

Intended Learning Outcomes

Knowledge and Understanding

- a1- Use high-level programming languages.
- a2. Outline fundamental topics in computer systems, including hardware architectures and operating systems
- a3. Define the concept of an algorithm and a few specific examples of algorithms
- a4. Identify the essential of the concepts of Programming Techniques.
- a5. Describe fundamental topics in computing including software architectures, software engineering principles and methodologies and software tools.
- a6. Outline basic knowledge and understanding of a core of analysis.
- a7. Explain the nature and function of a high level language's constructs and syntax(C++)
- a8. Describe the concept of control constructs
- a9. Explain Looping techniques
- a10. Describe key concepts of simple and dynamic data structures
- a11. Identifythe tools, practices and methodologies used in the specification, design, implementation and evaluation of computer programs.

Intellectual Skills

- b1. Perform problem analysis from written descriptions; derive requirements specifications from an understanding of problems (analysis, synthesis).
- b2. Break a large problem into smaller parts, writing each part as a module or function.
- b3. Generate an innovative design to solve a problem containing a range of commercial and industrial constraints.
- b4. Analyze the requirements of a range of computer-based systems and examine the design alternatives based on the constraints imposed by society, organizations, and technology.
- b5. Work with and model computer systems at different and appropriate levels of abstraction.
- b6. Enhance ability to approach problems systematically.

Professional Skills

- c1. Use appropriate programming languages and tools, and design methodologies.
- c2. Interpret verbal problem specifications into program code
- c3. Design, write and debug computer programs in C++ language

General Skill

- d1. Communicate effectively by oral, written and visual means.
- d2. Develop interpersonal skills planning and managing personal time and work.
- d3. Work effectively as an individual and as a member of a team.
- d4. Manage tasks and resources.
- d5. Manage ones own learning and development, including time management and organizational skills.

Course Contents

1 - Computer definition and different computer types.

2 - Data representation.

3 - Introduction to Algorithms, problem-solving and software development methodology.

4 - Basic programming in C++, essential concepts, programming style, variables and data types, long, float, double, Boolean, character, strings, enumeration, I/O format,

5 - Logical expressions and control constructs: if-else, nested if, switch, arithmetic and assignment statements;

6 - Looping techniques, for, while, do-while, nested loop, functions, including parameter passing mechanisms, scope, and return values, pointers

7 - Introduction to structures and arrays.

Teaching and Learning Methods

1 - Lectures
2 - Tutorial Exercises
3 - Practical Exercises
4 - Projects

Students Assessment

Assessment Method	TIME	MARKS
Final Exam	Week 16	50%
Practical Exercises		15%
Mid-Term Exam	Week 8	20%
Projects		15%

Books and References

Course note	Short course notes available at doctor’s office.
Essential books	H. M. Deitel. P. J. Deitel, “C How To Program”, Prentice Hall 2004, Fourth Edition.
Recommended books	J. Hanley, E. Koffman, Problem Solving & Program Design In C, Pearson Education, International of 7th edition (May 1, 2012) Brian W. Kernighan, Dennis M. Ritchie, The C programming language, 2nd edition, 1988.

Knowledge and Skills Matrix

Main Course Contents	Study Week	Knowledge and Understanding	Intellectual Skills	Professional Skills	General Skill
Computer definition and different computer types.	1	a2			d1-d5
Data representation.	2-3	a2,a4			d1-d5
Introduction to Algorithms, problem-solving and software development methodology.	4	a1,a3,a4,a5,a6	b1,b4	c1-c3	d1-d5
Basic programming in C++, essential concepts, programming style, variables and data types, long, float, double, Boolean, character, strings, enumeration, I/O format,	5-6	a8, a9, a12	b1-b5	c1-c3	d1-d5
Logical expressions and control constructs: if-else, nested if, switch, arithmetic and assignment statements.	7-8	a7,a8,a11	b2, b3, b5	c1-c3	d1-d5
Looping techniques, for, while, do-while, nested loop, functions, including parameter passing mechanisms, scope, and return values, pointers.	9-11	a7,a8,a9,a11	b2, b3, b4, b5	c1-c3	d1-d5
Introduction to structures and arrays.	12-14	a8, a10, a11	b2-b6	c1-c3	d1-d5