



**Jerash University**  
**Faculty of Computer Science and Information Technology**  
**Computer Sciences Department**

**Semester:** Fall Semester 2018/2019

<b>Course symbol and number:</b> 1001131	<b>Course Name:</b> البرمجة الكينونية
<b>Teaching Language:</b> English	<b>Prerequisites:</b> 1001130.
<b>Credits:</b> 3 hours.	<b>Course Level:</b> 100

### Course Description

This course is an introductory course to the Object Oriented Design. Topics covered include the C++ programming concepts, structures, functions, objects and classes, Class members: Data members (fields) and member functions (methods). Class member visibility (private, public, protected). Class hierarchies. Single and multiple inheritance. Inter-class relationships. Class variables and instance variables. Class methods and instance methods. Service methods and support methods. Scope. Constructors and destructors. Object initialization. Memory management. Garbage collection. Methods and messages. Method signatures. Method and operator overloading. Method overriding. Abstract classes. Dynamic (late) binding. Polymorphism. Software reuse. Subclasses (derived classes). Super classes (base classes). Invocation of super class methods and constructors.

### Course Objectives

Upon completion of this course, students will be able to do the following:

- Explain the motivation for and development of object oriented programming languages.
- Understand Object-Oriented Programming concepts and techniques.
- Understand the principles of software engineering in Object-Oriented languages.
- Understand the fundamentals of programming in C++.
- Learn how to create a class (Class Definition; Accessing Data Members and Member Functions. Constructors and Destructors).
- Learn about Constant Objects and Constant Member Functions. Friend Functions and Friend Classes.
- Illustrate Inheritance: Base Class and Derived Class.

- Using Member Functions; Overriding Base Class Members in a Derived Class.
- Public, Private, and Protected Inheritance.
- Learn about Virtual Functions, Polymorphism, Overloading and Overriding.
- Learn about generic function, generic class, and handling Exceptions.

### Learning Outcomes

Upon completion of this course, students should be able to:

Be able to design and implement object-oriented programs in C++, using a large sub-set of the language effectively.

- Have knowledge of key object-oriented principles and design strategies.
- Be able to use programming tools such as an integrated development environment (IDE), debugger, and code repository.

### Text Book(s)

<b>Title</b>	C++ Programming: From Problem Analysis to Program Design
<b>Author(s)</b>	D. S. Malik
<b>Publisher</b>	Thomson
<b>Year</b>	2010
<b>Edition</b>	Fifth Edition

### References

<b>Books</b>	<ol style="list-style-type: none"> <li>1. C++ How to Program, 9th edition, Deitel &amp; Deitel, Prentice-Hall, 2013</li> <li>2. C++ common knowledge : essential intermediate programming/ C++ (Computer program language) , Dewhurst, Stephen C. Addison-Wesley, Upper Saddle River, N. J.: 2005.</li> <li>3. C++ programming cookbook Herb Schildt's C++ programming cookbook / C++ (Computer program language) , Schildt, Herbert. McGraw-Hill, New York: c2008.</li> <li>4. Problem solving with C++: The object of programming / C++ (Computer program language) . Savitch, Walter. Pearson Addison Wesley, Boston: 2005. Fifth Edition (International ed. )</li> </ol>
<b>Internet links</b>	<a href="http://www.jpu.edu.jo/lms">http://www.jpu.edu.jo/lms</a>
<b>Course link</b>	<a href="#">Click here</a>

### Instructors

<b>Instructor</b>	Dr. Mohammed M. Abu Shquier
<b>Office Location</b>	الطابق السابع – 720
<b>Office Phone</b>	555
<b>E-mail</b>	<a href="mailto:Shquier@jpu.edu.jo">Shquier@jpu.edu.jo</a>

<b>Topics Covered</b>			
<b>Topics</b>	<b>Chapters in Text</b>	<b>Week number</b>	<b>Teaching hours</b>
<b>Introduction to C++ Applications:</b> <ul style="list-style-type: none"> <li>- Introduction</li> <li>- Simple Program: Printing a Line of Text</li> <li>- Another Simple Program: Adding Integers</li> <li>- Memory Concepts</li> <li>- Arithmetic</li> <li>- Equality and Relational Operators</li> <li>- Control Structures.</li> </ul>	Chapters 3, 5,6		
<b>Methods:</b> <ul style="list-style-type: none"> <li>- Method Definitions</li> <li>- Argument Promotion</li> <li>- C++ Namespaces</li> <li>- Value Types and Reference Types</li> <li>- Passing Arguments: Pass-by-Value vs. Pass-by-Reference</li> <li>- Random-Number Generation</li> <li>- Duration of Variables</li> <li>- Scope of declaration</li> <li>- Method Overloading.</li> </ul>	Chapter 7		
<b>Object-Based Programming– Part1:</b> <ul style="list-style-type: none"> <li>- Introduction</li> <li>- Controlling Access to Members</li> <li>- Initializing Class Objects: Constructors</li> <li>- Using Overloaded Constructors</li> <li>- Using the this Reference</li> <li>- Static Class Members</li> <li>- Read only Members.</li> </ul>	Chapter 4		
<b>Object-Based Programming – Part2:</b> <ul style="list-style-type: none"> <li>- Composition: Objects References as Instance Variables of Other Classes</li> <li>- Data Abstraction and Information Hiding</li> <li>- Software Reusability</li> </ul>	Chapter 4		
<b>Arrays and Arrays of objects:</b> <ul style="list-style-type: none"> <li>- Declaring and Allocating Arrays</li> <li>- Examples Using Arrays</li> <li>- for each statement</li> <li>- Passing Arrays to Procedures</li> <li>- Passing Arrays: By Value vs. By Reference</li> </ul>	Chapter 8		
<b>Inheritance and composition:</b> <ul style="list-style-type: none"> <li>- Introduction</li> <li>- Base Classes and Derived Classes</li> <li>- protected and internal Members</li> <li>- Relationship between Base Classes and Derived Classes</li> </ul>	Chapter 11		
<b>Polymorphism:</b> <ul style="list-style-type: none"> <li>- Introduction</li> <li>- Derived-Class-Object to Base-Class-Object Conversion</li> <li>- Abstract Classes and Methods</li> <li>- Operator Overloading</li> </ul>	Chapter 12		

<b>Exception Handling:</b> <ul style="list-style-type: none"> <li>- Exception handling importance</li> <li>- Options for exception handling.</li> <li>- Exception Handling Using Multiple catch statements.</li> <li>- Catching all exceptions</li> </ul>	Chapter 13		
<b>Files Programming: Files Programming:</b> <ul style="list-style-type: none"> <li>- Class Sequential File</li> <li>- Class Random File and</li> <li>- Final Exam review</li> </ul>	Chapter 14		

<b>Evaluation</b>		
<b>Assessment Tool</b>	<b>Expected Due Date</b>	<b>Weight</b>
Programming assignments and LMS		20 %
First Exam		20 %
Second Exam		20 %
Final Exam	According to the University final examination schedule	40 %

<b>Policy</b>	
<b>Attendance</b>	Attendance is very important for the course. In accordance with university policy, students missing more than the allowed absence rate of total classes are subject to failure. Penalties may be assessed without regard to the student's performance. Attendance will be recorded at the beginning or end of each class.
<b>Exams</b>	All exams will be CLOSE-BOOK; necessary algorithms/equations/relations will be supplied as convenient.

<b>Class Schedule &amp; Room</b>
----------------------------------

<b>Office Hours</b>
Sun: 8 - 9 Mon: 8 - 9:30 Tues: 11- 12 Wed: 11 – 12:30
* Or by an appointment through email

<b>Teaching Assistant</b>	
To announced later on.	
<b>Prerequisites</b>	
<b>Prerequisites by course</b>	1001130