ABET Courses Syllabi

For the

Computer Science Program

At

College of Computing - Alqunfudah
Umm Al-Qura University

Makkah, Alqunfudah, Saudi Arabia

Feb 2023

Computer Programming Syllabus

1. Course number and name

60011101-4 Computer Programming

2. Credits and contact hours

Credits Hours	4
Contact Hours	Theoretical contact hours (3) - Practical contact hours(3)

3. Course coordinator

Musab alzaghol

4. Textbooks/References

- Intro to Java Programming, Comprehensive Version, 10th Edition, Y. Daniel Liang, Pearson, 2014, ISBN-13: 978-0133761313.
- Java How to Program (Early Objects), 10th Edition, Dietel and Dietel, Pearson, 2015, ISBN-13: 978-0133807806.
- Absolute Java, 6th Edition, Walter Savitch, Pearson, 2016, ISBN-13: 978-0134041674.
- Java Programming, 8th Edition, Joyce Farrell, 2016, ISBN-13: 9781285856919.

5. Specific course information

a. Catalog Description

This course introduces the basic concepts of computer programming to students with some problem solving skills. Students will be using Java, a high-level programming language, to learn the fundamentals of computer programming including how to write, compile, and run programs in an IDE. Topics include algorithms and problem solving, variables and data types, methods, console input/output, control structures, arrays, Java best coding styles, and the mechanics of running, testing, and debugging.

b. Prerequisites

Passing first joint year

c. Course Type

Required

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Analyze and design a computational solution to a problem described in natural language and express the solution in an algorithmic way. 2. Convert the algorithmic solution effectively and intelligibly into a computer program. 3. Correctly use Java syntax and semantics to write error-free code and best coding style to enhance code readability and maintainability. 4. Properly use modularity (methods), repetitions, conditionals, and simple data structures (arrays) in problem solutions.

7. Brief list of topics to be covered

Problem solving concepts
Introduction to Java: Java history and features, types of Java programs, JDK and IDE, Java translation, Java program structure, programming errors, programming styles and documentation, and how to edit, compile, and execute Java programs
Data types, variables, arithmetic expressions, assignments statements, naming constants, data type conversions, Java literals, and the **String** type
Console input/output
Selection structure (**if**, **if-else**, and **switch-case**)
Repetition structure (**for**, **while**, and **do-while**)
Static methods: declaring and calling methods, call by-value/reference, **void** and **return** type methods, method signature, and method overloading

Discrete Structures I Syllabus

1. Course number and name

60011801-3 Discrete Structures I

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Saad Albogmi

4. Textbooks/References

- Discrete Mathematics and Its Applications, 7th Edition, By Kenneth Rosen, Pearson, ISBN-13: 978-0073383095.
- Essentials of Discrete Mathematics, 3rd Edition, By David J. Hunter, ISBN-13: 9781284056242.

5. Specific course information

a. Catalog Description

This is an introductory course in discrete mathematics. The goal of this course is to introduce students to ideas and techniques from discrete mathematics that are widely used in computer science and computer engineering. This course teaches the students techniques in how to think logically and mathematically and apply these techniques in solving problems. To achieve this goal, students will learn logic and proof, sets, sequences and sums, functions, as well as mathematical induction and mathematical reasoning.

b. Prerequisites

Passing first joint year

c. Course Type

Required

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Appreciate the basic principles of Boolean algebra, Logic, and Set theory.
2. Understand logical arguments and logical constructs and have a better understanding of sets and functions.
3. Use mathematical reasoning techniques including induction and recursion.
4. Communicate the solutions of technical problems to other professionals.

7. Brief list of topics to be covered

Introduction to discrete mathematics
Logic: Propositional logic; Truth tables; Propositional Equivalence; Implication, equivalence, converse, inverse, contrapositive and negation; Predicates and quantifier; Rules of inference
Sets: Venn diagrams; Sets operations; Cartesian product; Power sets; Cardinality of finite sets; Important numeric sets, notation and subset relations among them
Functions: Representation; Surjections, injections and bijections; Inverse; Composition; Important Numeric functions: floor, ceiling, log
Sequence and Sums: Arithmetic and geometric sequences; Geometric Progression and Arithmetic Progression; Summations and Linearity of summation; Sums of powers of integers
Proofs: Direct proofs; Proof by contradiction; Proof by contrapositive; Proof by Case
Induction: Mathematical Induction; Weak and strong induction; Recursive definitions of functions and sequences; Recurrence relation

Digital Logic Design Syllabus

1. Course number and name

60031201-4 Digital Logic Design

---

2. Credits and contact hours

Credits Hours	4
Contact Hours	Theoretical contact hours (3) - Practical contact hours(3)

3. Course coordinator

Bilal Alzobi

4. Textbooks/References

- M. Morris Mano and Michael D. Ciletti, Digital Design, 5th Edition, 2007, Prentice Hall, ISBN-10: 01398926X, ISBN-13: 978-013989269.

5. Specific course information

a. Catalog Description

The objective of this course is to provide fundamental knowledge of digital design. Topics covered include number systems, Boolean algebra, design and analysis of combinational and sequential circuits.

b. Prerequisites

Passing the first joint year

c. Course Type

Required

6. Specific goals for the course

   Course Learning Outcomes (CLOs)

1. Understand number systems and codes. [BL 1, Topics 1, 2] 2. Apply Boolean algebra for optimizing logic circuits. [BL 3, Topics 3] 3. Design combinational logic circuits. [BL 3, Topic 4] 4. Design sequential circuits. [BL 3, Topic 5] 5. Design and conduct experiments in the area of basic digital circuits [BL 3]

7. Brief list of topics to be covered

Introduction to Digital Systems
Number Systems and Codes
Boolean Algebra and Logic Circuits Optimization
Combinational Logic: Design and Analysis
Sequential Logic: Design and Analysis

Discrete Structures II Syllabus

1. Course number and name

60011802-3 Discrete Structures II

---

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Fatemah Albarqi

4. Textbooks/References

- Discrete Mathematics and Its Applications, 7th Edition, By Kenneth Rosen, Pearson, ISBN-13: 978-0073383095.
- Essentials of Discrete Mathematics, 3rd Edition, By David J. Hunter, ISBN-13: 9781284056242.

5. Specific course information

a. Catalog Description

Discrete Structures (II) together with Discrete Structures (I) is one of the core components of mathematics at the undergraduate level. The main goal of the sequence is that students obtain those skills in discrete mathematics and logic that are used in the study and practice of computer science. This course introduces and studies (with an emphasis on problem solving) several of the main areas of discrete mathematics, which provide important knowledge and skills for the applied scientists, such as number theory, advance counting, graph theory, and trees to solve real world problems.

b. Prerequisites

60011801-3 Discrete Structures I
60011101-4 Computer Programming

c. Course Type

Required

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Define statements and solve problems involving divisibility, congruence, greatest common divisor, prime numbers, and Euclidean algorithm.
2. Solve counting problems by applying elementary counting techniques such as the product and sum rules, permutations, combinations, the pigeon-hole principle, and binomial expansion.
3. Describe binary relations between two sets; determine if a binary relation is reflexive, symmetric, or transitive or is an equivalence relation; combine relations using set operations and composition.
4. Represent a graph using an adjacency list and an adjacency matrix and apply graph theory to application problems such as computer networks.

7. Brief list of topics to be covered

Discrete mathematics (I) review
Number Theory: Divisibility and Modular Arithmetic; Integer Representations and Algorithms; Primes and Greatest Common Divisors; Solving Congruences; Applications of Congruences
Counting: Introduction to Counting; Factorials and Permutations; Permutation Practice; Combinations; Binomial Theorem, Pascal's Triangle; Combinations with Repetition Permutations and Combinations Practice
Relations: Relations and Their Properties; n-ary Relations and Their Applications; Representing Relations; Closures of Relations; Equivalence Relations; Partial Orderings
Graphs: Graphs and Graph Models; Graph Terminology and Special Types of Graphs; Representing Graphs and Graph Isomorphism; Connectivity; Euler and Hamilton Paths; Shortest-Path Problems; Planar Graphs; Graph Coloring
Trees: Introduction to Trees; Applications of Trees.

Object Oriented Programming Syllabus

1. Course number and name

60011102-4 Object Oriented Programming

---

2. Credits and contact hours

Credits Hours	4
Contact Hours	Theoretical contact hours (3) - Practical contact hours(3)

3. Course coordinator

Mohammad Al mojamed

4. Textbooks/References

- Intro to Java Programming, Comprehensive Version, 10th Edition, Y. Daniel Liang, Pearson, 2014, ISBN-13: 978-0133761313.
- Java How to Program (Early Objects), 10th Edition, Dietel and Dietel, Pearson, 2015, ISBN-13: 978-0133807806.
- Absolute Java, 6th Edition, Walter Savitch, Pearson, 2016, ISBN-13: 978-0134041674.
- Java Programming, 8th Edition, Joyce Farrell, 2016, ISBN-13: 9781285856919.

5. Specific course information

a. Catalog Description

This course introduces students to the object-oriented programming, the most widely used programming paradigm, focusing on using classes and objects to design and build robust console applications. Students will learn to design larger programs by structuring them into multiple classes, with a variety of relationships between those classes, like association, aggregation, composition, generalization, and realization. Students will use UML notations to model classes and their relationships. The core concepts of object-oriented programming, including inheritance, encapsulation, abstraction, and polymorphism will be studied in detail with regular practical hands-on software development activities.

b. Prerequisites

60011101-4 Computer Programming

c. Course Type

Required

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Use and explain the object oriented concepts of instantiation, encapsulation, inheritance, abstraction, and polymorphism. 2. Design and construct moderately complex applications using the advanced OO features such as interface, exception handling, and generics. 3. Use object-oriented design notations (UML class diagrams and relationships) to model problem solutions. 4. Use library classes, generic lists, and iterators in problem solutions.

b. ABET-Enabled Student Outcomes:

- SO2: Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.

- SO2 will be evaluated through CLO3.

7. Brief list of topics to be covered

Java and Object Oriented programming overview
Object Oriented Programming Concepts: Classes and Objects, attributes, methods, constructors, methods overloading attributes scope (global and local variables) and using this
Java Inheritance: classes inheritance, methods overriding, using super
Polymorphism, Interfaces, abstract classes, abstract methods
Extra Object Oriented Programming Concepts : Modifiers and static, packages, overriding utility methods(toString, equals and compareTo) Objects comparison and parameters passing
Exception handling(try…catch) and Java API examples
UML(Use case diagram, Sequence diagram, Class diagram)

Computer Organization and Architecture Syllabus

1. Course number and name

60032205-4 Computer Organization and Architecture

2. Credits and contact hours

Credits Hours	4
Contact Hours	Theoretical contact hours (3) - Practical contact hours(3)

3. Course coordinator

Hamzah Awwad Allah

4. Textbooks/References

- D. Patterson and J. Hennessy, Computer Organization and Design, The Hardware/Software Interface, 5th Edition, Morgan Kaufmann (MK), 2013
- D. Harris and S. Harris, Digital Design and Computer Architecture, Morgan Kaufmann (MK), 2nd Edition, 2012

5. Specific course information

a. Catalog Description

Instruction-set architecture (ISA), Implementations of ISA (datapath and controller), Pipelined implementation, Hazards and solutions, Memory hierarchy design, I/O interfacing.

b. Prerequisites

60031201-4 Digital Logic Design

c. Course Type

Required

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Use MIPS instruction set architecture for assembly language programming [BL 3, Topic 1, 2]
2. Implement MIPS instruction set architecture by building data path and controller [BL 3, Topic 3]
3. Apply the concepts of pipelining to MIPS data path and control [BL 3, Topics 4, 5]
4. Understand and analyze memory hierarchy [BL 4, 5, Topics 6, 7]
5. Understand I/O and storage devices [BL 1, Topic 8]

6. Highlight the challenges/issues of modern computer architectures through a technical report/presentation* [BL 4, 5]
7. Write assembly language programs using MIPS assembly language [BL 3]

7. Brief list of topics to be covered

Introduction to computer organization and architecture
Instruction-set architecture (ISA) for MIPS
Implementation of ISA (datapath and controller)
Pipelined datapath and controller
Hazards in pipelining and remedies
Introduction to memory hierarchy design
Cache and virtual memory design

Numerical Methods for Computing Syllabus

1. Course number and name

60032401-4 Numerical Methods for Computing

2. Credits and contact hours

Credits Hours	4
Contact Hours	Theoretical contact hours (3) - Practical contact hours(3)

3. Course coordinator

Abdulkareem Abdulqader

4. Textbooks/References

- Steven C. Chapra and Raymond P. Canale, Numerical Methods for Engineers, 7th Edition, McGraw-Hill, 2015

5. Specific course information

a. Catalog Description

Theory of key concepts on equation solving, curve fitting, numerical integration and differentiation and the solution of differential equations are introduced with the computer implementation using MATLAB.

b. Prerequisites

28042402-4 Linear Algebra I
60011101-4 Computer Programming

c. Course Type

Required

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Apply Taylor Series to approximate functions [BL 3, Topic 1]
2. Apply various algorithms to locate the roots of equations [BL 3, Topic 2]
3. Solve problems involving linear algebraic equations [BL 3, Topic 3]
4. Apply least squares method and polynomials [BL 3, Topics 4, 5]
5. Solve numerical differentiation and integration problems [BL 3, Topics 6, 7]
6. Solve ordinary differential equations [BL 3, Topic 8]
7. Use MATLAB to solve various numerical problems [BL 3, Topics 1 to 8]

7. Brief list of topics to be covered

Introductory material (Absolute and relative errors, Rounding and chopping, Computer errors in representing numbers, Review of Taylor series
Locating roots of algebraic equations
Systems of linear equations
The Method of Least Squares
Interpolation
Numerical Integration
Numerical Differentiation
Ordinary Differential Equations

Data Structures Syllabus

1. Course number and name

60012401-3 Data Structures

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(3)

3. Course coordinator

Musab Alzaghol

4. Textbooks/References

- Data structures and algorithms in Java, 6th Edition, Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, 2014, ISBN-13: 978-1118771334.
- Data structures and problem solving using Java, 4th Edition, Mark Allen Weiss, 2009, ISBN-13: 978-0321541406.
- Data Structures and Algorithms in Java, 2nd Edition, Robert Lafore, 2003, ISBN-13: 978-0672324536.
- Object-Oriented Data Structures Using Java, 4th Edition, Nell Dale, Daniel T. Joyce, Chip Weems, 2016, ISBN-13: 978-1284089097.

5. Specific course information

a. Catalog Description

This course provides theoretical and practical knowledge of fundamental data structures. Topics covered include arrays, linked-list, stacks, queues, trees and hash tables. These data structures are explained using basic sorting and searching techniques with brief overview of recursion and memory management. The course also explores the implementation of a range of data structures in the Java programming language. The knowledge and practice of these structures are of utmost importance. It will make the students able to organize, represent and manipulate the data, which is central to computing.

b. Prerequisites

60011102-4 Object Oriented Programming

c. Course Type

Required

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Describe, compare and use fundamental data structures: lists, stacks, queues, trees, and heaps.
2. Evaluate the efficiency of a data structure and its implementation
3. Design and implement and choose the most appropriate data structure and for a

7. Brief list of topics to be covered

Arrays revision
Ordered arrays and binary search
Complexity analysis and Big Oh Notation
Sorting Algorithms
Stack and Queues
Linked Lists
Recursion
Trees and Binary search trees
Balanced Trees
Priority Queues and Heaps
Hashing tables and Java Collections

Operating Systems Syllabus

1. Course number and name

60012203-4 Operating Systems

2. Credits and contact hours

Credits Hours	4
Contact Hours	Theoretical contact hours (4)

3. Course coordinator

Aziz Alshehri

4. Textbooks/References

- Operating System Concepts, by Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, 9th Edition, John Wiley & Sons, 2012. ISBN-13: 978-8126554270.
- Operating Systems: Internals and Design Principles, by William Stallings, 9th Edition, Pearson, 2017. ISBN-13: 978-0134670959.

5. Specific course information

a. Catalog Description

This course introduces students to the basic concepts, objectives, services, and structures of multitasking systems and the foundations of operating systems design and implementation. The course covers the major components of modern operating systems, such as processes and threads management, CPU scheduling, synchronization, memory management (segmentation, paging, swapping), virtual memory, file systems, I/O system, and operating system protection and security.

b. Prerequisites

60011102-4 Object Oriented Programming

c. Course Type

Required

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Describe the major components of operating systems, the services operating systems provide, and the various ways of structuring an operating system.
2. Describe the representation of processes and threads, and the concepts of interprocess communication, process synchronization, and algorithms for process scheduling.
3. Understand and apply memory management concepts including paging, segmentation, address mapping, and page replacement algorithms.
4. Describe the structure and organization of file systems, I/O system and the security and protection issues in computer systems.
5. Computational problem-solving using operating system design and implementation techniques.

b. ABET-Enabled Student Outcomes:

- SO2: Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.

- SO2 will be evaluated through CLO2 & CLO3.

7. Brief list of topics to be covered

Introduction to Operating Systems
Operating Systems Services and Structures
Processes and Interposes Communication
Thread
Process Synchronization and Deadlocks
CPU scheduling
Memory management
Virtual memory
File and I/O systems
Protection and security

Database I Syllabus

1. Course number and name

60012301-3 Database I

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Aljawhrah Alzahrani

4. Textbooks/References

- Fundamentals of Database Systems, 7th Edition, Ramez Elmasri, Shamkant B. Navathe, 2015, ISBN-13: 978-0133970777
- Modern Database Management, 12th Edition, Jeffrey A. Hoffer, Ramesh Venkataraman, Heikki Topi, 2015, ISBN-13: 978-0133544619.

5. Specific course information

a. Catalog Description

This course provides foundational database knowledge and covers topics related to the conceptual design of database based on the functional requirements for organizations. It presents the basics of information storage and management, from the conceptual modelling of an organization's data requirements using the relational model, through to the implementation of these requirements with tools such as SQL and techniques such as normalization.

b. Prerequisites

60011102-4 Object Oriented Programming

c. Course Type

Required

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Understand different types of data modeling techniques and the supporting theoretical foundation.
2. Generate, retrieve data, and create database using query language.

3. Understand a variety of techniques for designing database schemas, associated index structures, and design and implementation of a database system.

b. ABET-Enabled Student Outcomes:

- SO1: Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
- SO6: Apply computer science theory and software development fundamentals to produce computing-based solutions.

- SO1 will be evaluated through CLO1.
- SO6 will be evaluated through CLO3.

7. Brief list of topics to be covered

File Systems VS DB Systems
Data Modelling: Entity Relationship Diagram
ERD to Relational Mapping
Normalization: 1NF, 2NF, 3NF, BCNF
Relational Algebra Operations
SQL: Data Definition Language
SQL: Data Manipulation Language
Indexing and Stored Procedures

Compilers Construction Syllabus

1. Course number and name:
60012109-3 Compilers Construction

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (3)

3. Course coordinator

Essa Muharish

4. Textbooks/References

- Compilers: Principles, Techniques, and Tools, 2nd Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, 2006, ISBN-13: 978-0321486813.

5. Specific course information
   a. Catalog Description

This course provides a wide overview of compilers constructions including lexical, syntax, and semantic analysis and other phases of compilation process. Topics covered include Grammar, parse tree, and ambiguous grammar, Translation schemes, Contextfree grammar & parsing and finite automata. Students will develop and test a simple compiler to translate infix mathematical expression to postfix one.

   b. Prerequisites

60012401-3 Data Structures

   c. Course Type

Required

6. Specific goals for the course
   Course Learning Outcomes (CLOs)

1. Understand the organization of a compiler.
2. Understand the concepts of scanning, parsing, and translation.
3. Understand the compiler writing tools.

7. Brief list of topics to be covered

Introduction to compilers structure & goals
Arithmetic expression processing using a stack
Simple compiler structure
Grammar, parse tree, and ambiguous grammar
Translation schemes
Context-free grammar & parsing
Introduction to left recursion and right recursion
Lexical analyzer (language, errors, pattern specifications)
Operations on languages and regular expressions
Finite automata
Parsers and errors and sentential error
Left recursion and left factoring
FIRST, FOLLOW, and transition diagrams

Computer Networks Syllabus

1. Course number and name
60033103-4 Computer Networks

2. Credits and contact hours

Credits Hours	4
Contact Hours	Theoretical contact hours (3) - Practical contact hours(3)

3. Course coordinator

Nawaf Mirza

4. Textbooks/References

- Computer Networking, a Top-Down Approach; J. F. Kurose, K. W. Ross (7th Edition)

5. Specific course information
   a. Catalog Description

   The objective of this course is to provide fundamentals of computer networks. Topics covered include Network Architectures and the ISO/OSI and TCP/IP Layers, Application layer, Transport layer, Network layer, Link layer, An overview of wireless and mobile networking.

   b. Prerequisites

   60011802-3 Discrete Structures II

   c. Course Type

   Required

6. Specific goals for the course
   a. Course Learning Outcomes (CLOs)

   1. Explain the concept of packet switching, and identify and analyze the different types of packet delays in packet-switched networks. [BL 2, Topic 1]
   2. Explain application layer protocols such as HTTP, FTP, DNS etc. [BL 2, Topic 2]
   3. Explain the essential principles of a transport layer protocol such as reliable data transfer, flow control and congestion control. [BL 2, Topic 3]
   4. Use IP addressing and apply routing algorithms to find shortest paths for network layer packet delivery. [BL 3, Topic 3]
   5. Design a network by addressing and subnetting. [BL 5, Topic 4]
   6. Apply error detection and correction algorithms on data link layer. [BL 3, Topic 5]

7. Develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions in Computer Networks. [BL 3, Lab]

b. ABET-Enabled Student Outcomes:

- SO5: Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.
- SO5 will be evaluated through CLO7.

7. Brief list of topics to be covered

Introduction: Network Architectures and the ISO/OSI and TCP/IP Layers
Data link layer: Error Detection and Correction; Framing; Design of Data-link protocols; Multiple Access, Controlled Access, Channelization;
Network Layer: IPV4; IPV6; Fragmentation; Transition from IPv4 to IPv6;
Transport Layer: Process-to-Process delivery; UDP; TCP; SCTP; Congestion Control, QoS
Application Layer: Domain Name System; Domain Name Space; DNS in the Internet; Mapping names to Addresses and vice versa; DDNS; SNMP; Multimedia
Delivery, Forwarding, Routing and routing tables, Unicast and Multicast Routing Protocols
An overview of Wireless and Mobile Networking

Algorithms Syllabus

1. Course number and name

60012402-4 Algorithms

2. Credits and contact hours

Credits Hours	4
Contact Hours	Theoretical contact hours (3) - Practical contact hours(3)

3. Course coordinator

Mehinaz Khan

4. Textbooks/References

- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, "Introduction to Algorithms", 3rd Edition, MIT Press, 2009.
- Robert Sedgewick and Kevin Wayne, "Algorithms", 4th Edition, Addison-Wesley Professional, 2011.

5. Specific course information
   a. Catalog Description

The objective of this course is to study design and analysis of algorithms. Learning different algorithms design strategies such as divide and conquer, dynamic programming, and greedy approach. Applications involve: sorting and searching, trees/graph, maximum flow algorithms, and string matching algorithms. Analysis of algorithms is essential part of this course. Study worst case, average case, and amortized analysis- with an emphasis on the close connection between the time complexity of an algorithm and the underlying data structures.

   b. Prerequisites

60012401-3 Data Structures

   c. Course Type

Required

6. Specific goals for the course
   a. Course Learning Outcomes (CLOs)

1. Apply key algorithmic paradigms such as divide and conquer, greedy algorithms into problems.

2. Possess a strong understanding of algorithm analysis to correctly analyze the time and space complexity of a problem.
3. Ability to recognize the intractability for some problems.

b. ABET-Enabled Student Outcomes

- SO1: Analyze a complex computing problem and apply principles of computing and other relevant disciplines to identify solutions.

  SO1 will be evaluated through CLO1.

7. Brief list of topics to be covered

Basic Concepts in Algorithmic Analysis
Searching & Sorting
Recurrence Relations and Recursion
Divide and Conquer
Dynamic Programming
Greedy Approach
String matching

Computer Graphics Syllabus

1. Course number and name

60012501-3 Computer Graphics

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Hafiza Othman

4. Textbooks/References

- Fundamentals of Computer Graphics, 4th Edition, Steve Marschner, Peter Shirley, 2015, ISBN-13: 978-1482229394.
- Computer Graphics with OpenGL, 4th Edition, Donald D. Hearn, M. Pauline Baker, Warren Carithers, 2010, ISBN-13: 978-0136053583.
- 3D Computer Graphics, 3rd Edition, Alan Watt, 1999, ISBN-13: 978-0201398557.

5. Specific course information

a. Catalog Description

The course offers an introduction to computer graphics hardware, algorithms, and software. Topics include overview of graphics hardware, 2D and 3D object representation, geometric transformations, 2D viewing, 3D viewing, illumination models, color modes, and color applications.

b. Prerequisites

28042402-4 Linear Algebra
60011102-4 Object Oriented Programming

c. Course Type

Required

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Understand the foundation of the implementation of computer graphics modeling and rendering systems.
2. Understand the mathematical background of computer graphics.
3. Understand the handling of colors.

4. Write the implementation of a graphics programming project in team.

b. ABET-Enabled Student Outcomes:

- SO5: Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.
- SO5 will be evaluated through CLO4.

7. Brief list of topics to be covered

Mathematical Foundations of Computer Graphics
Graphics APIs and Programming
Computer Graphics Hardware
Modeling and Rendering
Curve and Spline
Transformations
Viewing
Illumination and Color Models

Artificial Intelligence Syllabus

1. Course number and name

60013701-4 Artificial Intelligence

---

2. Credits and contact hours

Credits Hours	4
Contact Hours	Theoretical contact hours (2) - Practical contact hours(3)

3. Course coordinator

Amani Alluhaibi

4. Textbooks/References

- Artificial Intelligence: A Modern Approach, 3rd Edition, Russell & Norvig, 2009, Prentice Hall, ISBN 0136042597.

5. Specific course information

a. Catalog Description

This course introduces basics of Artificial Intelligence, concept of Intelligent agents and various types of agents. It includes various search techniques, Propositional logic and First order logic. It further introduces the concept of knowledge engineering and inference systems.

b. Prerequisites

60012402-4 Algorithms

c. Course Type

Required

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Learn basics of AI, Intelligent Agents and their different types and applications.
2. Learn in detail different search techniques including uninformed search, heuristic search, adversarial search that can be used in Game playing and other AI applications.
3. Learn logical agents, first order logic and first order inference system.

b. ABET-Enabled Student Outcomes

- SO1: Analyze a complex computing problem and apply principles of computing and other relevant disciplines to identify solutions.

  SO1 will be evaluated through CLO3

7. Brief list of topics to be covered

Introduction to AI, History of AI
AI Domains, AI Applications
Intelligent Agents and Environment
Structure of Different type of Agents
Problem Solving through search: i. Un-informed search (BFS, DFS, Depth First, Depth limited and iterative deepening search) ii. Informed Search (Greedy best first search, A* search, Heuristics) iii. Local Search Algorithms (Hill Climbing, Genetic Algorithm)
Adversarial Search (Minimax Algorithm, Alpha Beta Pruning, Chance Minimax)
Logical Agents (knowledge based agents, propositional logic)
First Order Logic
Inference in FOL

Internet Applications Syllabus

1. Course number and name

60013104-3 Internet Applications

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(3)

3. Course coordinator

Khudran Alzhrani

4. Textbooks/References

- HTML5 and CSS3, 8th Edition, ElizabEth Castro, Bruce hyslop, 2013, ISBN: 9780321928832.
- JavaScript: The Definitive Guide, 6th Edition, David Flanagan, 2011, ISBN: 0596805527.
- Java Servlet Programming, 2nd Edition, William Crawford, Jason Hunter, 2001, ISBN: 0596000405.
- Programming the World Wide Web, 8th Edition, Robert Sebesta, 2014, ISBN-13: 9780133775983.

5. Specific course information

a. Catalog Description

This course enables students to develop skills in website development and administration. Students will explore front-end Web technologies such as HTML, CSS and JavaScript. The course will provide the students with the basic concept of building server-side applications using a 3-tier architecture with Servlets and JSP. It will also focus on building complete websites and issues related to user input form-validation.

b. Prerequisites

60011102-4 Object Oriented Programming

c. Course Type

Required

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Learn the basic principles and techniques for building Internet applications.
2. Design and experiment using various Internet programming concepts and components to increase overall understanding of modern Internet design and programming.
3. Design, implement, and evaluate small-scaled dynamic Web applications.

b. ABET-Enabled Student Outcomes

- SO6: Apply computer science theory and software development fundamentals to produce computing-based solutions.

SO6 will be evaluated through CLO2.

7. Brief list of topics to be covered

Introduction to web programming
HTML5
Introduction to CSS
Introduction to JavaScript
Introduction to Servlets Programming
Introduction to JSP and multi-tier applications

Advanced Programming Syllabus

1. Course number and name

60013103-4 Advanced Programming

---

2. Credits and contact hours

Credits Hours	4
Contact Hours	Theoretical contact hours (3) - Practical contact hours(3)

3. Course coordinator

Ahmad Alalwi

4. Textbooks/References

- Intro to Java Programming, Comprehensive Version, 10th Edition, Y. Daniel Liang, Pearson, 2014, ISBN-13: 978-0133761313.
- Java How To Program (Early Objects), 10th Edition, Dietel and Dietel, Pearson, 2015, ISBN-13: 978-0133807806.
- Absolute Java, 6th Edition, Walter Savitch, Pearson, 2016, ISBN-13: 978-0134041674.
- Java Programming, 8th Edition, Joyce Farrell, 2016, ISBN-13: 9781285856919.

5. Specific course information

a. Catalog Description

This course continues the presentation of object-oriented programming begun in 14011102, with more emphasis on graphic user interface development, event-driven programming, generic collections, I/O file management, and database connectivity. Students will acquire the experience of working in groups to design and develop complete GUI-based Java application projects.

b. Prerequisites

60012301-3 Database I

c. Course Type

Required

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Design and develop and demonstrate GUI-based and event-driven interactive programs.

2. Use I/O file management and object persistence to transfer data either to or from a storage medium.
3. Use different types of generic collections available in the standard library and the utility operations of the Arrays and Collections classes.
4. Map java objects to database tables using Hibernate ORM framework.
5. Understanding of SDLC through practical implementation of each phase.

b. ABET-Enabled Student Outcomes

- SO3: Communicate effectively in a variety of professional contexts.
- SO3 will be evaluated through CLO1

7. Brief list of topics to be covered

Java and Object Oriented programming overview
Java Collections
Mapping Java Objects to Tables
Building GUI
Generic Programming
File I/O
Recursion Techniques
Introduction to Design patterns
Java Database Connectivity (JDBC)

Software Engineering I Syllabus

1. Course number and name

60013303-3 Software Engineering I

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Mehnaz Khan

4. Textbooks/References

- Software Engineering: A Practitioner's Approach, 7th Edition, Roger Pressman, 2017, ISBN-13: 978-9339212087.
- Software Engineering, 10th Edition, Ian Sommerville, 2018, ISBN-13: 9789332582699.
- Software Engineering: Principle and Practice, 3rd Edition, Hans Van Vliet, 2008, ISBN-13: 978-0470031469.
- Systems Analysis and Design: An Object-Oriented Approach with UML, 5th Edition, Alan Dennis, Barbara Wixom, David Tegarden, 2015, ISBN-13: 978-1118804674.

5. Specific course information

a. Catalog Description

This course is a detailed introduction to the application of engineering approach to computer software design and development. Topics include: life cycle models, software requirements, specification, conceptual model design, detailed design, validation and verification, design quality assurance, software design/development environments and project management.

b. Prerequisites

60012301-3 Database I

c. Course Type

Required

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Understand the main software engineering concepts, principles, and essential processes of the SDLC.
2. Create UML and DFD artifacts for requirements gathering, analysis as well as design phases using an object-oriented methodology.

3. Design and prototype software from written specifications and/or supplied application libraries.
4. Evaluate proposed solution by validating and verification techniques.

7. Brief list of topics to be covered

Software Engineering overview
Software Engineering Life Cycle (Software Process Activities)
Software specification (Analysis Phase) Requirements Engineering, Information Gathering Techniques, User and System Requirements, Functional and Non- Functional Requirements, and Requirements validation
System Modeling: Engineering OO Software with Unified Modelling Language (UML) Context diagram Use case modelling Scenarios Sequence diagrams Activity diagrams Class analysis and object diagrams State diagrams Communication or collaboration diagram
System Modeling - Structured Analysis, Process Modeling (Data Flow Diagrams), Pseudocode Guidelines for Selecting Appropriate Tools.
Project Management Risk management Managing people Teamwork

Software Engineering II Syllabus

1. Course number and name

60013304-3 Software Engineering II

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Edrees Alkinani

4. Textbooks/References

- Software Engineering, 10th Edition, Ian Sommerville, 2018, ISBN-13: 9789332582699.
- UML Distilled UML Distilled: A Brief Guide to the Standard Object Modeling Language 3rd Edition, Martin Fowler, 2003, ISBN-13: 978-0321193681.
- Requirements Engineering: From System Goals to UML models to Software Specifications, 1st Edition, Axel van Lamsweerde, 2014, ISBN-13: 978-8126545896.
- Software Project Management, 5th Edition, Bob Hughes and Mike Cotterell, 2009.
- Mastering the Requirements Process, 2nd Edition, Suzanne and James Robertson, 2006, ISBN-13: 978-0321419491.

5. Specific course information

a. Catalog Description

The course presents the design and architecture part of software engineering, with emphasis on object oriented modelling and UML techniques for specifying software systems. The objectives of the module are a) introduce agile method for managing software development b) present standard UML software analysis and design c) introduce key object-oriented design principles d) introduce design patterns and demonstrate how they can facilitate the overall development process e) introduce enterprise architecture and SOA.

b. Prerequisites

60013303-3 Software Engineering I

c. Course Type

Required

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Understand and identify the importance of architecture in distributed software systems, styles and patterns, and the design trade-offs involved.
2. Use UML in the analysis, architecture, and design of software systems.
3. Design software systems using best practice design principles.
4. Manage and perform the systems development process more effectively.

b. ABET-Enabled Student Outcomes:

- SO2: Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.

  SO2 will be evaluated through CLO3.

- SO5: Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.

  SO5 will be evaluated through CLO4.

- SO6: Apply computer science theory and software development fundamentals to produce computing-based solutions.

  SO6 will be evaluated through CLO2.

7. Brief list of topics to be covered

Brief Overview of Software Lifecycle Agile Software Development
Engineering OO Software with Unified Modelling Language (UML)
Introduction to Software Architecture and Styles
Introduction to Service-oriented Architecture and Cloud SaaS
Software Maintenance and Quality assurance
CMMI and Process Improvement

Parallel Computing Syllabus

1. Course number and name

60013204-3 Parallel Computing

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Reem Alharbi

4. Textbooks/References

- Distributed Systems: Principles and Paradigms, 2nd Edition, Andrew S. Tanenbaum, Maarten Van Steen, 2007, ISBN-13: 978-8120334984.
- Parallel Programming for Modern High Performance Computing Systems, 1st Edition, Pawel Czarnul, 2018, ISBN-13: 978-1138305953.
- An introduction to parallel programming, 1st Edition, Peter Pacheco, 2011, ISBN-13: 978-0123742605.

5. Specific course information

a. Catalog Description

This course provides students with required skills to achieve high computing performance using different pervasive technologies such as parallel computing. Parallelisation is mainly based on distributing data and processing among a set of distributed/parallel machines. The students will learn several topics such as parallel architectures, parallel programming methods and techniques, parallel algorithm design, and parallel performance analysis.

b. Prerequisites

60012203-4 Operating Systems
60012401-3 Data Structures

c. Course Type

Required

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Explain large-scale distributed/parallel systems designed architectures and how massive parallelism are implemented.

2. Design and write parallel algorithms and applications using different technologies in heterogeneous platforms.
3. Acquire performance analysis skills and methodology for parallel programs.

7. Brief list of topics to be covered

Introduction + Multi-core and its implications on software engineers
Parallel architectures
Designing parallel program covering: Patterns, dependences, granularity, Data locality, Load balancing, Communication, Synchronization
Parallel Performance (Speedup, Scalability, Amdahl's Law)
Accessing shared data safely
Shared memory programming openMP
distributed memory programming openMP

Computer Security Syllabus

1. Course number and name

60013602-3 Computer Security

---

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Hassan Alamri

4. Textbooks/References

- Introduction to Computer Security, 1st Edition, Michael T. Goodrich and Roberto Tamassia, Addison Wesley. 2011, ISBN-13: 978-0321512949
- Security in Computing, 3rd Edition, Charles P. Pfleeger and Shari L. Pfleeger. PrenticeHall. 2003. ISBN: 0-13-035548-8.

5. Specific course information

a. Catalog Description

This course is the first level of computer and network security. The course will cover various topics related to computer security, data privacy, network protection against various attacks. The course gives students enough knowledge and a reasonable background to understand network security, active and passive attacks, Internet privacy, secure communications. Students are expected to practice biweekly homework, develop critical thinking about computer and network security, and apply learned materials in different contexts of various attacks, wireless and Internet security.

b. Prerequisites

60033103-4 Computer Networks

c. Course Type

Required

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Define information security and outline its major components.

2. Identify and demonstrate the major types of threats to information security and the associated attacks.
3. Develop strategies to protect organization information assets from common attacks based on legal standards.
4. Identify the major techniques, approaches and tools used to discover network and system vulnerabilities.

b. ABET-Enabled Student Outcomes:

- SO3: Communicate effectively in a variety of professional contexts.
- SO4: Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.

- SO3 will be evaluated through CLO2.
- SO4 will be evaluated through CLO3.

7. Brief list of topics to be covered

Introduction: Basic Concepts: Threats, Vulnerabilities, Controls; risk; confidentiality, integrity, availability, security policies, security mechanisms, assurance; prevention, detection, deterrence
Basic cryptography: Basic cryptographic terms, Historical background, Symmetric crypto primitives, Modes of operation, Cryptographic hash functions, Asymmetric crypto primitives
Program security: Flaws: Malicious code: viruses, Trojan horses, worms; Program flaws: buffer overflows, time-of-check to time-of-use flaws, incomplete mediation, Defenses: Software development controls, Testing techniques
Operating Systems Security: Operating Systems Concepts, Operating Systems Security, File System Access Control, Buffer Overflow Attacks
Database management systems security: Database integrity, Database secrecy, Inference control, Multilevel databases
Network security: Network threats: eavesdropping, spoofing, modification, denial of service attacks, Introduction to network security techniques: firewalls, virtual private networks, intrusion detection
Web security: Basic web security model, Web application security, Session management and user authentication, HTTPS: goals and pitfalls
Management of security: Security policies, Risk analysis, Physical threats and controls
Miscellaneous: Legal aspects of security, Privacy and ethics

User Interface Design Syllabus

1. Course number and name

60014502-3 User Interface Design

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(3)

3. Course coordinator

Obiad Alhedriati

4. Textbooks/References

- Designing the User Interface: Strategies for Effective Human-Computer Interaction, Global Edition, Catherine Plaisant, et al., 2017, ISBN-13: 978-1292153919.
- Interaction Design: Beyond Human-Computer Interaction, 5th Edition, Helen Sharp, Jenny Preece, Yvonne Rogers, 2019, ISBN-13: 978-1119547259.
- The Design of Everyday Things: Revised and Expanded Edition, Don Norman, 2013, ISBN-13: 978-0465050659.

5. Specific course information

a. Catalog Description

This course introduces students to the field of User Interface Design (UID) and Human Computer Interaction (HCI). It will provide students with a foundational understanding and practical experience on designing and evaluating usable interactive user interfaces. Topics covered include usability principles, design guidelines and heuristics, usercentered design and evaluation techniques, and the key concepts and theories in UID and HCI. Students will be capable of assessing and arguing the usability of a system front-end such as a mobile application or website and designing more usable and intuitive interfaces.

b. Prerequisites

60013103-4 Advanced Programming

c. Course Type

Required

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Design and implement a prototype and evaluate it.

2. Use the state of art user-centered design principles, techniques and tools to design usable and useful interfaces.
3. Identify users' needs, their design implications and impact on user experience.
4. Understand the designer ethical responsibility when interacting with users.

7. Brief list of topics to be covered

Introduction to Human Computer Interaction, Interaction Design and Usability
User Requirements Analysis: User-centered design, Understanding users' needs, context and design problem, Persona, Scenario, Task analysis
Design: Conceptual and Physical design, Design principles, Interaction Styles, Design patterns
Prototyping: Low-fidelity, high fidelity and The Wizards of Oz prototyping
Evaluation Techniques: Cognitive Walkthrough, Usability Heuristics, Usability Testing, HCI Models and Theories: GOMS, MHP, Fitts' Law
Human Reliability and Error
Universal and Inclusive Design: Accessibility, Multimodal interaction

Summer Training Syllabus

1. Course number and name

60013888-2 Summer Training

2. Credits and contact hours

Credits Hours	2
Contact Hours	-

3. Specific Course Information

a. Catalog Description

The training experience provides students with hands-on work experience in various sectors (industrial, governmental, academic, etc.).

a. Help students to adapt to the work environment.
b. Help students to learn how to make decisions and to take responsibility.
c. Help students to apply theoretical concepts into concrete practical realities.
d. Help students to improve their communication skills.
e. Help students to enhance their skills in writing technical reports.
f. Apply ethical principles and commit to professional ethics, responsibilities and norms of computer science practice.
g. Give an opportunity to companies/organizations to identify distinctive cadres in order to recruit them after graduation.

The duration of training is 240 hours during the EIGHT weeks of Summer term. The students should be able to register for Summer training when they are in Level 8. The department (Summer training committee) assigns a faculty member as a supervisor that should follow, advice, and evaluate the students' work.

At the end of Summer training, the students are required to submit a detailed report that shows their training experience and the gained knowledge. They are also required to give a presentation that presents their training experience

The Summer training committee carries out a rubric assessment based on the submitted report, presentation, employer evaluation letter, and supervisor evaluation.

b. Prerequisites

-

c. Course Type

Required

Computers and Society Syllabus

1. Course number and name

60014305-2 Computers and Society

2. Credits and contact hours

Credits Hours	2
Contact Hours	Theoretical contact hours (2)

3. Course coordinator

Alwi bamahdi

4. Textbooks/References

- A Gift of Fire: Social, Legal, and Ethical Issues for Computing and the Internet, 4th Edition, Sara Baase, 2012, ISBN-13: 978-0132492676.
- Ethics for the Information Age, 7th Edition, Micheal.J. Quinn, 2016, ISBN-13: 9780134296548.
- ACM/IEEE CODE of Ethics.

5. Specific course information

a. Catalog Description

This course explores basic cultural, social, legal, and ethical issues inherent in the discipline of information technology. Students will investigate important non-technical aspects of their role as a computing expert such as personal responsibility in ensuring faulty products are not released to market. Throughout the course, students will analyze different ethical, social, and professional issues related to information technology from the perspective of varying ethical theories and historical, political, or economic disciplines. Students will gain tools to help them analyze situations and make decisions whose implications will affect other people and ultimately society as whole.

b. Prerequisites

60013303-3 Software Engineering I

c. Course Type

Required

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Evaluate and infer benefits and risks of current information technologies.

2. Identify and understand the social, ethical, legal, professional, and privacy issues related to information technology.
3. Analyze ethical, social, and professional issues related to information technology from the perspective of varying ethical theories.
4. Explain the roles and responsibilities of a computer professional.
5. Analyze relationships between information technology and society using critical perspectives or examples from historical, political, or economic disciplines.

b. ABET-Enabled Student Outcomes:

- SO4: Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.

- SO4 will be evaluated through CLO2 & CLO4.

7. Brief list of topics to be covered

Social (cultural, international, govt) implications of computing, and internet
Identifying and evaluating ethical choices in software design
Professionalism (care, attention, responsibility). Importance of keeping up to date
Codes of ethics, maintaining awareness of ethical consequences, ethical dissent
Historical examples of software risks (such as the Therac-25 case)
Computing in the workplace issues
Implications of software complexity
Risk assessment and risk management; risk removal, risk reduction and risk control
Foundations of intellectual property (copyrights, patents)
Software piracy
Ethical and legal basis for privacy protection
Privacy implications of database systems (e.g., data gathering, storage, and sharing)
Technological strategies for privacy protection

Graduation Project I Syllabus

1. Course number and name

60014902-4 Graduation Project I

2. Credits and contact hours

Credits Hours	4
Contact Hours	Theoretical contact hours (4)

3. Course coordinator

GP coordinator

4. Textbooks/References

- UQU Undergraduate Final Year Project Handbook.
- Christian W. Dawson, "Projects in Computing and Information Systems", AddisonWesley Professional; 3rd Edition (2015), ISBN-13: 978-1292073460.

5. Specific course information

a. Catalog Description

This course is the first of a two-course sequence devoted to a team project that constitutes the BSc graduation capstone project. Students are engaged in softwarebased or research-based projects, under the supervision of faculty members, to demonstrate their intellectual, technical and creative abilities acquired from previous courses. In this course, student teams will investigate a particular problem, explore appropriate solutions to the problem to meet the project's requirements, provide an analysis study, and build a prototype design. At the end of the semester, student teams will describe the key ideas of their independent project via an oral presentation and a written report.

b. Prerequisites

60013303-3 Software Engineering
60013103-4 Advanced Programming

c. Course Type

Required

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Identify and analyze a CS-related problem, and define the computing requirements appropriate to its solution.
2. Use the current techniques, skills, and tools necessary for computing practice.
3. Plan effectively for the various project lifecycle activities.
4. Demonstrate the ability to work independently and as part of a team with colleagues and advisors utilizing good work dynamics.

5. Conduct an effective literature survey and be able to contrast and critique related work, as well as produce an academic proposal.

7. Brief list of topics to be covered

Establish a project team and select the project topic to be explored
Project proposal: describe problem definition, project objectives, project scope, and expected project benefits
Literature review and feasibility study
Project management plan: describe the project development approach, associated milestones, agreed deliverables and timeline (Gantt Charts)
System requirements specification (SRS): describe the functional requirements, non-functional requirements, system requirements, expected features, constraints, and assumptions
System design: describe system architecture design, design constraints, roles and responsibilities, database design, and user interface design
Final report: bind all project deliverables in a single report
Final presentation: describe the key ideas of the project via an oral presentation

Graduation Project II Syllabus

1. Course number and name

60014903-4 Graduation Project II

2. Credits and contact hours

Credits Hours	4
Contact Hours	Theoretical contact hours (4)

3. Course coordinator

GP coordinator

4. Textbooks/References

- UQU Undergraduate Final Year Project Handbook.
- Christian W. Dawson, "Projects in Computing and Information Systems", AddisonWesley Professional; 3rd Edition (2015), ISBN-13: 978-1292073460.

5. Specific course information

a. Catalog Description

This course is a continuation of the graduation project started in 14014902-4. The focus in this course will be on low-level design, implementation, testing and quality assurance as well as management of the project. The outcome of this course must be a significant software or/and hardware system, utilizing knowledge acquired in preceding courses. At the end of semester, students must deliver a complete source code, a final report, an oral presentation, a demo, and a poster.

b. Prerequisites

60014902-4 Graduation Project I

c. Course Type

Required

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Commit to the plan and monitor the decisions made in the previous course in developing the software system.
2. Ability to design and implement a computer-based system, process, component or program to meet desired goal.
3. Apply testing and validation concepts and techniques to the system.
4. Apply core computing knowledge such as programming, database, algorithm analysis, modelling and design, demonstrating the comprehension of trade-offs.

5. Demonstrate the ability to work independently and as part of a team with colleagues and advisors utilizing good work dynamics.

b. ABET-Enabled Student Outcomes:

SO2: Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.
SO2 will be evaluated through CLO2.

SO3: Communicate effectively in a variety of professional contexts.
SO3 will be evaluated through CLO5.

SO5: Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.
SO5 will be evaluated through CLO5.

SO6: Apply computer science theory and software development fundamentals to produce computing-based solutions.
SO6 will be evaluated through CLO4.

7. Brief list of topics to be covered

Implementation: describe the used tools and techniques to implement the system, the implementation issues, user interfaces, and how the system operates
Testing and validation: describe the strategy used to test the implemented system and how it is validated against the problem requirements
Final project report and poster
Final presentation and demo

Mobile Applications Syllabus

1. Course number and name

60014105-3 Mobile Applications

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Nura Albehiari

4. Textbooks/References

- Android Programming: The Big Nerd Ranch Guide (Big Nerd Ranch Guides) 4th Edition, Bill Phillips, Chris Stewart, and Kristin Marsicano, 2019, ISBN-13: 9780135245125.
- Professional Android Application Development, 4th Edition, Reto Meier, 2018, ISBN13: 978-1118949528.
- Android Programming for Beginners, 2nd Edition, John Horton, 2018, ISBN-13: 9781789538502.
- Professional Mobile Application Development, 1st Edition, Jeff McWherter and Scott Gowell, 2012, ISBN-13: 978-1118203903.

5. Specific course information

a. Catalog Description

This course is a continuation of the graduation project started in 14014902-4. The focus in this course will be on low-level design, implementation, testing and quality assurance as well as management of the project. The outcome of this course must be a significant software or/and hardware system, utilizing knowledge acquired in preceding courses. At the end of semester, students must deliver a complete source code, a final report, an oral presentation, a demo, and a poster.

b. Prerequisites

60014302-3 Database II
60013104-3 Internet Applications

c. Course Type

Elective

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Apply the Java programming concepts to design and develop Android applications.
2. Design and develop user Interfaces for the Android platform.

3. Use messaging, local database access, and networking features to handle advanced mobile applications.
4. Publish Android applications to the cloud.

7. Brief list of topics to be covered

Introduction to Mobile Applications Programming
Basics of Android Framework
Building First Android Application using Commercial IDE
Developing Android Services and Testing through Android Emulator
Activities and Intents
Getting to know the Android User Interface
Designing User Interface Using Views
Displaying Pictures and Menus with Views
Data Persistence and SQLite Database Programming
Creating Simple Forms for Android
Messaging and Networking
Publishing Android Applications

Cloud Computing Syllabus

1. Course number and name

60014205-3 Cloud Computing

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Aziz Alshehri

4. Textbooks/References

- Cloud Computing: Concepts, Technology and Architecture, 1st Edition, Thomas Erl, Ricardo Puttini, Zaigham Mahmood, 2013, ISBN-13: 978-0133387520.
- Cloud Computing: Theory and Practice, 2nd Edition, Dan C. Marinescu, 2017, ISBN13: 978-0128128107.

5. Specific course information

a. Catalog Description

This course aims to introduce the fundamental techniques, algorithms, and designs of cloud systems. It covers traditional distributed system algorithms that form the basis of modern cloud computing, as well as touching on the architecture of big data platforms such as Hadoop.

b. Prerequisites

60033103-4 Computer Networks

c. Course Type

Elective

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Understand different protocols, P2P systems, and the supporting theoretical foundations.
2. Understand concurrency and replication control.
3. Understand a variety of techniques distributed file systems.

7. Brief list of topics to be covered

Introduction
Gossip and Membership Protocols
P2P Systems
Distributed key-value stores
Time and Ordering
Snapshots, Multicast, Paxos
Leader Election
Concurrency and Replication Control
Stream and Graph Processing
Distributed File Systems

Big Data Analytics Syllabus

1. Course number and name

60014305-3 Big Data Analytics

---

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Khalil alsolbi

4. Textbooks/References

- Big Data, Data Mining, and Machine Learning: Value Creation for Business Leaders and Practitioners, 1st Edition, Jared Dean, 2014, ISBN-13: 978-1118618042.
- Big Data, Mining, and Analytics: Components of Strategic Decision Making, 1st Edition, Stephan Kudyba, 2014, ISBN-13: 978-1466568709.

5. Specific course information

a. Catalog Description

Storage, retrieval, analysis, and knowledge discovery using Big Data has made significant inroads in several domains in industry, research, and academia. In this course, we will look at the dominant software systems and algorithms for coping with Big Data. Topics covered include scalable computing models, large-scale nontraditional data storage frameworks including graph, key-value, and column-family storage systems; data stream analysis; scalable prediction models and in-memory storage systems.

b. Prerequisites

60014302-3 Database II

c. Course Type

Elective

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Understand various types of models like computing, data and prediction.
2. Understand different distributed systems.
3. Analysis real-time data stream.
4. Understand the frameworks for the graph data analytics.

7. Brief list of topics to be covered

Batch computing models for Big Data computing
Key-value storage systems
Scalable prediction models
Distributed file systems
Scalable data analytics
Data models
Real-time data stream analytics
Frameworks for the graph data analytics In-memory distributed data storage systems

Introduction to Cryptography Syllabus

1. Course number and name

60014604-3 Introduction to Cryptography

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Khalid almurahbi

4. Textbooks/References

- Cryptography and Network Security: Principles and Practice, 7th Edition, by William Stallings, PEARSON, ISBN-13: 978-0134444284.

5. Specific course information

a. Catalog Description

This course introduces cryptography from a theoretical and practical perspective. It is an introductory course on methods, algorithms, techniques, and tools of cryptography. It will cover how cryptography works, how security is analyzed theoretically, and how exploits work in practice. To achieve these goals, students will learn symmetric-key encryption, stream ciphers, block ciphers, message authentication codes, as well as asymmetric encryption (RSA- and discrete-log-based), and digital signatures.

b. Prerequisites

60011802-3 Discrete Structures II

c. Course Type

Elective

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Understand basic principles of cryptography and general cryptanalysis.
2. Be acquainted with the concepts of symmetric encryption and authentication as well as public key encryption, digital signatures, and key establishment.
3. Know and understand common examples and uses of cryptographic schemes, algorithms, as well as protocols, and know how and when to apply them.
4. Be able to compose, build and analyze simple cryptographic solutions.

7. Brief list of topics to be covered

Introduction and Overview Classical Cryptography
Mathematics of Cryptography
More Classical ciphers :
Stream cipher
Block cipher
Public-key Cryptography
Hash Functions
Secret Sharing

Bioinformatics Syllabus

1. Course number and name

60014404-3 Bioinformatics

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Hanan Alghamdi

4. Textbooks/References

- Understanding Bioinformatics, 1st Edition, Marketa Zvelebil, Jeremy Baum, 2007, ISBN-13: 978-0815340249.
- Introduction to Bioinformatics, 4th Edition, Arthur Lesk, 2014, Oxford University Press, ISBN-13: 978-0199651566.

5. Specific course information

a. Catalog Description

This course introduces students to the emerging field of bioinformatics and how computational techniques can be employed in this area. The course is aimed at computer science students to give them knowledge of

a. basic introduction to bioinformatics
b. biological databases
c. genes, genomes, mapping and DNA sequencing algorithms
d. visualization/clustering gene pattern
e. linking genes and disease

b. Prerequisites

60012402-4 Algorithms

c. Course Type

Elective

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Learn the basics of bioinformatics, Gene, Genome and biological databases.
2. Poses good understanding of sequence alignment algorithms including patterns, profiles and multiple alignments.
3. Develop prediction algorithms for protein structures and microarray data analysis.

7. Brief list of topics to be covered

Introduction to Bioinformatics, Gene and Genome
Biological Databases
Sequence Alignment algorithms and dynamic programming
Patterns, Profiles, and Multiple Alignments
Prediction Algorithms for Protein Structures
Microarray data analysis
Visualization/Clustering of Gene Patterns

Information Retrieval Systems Syllabus

1. Course number and name

60014308-3 Information Retrieval Systems

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (3)

3. Course coordinator

Musab Alzaghol

4. Textbooks/References

- Information retrieval: Implementing and evaluating search engines, Stefan Büttcher, Charles L. A. Clarke, Gordon V. Cormack, 2016, ISBN-13: 978-0262528870.
- Introduction to Information Retrieval, 1st Edition, Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze, 2008, ISBN-13: 978-0521865715.

5. Specific course information
   a. Catalog Description

This course is aimed at finding the relevant information and subsequently extracting meaningful patterns out of it. While the basic theories and mathematical models of information retrieval and data mining are covered, the course is primarily focused on practical algorithms of textual document indexing, relevance ranking, web usage mining, text analytics, as well as their performance evaluations. Practical retrieval and data mining applications such as web search engines, personalization and recommender systems, business intelligence, and fraud detection will also be covered.

   b. Prerequisites

60013103-4 Advanced Programming

   c. Course Type

Elective

6. Specific goals for the course
   a. Course Learning Outcomes (CLOs)

1. Design and implement data retrieval solutions.
2. Understand indexing and ranking techniques.
3. Apply machine learning techniques in information retrieval.

7. Brief list of topics to be covered

Boolean retrieval
Term vocabulary and posting lists
Dictionaries and tolerant retrieval
Index construction and compression
Term scoring, weighting, and vector space model
Computing scores in a complete search systems
Information retrieval evaluation
XML Retrieval
Web search basics

Artificial Neural Networks Syllabus

1. Course number and name

60014702-3 Artificial Neural Networks

---

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Amani Alluhiabil

4. Textbooks/References

- Fundamentals of Neural Networks: Architectures, Algorithms And Applications, 1st Edition, Laurene V. Fausett, 1993, ISBN-13: 978-0133341867.
- Neural Network Design, 2nd Edition, Martin T Hagan, et al., 2014, ISBN-13: 9780971732117.

5. Specific course information

a. Catalog Description

By the end of this course, student should be capable of understanding basic neural network architectures, their applications, understanding various learning algorithm related to neural networks and can apply the Neural Networks to solve practical problems.

b. Prerequisites

60013701-4 Artificial Intelligence

c. Course Type

Elective

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Learn the basics of Neural Networks and related mathematical concepts.
2. Learn in detail different single layer and multilayer neural network architectures and their learning mechanisms for supervised learning.
3. Learn different neural network architectures for unsupervised learning.

7. Brief list of topics to be covered

Introduction to Neural Networks
Revision on Math Concepts needed for Neural Networks
Perceptron, linear systems and their limitations
Multi-layer networks and back propagation
Supervising learning, optimization and over-fitting
Unsupervised learning
Reinforcement learning

Database II Syllabus

1. Course number and name

60014302-3 Database II

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (3)

3. Course coordinator

Fatemah Albargi

4. Textbooks/References

- Fundamentals of Database Systems, 7th Edition, Ramez Elmasri, Shamkant B. Navathe, 2015, ISBN-13: 978-0133970777.
- Modern Database Management, 12th Edition, Jeffrey A. Hoffer, Ramesh Venkataraman, Heikki Topi, 2015, ISBN-13: 978-0133544619.
- Database Systems: Models, Languages, Design and Application Programming, 6th Edition, Ramen Elmasri, Shamkant Navathe, 2011, ISBN-13: 978-0132144988.

5. Specific course information

a. Catalog Description

This course provides advanced database knowledge. It presents the basics of transactions, data mining and warehousing, query processing and optimization, database tuning, distributed and NoSQL databases.

b. Prerequisites

60012301-3 Database I

c. Course Type

Elective

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Understand different types of database systems such as semi structured databases and NoSQL.
2. Apply advanced database techniques such as indexing, query processing, local and distributed transaction processing, backup and recovery and security to manage and optimize a database performance.
3. Use techniques of advanced schema mapping i.e. from enhanced entity relation to relational, object to relational, object-relational to relational, and xml to relational.

7. Brief list of topics to be covered

The Enhanced Entity-Relationship (EER) model
Transactions: failures, atomicity, consistency, isolation, durability
Query Processing and Query Optimization Techniques
Database Backup and Recovery
Object and Object-Relational Databases
XML for Semi-Structured Data
Distributed Databases (DDB)
Database Security
Distributed and NoSQL Databases

Theory of Computing Syllabus

1. Course number and name

60014803-3 Theory of Computing

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (3)

3. Course coordinator

Awadh Ramadan

4. Textbooks/References

- An Introduction to Formal Languages and Automata, 6th Edition, Peter Linz, 2016, ISBN: 9781284077254.
- Introduction to the Theory of Computation, 3rd Edition, Michael Sipser, 2012, ISBN13: 978-1133187790.
- Introduction to Automata Theory, Languages, and Computation, 3rd Edition, Rajeev Motwani, Jeffrey D. Ullman, John E. Hopcroft, 2013, ISBN-13: 978-1292039053.

5. Specific course information
   a. Catalog Description

This course introduces the key concepts of theory of computation. It discusses automata and their relationship to regular, context-free and phrase-structure languages. The computability theory is also introduced, including Turing machines, and decidability.

   b. Prerequisites

60011802-3 Discrete Structures II

   c. Course Type

Elective

6. Specific goals for the course
   a. Course Learning Outcomes (CLOs)

1. Learn basics of finite automata as a modeling tool of computational problems.
2. Learn context-free languages and their limitations.
3. Understand in defining classes of equivalent problems from a computational perspective.

7. Brief list of topics to be covered

Finite Automata (DFA)
Non deterministic automata (NFA), Equivalence with DFA
Regular expression, Regular languages, Equivalence with DFA
Nonregular languages, Pumping Lemma
Context-free Grammar and Languages
Pushdown Automata (PDA)
Equivalence of PDA and CFG
Turing machines
Decidability and complexity

Image Processing Syllabus

1. Course number and name
   60014503-3 Image Processing

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Ebulkareem abdulkader

4. Textbooks/References

- Digital Image Processing, 4th Edition, Rafael C. Gonzalez, Richard E. Woods, 2017, ISBN-13: 978-0133356724.
- Digital Image Processing Using MATLAB, 2nd Edition, Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, 2009, ISBN-13: 978-0982085400.

5. Specific course information
   a. Catalog Description

This course introduces students to the fundamental concepts, principles, and tools of digital image processing. The course includes topics: image sampling and quantization, color, point operations, segmentation, morphological image processing, linear image filtering, image transforms, noise reduction and restoration and compression.

   b. Prerequisites

60012402-4 Algorithms

   c. Course Type

Elective

6. Specific goals for the course
   a. Course Learning Outcomes (CLOs)

1. Learn image processing fundamentals and enhancements.
2. Learn some advanced image processing concepts of image transform, recovery and segmentations architectures and their learning mechanisms for supervised learning.
3. Learn the concept of image compression, morphological and color image processing.

7. Brief list of topics to be covered

Introduction to Image Processing, Digital Image Fundamentals (Sampling, Quantization) Point Operations, Histograms
Image Enhancement, Linear Image Processing and Filtering
Image transforms, noise reduction and restoration
Image Segmentation and Edge Detection
Image Compression
Morphological and Color Image Processing

Software Testing Syllabus

1. Course number and name

60014306-3 Software Testing

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Edrees Alkinani

4. Textbooks/References

- The Art of Software Testing, 3rd Edition, Glenford Myers, Corey Sandler, Tom Badgett, 2011, ISBN-13: 978-1118031964.
- Advanced Software Testing - Vol. 1, 2nd Edition, Rex Black, 2015, ISBN-13: 9781937538682.
- Software Testing - An ISTQB-BCS Certified Tester Foundation Guide, 3rd Edition, Brian, Peter, Angelina, Geoff, Peter, 2015, ISBN-13: 978-1780172996.
- Quality Code: Software Testing Principles, Practices, and Patterns, 1st Edition, Stephen Vance, 2013, ISBN-13: 978-0321832986.

5. Specific course information

a. Catalog Description

This course introduces software testing and quality assurance. The relationship of software testing to quality is examined with an emphasis on testing techniques. Students will learn quality and testing concepts. They will also acquire the ability to write test plans, test design specifications, test cases, and use test metrics to manage the test process.

b. Prerequisites

60013304-3 Software Engineering II

c. Course Type

Elective

6. Specific goals for the course
   a. Course Learning Outcomes (CLOs)

1. Identify typical objectives of testing and differentiate between testing and debugging. 2. Design test plan and test cases and asses the software product correctly. 3. Manage and perform the software testing and quality assurance activities.

7. Brief list of topics to be covered

SQA Concepts Basic notions: Quality Assurance, Detection vs. Prevention, Verification & Validation, testing
Testing Concepts :Definition, Types and Levels of testing, Black vs. White Box testing
Static Analysis Techniques
Specification-based or Black-box Techniques
Structure-based or White-box Techniques
Test Tools and Automation
Test Metrics : Pre-process metrics: Estimation In-process metrics: Process Management, End-process metrics: Process Improvement
Test Management : Test planning, resource management, test reporting, tools

Pattern Recognition Syllabus

1. Course number and name

60014703-3 Pattern Recognition

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Muna Alawfi

4. Textbooks/References

- Pattern Recognition, 4th Edition, Sergios Theodoridis, Konstantinos Koutroumbas, 2008, ISBN-13: 978-1597492720.
- Pattern Recognition and Machine Learning, Christopher Bishop, 2011, ISBN-13: 9780387310732.

5. Specific course information

a. Catalog Description

This course introduces pattern recognition and machine learning techniques. Topics include modern statistical methodologies (Bayesian Decision Theory, Probability Theory), clustering (K-Means, Fuzzy Clustering), classifiers (Support Vector Machines, Decision Trees, Nearest Neighbor Classification), and ensemble methods.

b. Prerequisites

60012402-4 Algorithms

c. Course Type

Elective

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Learn mathematical and statistical concepts required for pattern recognition.
2. Learn various classifiers and their applications in real-life applications.
3. Learn various clustering methods.
4. Learn different feature selection methods.

7. Brief list of topics to be covered

Introduction to Pattern Recognition
Statistical Analysis-Bayesian Decision Theory, Probability Theory
Classifiers-Support Vector Machines, Decision Trees, Nearest Neighbor Classification
Clustering-K-Means, Fuzzy Clustering
Graphical Models
Sequential Pattern Recognition

Game Programming Syllabus

1. Course number and name

60014108-3 Game Programming

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Essa Muharish

4. Textbooks/References

- Game Coding Complete, Mike McShaffry, David Graham, 4th Edition, 2012, ISBN-13: 978-1133776574.
- Programming Game AI by Example, 1st Edition, Mat Buckland, 2004, ISBN-13: 9781556220784.
- Mathematics for 3D Game Programming and Computer Graphics, 3rd Edition, Eric Lengyel, 2011, ISBN-13: 978-1435458864.

5. Specific course information

a. Catalog Description

This course provides students with hands-on experience in game programming. Students will learn the rudiments of game design and the common workflow practices within the industry. The course covers the basic techniques, including game design, game engine, game mechanics, game strategies, interface design, event handling, graphics algorithms, and intelligent behaviors. Students will learn the skills of designing and developing interactive and real-time three-dimensional games with multimedia elements such as animation, audio, and video. The course assumes that students have proficient programming skills in computer graphics (preferably in C++) and sufficient knowledge of user interface design.

b. Prerequisites

60012501-3 Computer Graphics

c. Course Type

Elective

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Understand and employ the common approaches to game AI, game physics, game networking, game graphics (2D and 3D), and game engine.
2. Apply effective, industry-standard design, production and testing techniques through all phases of game development.
3. Develop interactive games using a variety of techniques including geometry and artificial intelligence.
4. Build and then integrate 3D game technologies such as multimedia, artificial intelligence, and physics modeling into a cohesive, interactive game application.

7. Brief list of topics to be covered

Introduction to Game Programming and Game Engine
Game Architecture
Game Logic and Mechanics
Game Actors
Memory Management and Coding Optimization
Collision and Simple Physics
UI Design
Event Management
AI for Games
Network Programming for Multiplayer Games

Programming Languages Syllabus

1. Course number and name

60014106-3 Programming Languages

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (3)

3. Course coordinator

Ali Alfaqeeh

4. Textbooks/References

- Concepts of Programming Languages, 11th Edition, Robert W Sebesta, 2016, ISBN-13: 978-0133943023.
- Programming Language Explorations, 1st Edition, R. Toal, R. Rivera, A. Schneider, E. Choe, 2017, ISBN-13: 978-1498738460.

5. Specific course information

a. Catalog Description

This course is an introduction to the theory, design, and implementation of programming languages. Topics covered in this course include control structures, data types and structures, runtime, environments, binding strategies, compilers, and interpreters.

b. Prerequisites

60013103-4 Advanced Programming

c. Course Type

Elective

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Apply concepts from prerequisite courses, especially formal languages and architecture courses, in the context of evaluating the features of programming languages.
2. Explain and evaluate design and implementation features of programming languages.
3. Apply conceptual knowledge of the syntax of languages, as well as the design of language data structures and control statements, to the efficient implementation of a working language.

7. Brief list of topics to be covered

Preliminaries: Reason for studying concepts of programming languages, Programming Domains, Language Evaluation Criteria Influences on Language Design, Language Categories, Language design Trade-Off, Implementation, Introduction of Object Oriented Programming.
Describing Syntax and Semantics: Introduction of Syntax and Semantics, The General Problems of Describing Syntax (Language Recognizers, Language Generator), Formal Methods of Describing Syntax.
Describing Syntax and Semantics: Attribute of Grammars: Static Semantics, Attribute & Grammars Defined, Intrinsic Attributes, Examples of Attribute Grammars, Computing Attribute Values, Evaluation, Describing the Meaning of Programs: Dynamic Semantics.
Data Types: Introduction of Data Types, Primitive Data Types, Character String Types, User-Defined Ordinal Types
Data Types: Types of Arrays, Associative Arrays, Record Types
Data Types: Union Types, Structure Types, Set Types, Pointer Types
Support of Object Oriented Programming Concepts: Design Issues for Object Oriented Languages, Support for Object Oriented Programming in C++,
Support of Object Oriented Programming Concepts: Support for Object Oriented Programming in Java, The Object Oriented Model of java Script
Exception Handling: Introduction to Exception Handling, Exception handling in PL/I, Exception Handling in C++, Exception Handling in Java

Advanced Web Programming Syllabus

1. Course number and name

60014110-3 Advanced Web Programming

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Khudran Alzhrani

4. Textbooks/References

- HTML5 and CSS3, 8th Edition, ElizabEth Castro, Bruce hyslop, 2013, ISBN: 9780321928832.
- JavaScript: The Definitive Guide, 6th Edition, David Flanagan, 2011, ISBN: 0596805527.
- Java Servlet Programming, 2nd Edition, William Crawford, Jason Hunter, 2001, ISBN: 0596000405.
- Ajax: The Definitive Guide, 1st Edition, Anthony Holdener III, 2008, ISBN: 0596528388.
- Servlet, JSP and Spring MVC: A Tutorial, Budi Kurniawan, Paul Deck, 2015, ISBN: 9781771970020.

5. Specific course information

a. Catalog Description

This course enables students to develop advanced skills in multi-tier website development and administration. Students will explore front-end and back-end Web technologies such as HTML, CSS and JavaScript, Servlets, JSP and integrations with relational databases. The course will also focus on building complete websites and issues related to session management, authentication, AJAX (Asynchronous JavaScript And Xml), and CRUD (Create, Read, Update and Delete) database operations.

b. Prerequisites

60013104-3 Internet Applications

c. Course Type

Elective

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Build basic Web applications using HTML, CSS and JavaScript for building the application front-end, and JavaServlet and JSP for building the application backend.

2. Use advanced JavaScript and CSS techniques to improve the usability of the Web applications.
3. Use Asynchronous JavaScript and XML (AJAX) to build responsive web applications.
4. Use industrial framework such as Spring.

7. Brief list of topics to be covered

Revision on HTML, CSS and JavaScript
JQuery, Object Oriented JavaScript and Dynamic HTML
Servlet Programming, JSP, Session Management, Database Integration
XML/JSON (Extensible Mark-up Language/JavaScript Object Notation)
AJAX (Asynchronous JavaScript)-Based Web Applications
Using Web Framework (such as Spring)

Software Architecture Syllabus

1. Course number and name

60014307-3 Software Architecture

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Edrees Alkinani

4. Textbooks/References

- Java RMI, 1st Edition, William Grosso, 2001, ISBN-10: 1565924525.
- SOA with Java: Realizing Service-Orientation with Java Technologies, 1st Edition, Thomas Erl, Andre Tost, Satadru Roy, Philip Thomas, 2014, ISBN-10: 0133859037.
- Getting Started with OAuth 2.0, 1st Edition, Ryan Boyd, 2011, ISBN-10: 1449311601.
- Web Security Testing Cookbook: Systematic Techniques to Find Problems Fast, 1st Edition, Paco Hope, Ben Walther, 2008, ISBN-10: 0596514839.

5. Specific course information

a. Catalog Description

The goal of this course is to get the students to become familiar with the different software architectures and to gain the knowhow on using these architectures. Students will learn the following topics: the concepts of virtual software bus, CORBA, RMI, SOA, Web services, OAuth, trust, security, Web bases attacks (SQL injections, session stealing, etc...), and how to store login information on the client machine using temporary tokens.

b. Prerequisites

60014110-3 Advanced Web Programming

c. Course Type

Elective

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Learn the concept of virtual software bus architectures such as CORBA, RMI, SOA, and how to use them.
2. Learn the concepts of OAuth, trust, security, and how to implement OAuth applications.

3. Learn Web bases attacks (SQL injections, session stealing, etc.) and how to protect your applications.

7. Brief list of topics to be covered

Virtual bus concept
CORBA concepts and implementation
RMI/Java
SOA and web services
OAuth concepts and implementation
Multi-tier web applications and web framework (such as Spring)
Web bases attacks (SQL injections, session stealing, etc...) and how to protect your application

Forensics Computing Syllabus

1. Course number and name

60014605-3 Forensics Computing

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Khalid Almarhabi

4. Textbooks/References

- Guide to Computer Forensics and Investigations, 5th Edition, by Bill Nelson, Amelia Phillips, Christopher Steuart, 2015, ISBN-13: 978-1285060033.
- Digital Forensics with Open Source Tools, 1st Edition, by C. Altheide & H. Carvey Syngress, 2011. ISBN: 9781597495868.

5. Specific course information

a. Catalog Description

This course presents an overview of the principles and practices of digital investigation. The objective of this class is to emphasize the fundamentals and importance of digital forensics. Students will learn different techniques and procedures that enable them to perform a digital investigation. This course will focus on the analysis of physical storage media and volume analysis. It covers the major phases of digital investigation such as preservation, analysis and acquisition of artifacts that reside in hard disks and random-access memory.

b. Prerequisites

60013602-3 Computer Security

c. Course Type

Elective

6. Specific goals for the course

a. Course Learning Outcomes (CLOs)

1. Correctly define and cite appropriate instances for the application of computer forensics.
2. Correctly collect and analyze computer forensic evidence.
3. Identify the essential and up-to-date concepts, algorithms, protocols, tools, and methodology of computer forensics.

7. Brief list of topics to be covered

Intro to Forensics
Forensics Technologies
Data Recovery
Disk and File Systems
Evidence Collection
Preservation, Verification, and Authentication
Discovery and Identification
Windows Artifacts
Linux Artifacts
Mac OS Artifacts
File Analysis
Net Analysis

Natural Language Processing Syllabus

1. Course number and name

60014704-3 Natural Language Processing

2. Credits and contact hours

Credits Hours	3
Contact Hours	Theoretical contact hours (2) - Practical contact hours(2)

3. Course coordinator

Hanan Alghamdi

4. Textbooks/References

- Speech and Language Processing, 2nd Edition, Daniel Jurafsky, James H. Martin, 2008, ISBN-13: 978-0131873216.
- Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit, 1st Edition, Steven Bird, Ewan Klein, Edward Loper, 2009, ISBN-13: 9780596516499.
- Modern Information Retrieval: The Concepts and Technology behind Search, 2nd Edition, Ricardo Baeza-Yates, Berthier Ribeiro-Neto, 2011, ISBN-13: 9780321416919.

5. Specific course information

a. Catalog Description

This course introduces the key concepts and ideas in natural language processing. It covers both the algorithms available for the processing of linguistic information and the fundamental computational properties of languages.

b. Prerequisites

60013701-4 Artificial Intelligence

c. Course Type

Elective

6. Specific goals for the course

Course Learning Outcomes (CLOs)

1. Learn basic understanding of natural language processing.
2. Learn semantics and pragmatics of English language for processing.
3. Learn POS tagging and context free grammar for English language.
4. Learn information retrieval techniques.

7. Brief list of topics to be covered

Introduction to Natural Language Processing
Linguistic Essentials
Regular Expressions
Finite State Automata, Finite State Transducers
Basic Notions of Probability Theory, Statistical NLP (n-gram, smoothing)
Part-of-Speech Tagging
Grammars
Complexity, Semantics
Information Retrieval, Information Extraction, Question-Answering