Course Specifications

Institution:	**College of Science at Az Zulfi**
Academic Department :	**Computer Science and Information**
Programme :	Computer Science and Information
Course :	**Programming 1**
Course Coordinator :	**Noureldin Laban**
Programme Coordinator :	Dr Yossry Azzam
Course Specification Approved Date :	22 / 12 / 1435 H

A. Course Identification and General Information

1 - Course title : : Programming 1	Course Code: CSI 211
2. Credit hours : **3** (2 Lecture + 2 Lab)	
3 - Program(s) in which the course is offered: CSI	
4 – Course Language : English	
5 - Name of faculty member responsible for the course: **Noureldin Laban**	
6 - Level/year at which this course is offered : **3rd level**	
7 - Pre-requisites for this course (if any) : • Computer Skills (PCOM 113)	
8 - Co-requisites for this course (if any) : • **N/A**	
9 - Location if not on main campus **:** • **N/A**	

10 - Mode of Instruction (mark all that apply)

A - Traditional classroom	√	What percentage?	**80 %**
B - Blended (traditional and online)	√	What percentage?	**10 %**
D - e-learning	√	What percentage?	**10 %**
E - Correspondence		What percentage?	……. %
F - Other		What percentage?	…….%

Comments :

B Objectives

What is the main purpose for this course?

This course introduces the student to object-oriented programming through a study of the concepts of program specification and design, algorithm development, and coding and testing using a modern software development environment. Students learn how to write programs in an object-oriented high level programming language. This course introduces the students to basic programming concepts and constructs. Topics include: control structures, functions, recursion, arrays, pointers and strings of the C++ programming language. The course introduces students to structured, top-down programming design and implementation. This course should serve as a foundation for programming to the students in the program.

The purpose of this course is to:

1. Provide students with the ability to develop C++ using pseudo code and flow chart and structured programming design (design, write, debug, comment and modify a C++ program).
2. Acquaint students the ability to   understand the concept of data types, variables and assignments.
3. Introduce students to the object oriented environment.

4. Enable students to be efficient in their work.

Briefly describe any plans for developing and improving the course that are being implemented :

1. Using group discussion
2. Updating the materials of the course to cover the new topics of the field.
3. Motivate students to solve practical problems using C++ programming language.

C. Course Description

1. Topics to be Covered

List of Topics	No. of Weeks	Contact Hours
1. Introduction to computers and C++: Introduction, Computers and the Internet in Industry and Research, Hardware and Software, Moore's Law, Computer Organization, Data Hierarchy, Machine Languages and Typical C++ Development Environment.	2	8
2. Introduction to C++ Programming: First Program in C++ , Input / Output and Operators, Memory Concepts ,Arithmetic, Decision Making: Equality and Relational Operators	2	8
3. Introduction to Classes, Objects and Strings: Defining a Class with a Member Function, Defining a Member Function with a Parameter, Data Members, set Member Functions and get Member Functions, Initializing Objects with Constructors.	2	8
4. Control Statements (Part 1): If Selection Statement, if…else Double-Selection Statement, while Repetition Statement, Counter-Controlled Repetition, Assignment Operators, Increment and Decrement Operators .	3	12
5. Control Statements (Part 2): For Repetition Statement, do…while Repetition Statement, switch Multiple-Selection Statement, break and continue Statements, Logical Operators.	2	8
6. Functions and an Introduction to Recursion: Function Definitions with Multiple Parameters, Function Prototypes and Argument Coercion, Scope Rules, Function Call Stack, Inline Functions, References and Reference Parameters, Default Arguments, Function Overloading, Function Templates, Recursion.	2	8
7. Array and vector: Arrays, Declaring arrays, Examples Using arrays, declaring an array and using a Loop to initialize the array's Elements, Initializing an array in a Declaration with an Initializer List.	1	4
8. Pointers: Pointer Variable Declarations and Initialization, Pointer Operators, Pass-by-Reference with Pointers.	1	4

2. Course components (total contact hours and credits per semester):

	Lecture	Tutorial	Laboratory	Practical	Other:	Total
Contact Hours	30	30	-	-	-	60
Credit	30	15	-	-	-	45

3. Additional private study/learning hours expected for students per week.

5 Hours

The total workload of the student in this course is then: 60 + 5 x 15 = 135 work hours.

4. Course Learning Outcomes in NQF Domains of Learning and Alignment with Assessment Methods and Teaching Strategy

		NQF Learning Domains And Course Learning Outcomes	Course Teaching Strategies	Course Assessment Methods
1.0	**Knowledge**			
1.1		Construct error free C++ programs.	Lectures. Lab demonstrations. Case studies. Individual presentations.	Written Exam Homework assignments Lab assignments Class Activities Quizzes
1.2		Divide a problem into its logical components.		
1.3		Design and code small to medium sized problems from the start using C/C++ constructs, such as input/output statements, if-then-else statements, while and for loops, functions.		
2.0	**Cognitive Skills**			
2.1		Apply knowledge of computing and mathematics appropriate to the discipline.	Lectures. Lab demonstrations. Case studies. Individual	Written Exam Homework assignments Lab
2.2		Design, implement and evaluate a computer-based system, process, component, or program to meet desired needs.		

		NQF Learning Domains And Course Learning Outcomes	Course Teaching Strategies	Course Assessment Methods
			presentations. Brainstorming.	assignments Class Activities Quizzes
3.0		**Interpersonal Skills & Responsibility**		
3.1		Analyze a problem, and identify and define the computing requirements appropriate to its solution.	Small group discussions. Whole group discussions. Brainstorming. Presentations.	Written Exam Homework assignments Lab assignments Class Activities Quizzes
3.2		Understand professional, ethical, legal, security, and social issues and responsibilities.		
4.0		**Communication, Information Technology, Numerical**		
4.1		Work cooperatively in a small group environment.	Small group discussions. Whole group discussions. Brainstorming. Presentations.	Written Exam Homework assignments Lab assignments Class Activities Quizzes
4.2		Save time and space in each task.		
5.0		**Psychomotor**		
5.1		N/A		

5. Schedule of Assessment Tasks for Students During the Semester:

	Assessment task	Week Due	Proportion of Total Assessment
1	First written mid-term exam	6	15%

2	Second written mid-term exam	12	15%
3	Presentation, class activities, and group discussion	Every week	10%
4	Homework assignments	After Every chapter	10%
5	Implementation of presented programs	Every two weeks	10%
6	Final written exam	16	40%

D. Student Academic Counseling and Support

Office hours: Sun: 8-12, Mon. 8-10, Wed. 8-10
Office call: Sun. 12-1 and Wed 12-1

Email: n.laban@mu.edu.sa

E. Learning Resources

1. List Required Textbooks :
- Deitel and Deitel, " C++: How To Program ", 2010, Prentice Hall, ISBN 978-007351725.

2. List Essential References Materials :
- S. Malik, " C++ Programming: From Problem Analysis to Program Design ", Course Technology, ISBN 061916042X.

3. List Recommended Textbooks and Reference Material :
- **Journal of Science of Computer Programming.**

4. List Electronic Materials :
- **http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-096-introduction-to-c-january-iap-2011/**
- **http://www.cplusplus.com/reference/**

5. Other learning material :
- Video and presentation are available with me

F. Facilities Required

1. Accommodation
- Classroom and Lab, as those that are available at college of science at AzZulfi.

2. Computing resources
- Smart Board

3. Other resources
- **N/A**

G   Course Evaluation and Improvement Processes

1 Strategies for Obtaining Student Feedback on Effectiveness of Teaching:
- Questionnaires (course evaluation) achieved by the students and it is electronically organized by the university.
- Student-faculty management meetings.

2 Other Strategies for Evaluation of Teaching by the Program/Department Instructor :
- Discussion within the staff members teaching the course
- Departmental internal review of the course.

3 Processes for Improvement of Teaching :
- Periodical departmental revision of methods of teaching.
- Monitoring of teaching activates by senior faculty members.
- Training course.

4. Processes for Verifying Standards of Student Achievement
- Reviewing the final exam questions and a sample of the answers of the students by others.
- Visiting the other institutions that introduce the same course one time per semester.
- Watching the videos of other courses by international institutions

5 Describe the planning arrangements for periodically reviewing course effectiveness and planning for improvement :
- Course evaluation
- Exam evaluation
- Improvement plan

Course Specification Approved
Department Official Meeting No (6) Date 22 / 12 / 1435 *H*

Course's Coordinator

Name : **Noureldin Laban**

Signature : ....

Date : 17/ 12 / 1435 *H*

Department Head

Name : .Dr. Yossry Azzam..

Signature : ....

Date : 22 / 12 / 1435 *H*