

Curriculum / Scheme of Studies
of
Bachelor of Science in Computer Science (Post
ADP)
BS Computer Science (Post ADP)
(2023)



University of Education, Lahore

Table of Contents

| | | |
|------|--|----|
| 1. | Introduction to BS Computer Science Program (Post ADP) | 3 |
| 2. | Program Vision Statement..... | 3 |
| 3. | Program Mission Statement | 4 |
| 4. | Program Objectives | 4 |
| 5. | Curricula Consideration | 4 |
| 5.1. | Association of Computing Machinery (ACM) - Guidelines..... | 5 |
| 5.2. | Knowledge Areas in ACM CS 2013 Curriculum..... | 6 |
| 6. | Outcome Based Education (OBE) System and Seoul Accord: | 7 |
| 7. | Curriculum Model for Bachelor of Science in Computer Science..... | 9 |
| 8. | Semester/Study Plan for BSCS (Post ADP)..... | 11 |
| 9. | Eligibility Criteria, Duration of the Program and Award of Degree: | 13 |
| 10. | Course Outlines | 15 |
| 11. | FYP Deliverable 1 and Deliverable 2 Guidelines | 48 |

1. Introduction to BS Computer Science Program (Post ADP)

Computer science is the study of the theory, experimentation, and engineering that form the basis for the design and use of computers. It is the scientific and practical approach to computation and its applications and the systematic study of the feasibility, structure, expression, and mechanization of the methodical procedures (or algorithms) that underlie the acquisition, representation, processing, storage, communication of, and access to information [ref WordNet Princeton definition].

Computer Science is the application of a systematic, disciplined and quantifiable approach to the design, development, operation, and maintenance of software systems. It is in fact the practice of designing and implementing large, reliable, efficient and economical software by applying the principles and practices of engineering. The program aims to train students in all aspects of software life cycle from specification through analysis and design to testing, maintenance and evaluation of software product.

Computer Science spans a wide range, from its theoretical and algorithmic foundations to cutting-edge developments in robotics, computer vision, intelligent systems, bioinformatics, and other exciting areas. The overall scope of Computer Science may be viewed into the following three categories:

- To develop effective ways to solve computing problems. For example, Computer Science develops the best possible ways to store information in databases, send data over networks, and display complex images. The theoretical background offered by Computer Science allows determining the best performance possible, and their study of algorithms. It enables to develop new problem-solving approaches that provide better performance.
- It devises new ways to use computers intelligently and effectively. Progress in the areas of networking, database, and human-computer-interface came together as a result of the world-wide-web, which changed the entire world. Now, researchers are working to make robots that are practical aides and demonstrate intelligence, databases that create new knowledge and, in general, use computers to do new things.
- It deals with the design and implementation of software systems. Computer Science provides training and skills for the successful implementation of software systems that solve challenging

programming jobs. Computer Science spans the range from theory to models, design and programming. Computer Science offers a comprehensive foundation that permits graduates to adapt to new technologies and new ideas.

2. Program Vision Statement

The BSCS (Post ADP) program aspires to innovate, broaden, publish and impart the advanced knowledge of computer science enabling students to participate and contribute in their field locally, nationally and globally through academia, research and applications.

3. Program Mission Statement

The mission of the program is to impart modern, quality, comprehensive and effective theoretical as well as applied education in various domains of Computer Sciences. Also, to instill high degree professionalism in student by developing their communication, problem solving and technical skills to meet modern and future challenges.

The program is designed to equip students with diverse professional knowledge and skills through a comprehensive degree program that enables them to face the future challenges of evolution in the field of computer sciences and contribute in the industry as software developers, technology professionals and computer science academics.

4. Program Objectives

The objective of the program is providing a learning experience that produces high quality professionals equipped with technological, research, communication, problem solving, managerial and leadership skills that contribute to their profession as managers, academics, innovators and planners in the field of computer sciences and its related fields. Provide an environment that encourages innovation and enables imparting of theoretical knowledge, supports its practical applications and empowers acquisition of new competences for the future. Also to train students with the fundamental knowledge that is in line with the state of art researches and technologies to prepare them for global challenges and national needs. Some of the key objectives of the program are listed below:

- To provide a broad and in depth latest knowledge of concepts, theory, techniques through intensive education/training in varying domains within the field of computer science as well as supporting areas of science, mathematics and management.
- To impart problem solving and analytical thinking capabilities to propose and develop solution with practical applications in the industry and research.
- To encourage students to innovate and provide necessary communication skill and environment that enables them to present their proposed solution through well-organized reports, presentations and project utilizing latest tools and technologies.
- To develop awareness regarding state of the art researches and the ever-changing technological environment in the industry.
- To provide formal foundations for higher learning and education while also preparing the students for industry.

5. Curricula Consideration

During the revision of the Computing Curricula two major guidelines have been considered (ACM and Seoul Accord). However, in some cases the main focus of these guidelines is mostly traditional Computer Science program.

5.1. Association of Computing Machinery (ACM) - Guidelines

Association of Computing Machinery (ACM), USA is the largest body in the world for computer scientists. Its membership is spread over the entire globe. It has a pool of highly reputed professionals which meet after a few years to assess the directions being taken by the computing discipline. In view of its assessment, it identifies knowledge areas and also their relative importance in the years to come. Thus, ACM shows the path to follow to the computing academia and professionals all over the world. Computing curricula are designed keeping in view following identified knowledge areas of ACM [ref # ACM 2013 curriculum report]. It has been tried to reasonably cover all knowledge areas without compromising the flexibility needed for a national model curriculum. The mapping of these key knowledge areas with the courses are given in table below.

- AL -Algorithms and Complexity
- AR -Architecture and Organization
- CN -Computational Science
- DS -Discrete Structures
- GV -Graphics and Visual Computing
- HCI -Human-Computer Interaction
- IAS -Information Assurance and Security
- IM -Information Management
- IS -Intelligent Systems
- NC -Networking and Communications
- OS -Operating Systems
- PBD - Platform-based Development
- PD -Parallel and Distributed Computing
- PL -Programming Languages
- SDF -Software Development Fundamentals
- SE -Software Engineering
- SF -Systems Fundamentals
- SP -Social Issues and Professional Issue

The following knowledge areas have been addressed with the major computing courses.

5.2. Knowledge Areas in ACM CS 2013 Curriculum

| # | Knowledge Area | CS 2013 | | ACM 2013 Subjects Taught in Various Universities | NCEAC Revised 2023 Subjects in Core |
|----|--|---------|--------|---|--|
| | | Tier-1 | Tier-2 | | |
| 1 | AL-Algorithms and Complexity | 19 | 9 | Algorithms; Algorithms and Data Structures; Algorithm Design and Analysis | Data structures, Analysis of Algorithms, Theory of Automata |
| 2 | AR-Architecture and Organization | 0 | 16 | Intro to Computer Architecture; DLD; Computer Engineering | DLD, Computer Org & Assembly Language, Computer Architecture |
| 3 | CN-Computational Science | 1 | 0 | eScience; Modeling and Simulation; Computer Graphics | HCI & Computer Graphics; (Elective: Numerical Analysis) |
| 4 | DS-Discrete Structures | 37 | 4 | Discrete Mathematics; Mathematical Foundations of CS; Probability for CS; Discrete Structures 1; Discrete Str 2 | Discrete Structures, Introduction to Statistics |
| 5 | GV-Graphics and Visualization | 2 | 1 | Computer Graphics; Computer Graphics | HCI & Computer Graphics; (Elective: Computer Graphics) |
| 6 | HCI-Human-Computer Interaction | 4 | 4 | Human Computer Interaction | HCI & Computer Graphics |
| 7 | IAS-Information Assurance and Security | 3 | 6 | Computer Systems Security | Information Security; (Elective: Cyber Security) |
| 8 | IM-Information Management | 1 | 9 | Database Systems | Database Systems; Adv Database Management Sys |
| 9 | IS-Intelligent Systems | 0 | 10 | Artificial Intelligence Programming; Artificial Intelligence | Artificial Intelligence |
| 10 | NC-Networking and Communication | 3 | 7 | Introduction to Computer Networking; Computer Networks | Computer Networks |

| | | | | | |
|----|--|------------|------------|--|--|
| 11 | OS-Operating Systems | 4 | 11 | Operating Systems | Operating Systems |
| 12 | PBD-Platform-based Development | 0 | 0 | | (Electives: Web Technology {ASP, Javascript}, Visual Prog {C#}, Mobile App Dev {React/Flutter/Kotlin/Swift}) |
| 13 | PD-Parallel and Distributed Computing | 5 | 10 | Parallel Programming Principle and Practice; | Parallel & Distributed Computing |
| 14 | PL-Programming Languages | 8 | 20 | Introduction to Compilers; Compilers; Introduction to Programming; Programming Languages | Programming Fundamentals, OOP, Compiler Construction |
| 15 | SDF-Software Development Fundamentals | 43 | 0 | Java Programming I; Introduction to Program Design: Introduction to Programming; OOP | Programming Fundamental, Object Oriented Programming, Data Structures |
| 16 | SE-Software Engineering | 6 | 22 | Software Engineering | Software Engineering |
| 17 | SF-Systems Fundamentals | 18 | 9 | Computer Systems and Networks; Great Ideas in Computer Architecture; System Programming | DLD, Computer Networks, Computer Architecture |
| 18 | SP-Social Issues and Professional Practice | 11 | 5 | Ethics in Technology; Technology Consulting in the Community | Professional Practices |
| | Total Core Hours | 165 | 143 | | |
| | | 308 | | | |

6. Outcome Based Education (OBE) System and Seoul Accord:

Keeping in view the latest transformation from knowledge-based education philosophy to Outcome based education (OBE) system, the OBE model based on Seoul Accord has also been considered. Computing programs prepare students to attain educational objectives by ensuring that students demonstrate achievement of the following outcomes (derived from Graduate Attributes define by Seoul Accord www.seoulaccord.org).

| S# | Program Learning Outcomes (PLOs) | Computing Professional Graduate |
|----|--|--|
| 1 | Academic Education | To prepare graduates as computing professionals |
| 2 | Knowledge for Solving Computing Problems | Apply knowledge of computing fundamentals, knowledge of a computing specialization, and mathematics, science, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and Requirements. |
| 3 | Problem Analysis | Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines. |
| 4 | Design/ Development of Solutions | Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations. |
| 5 | Modern Tool Usage | Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations. |
| 6 | Individual and Team Work | Function effectively as an individual and as a member or leader in diverse teams and in multi-disciplinary settings. |
| 7 | Communication | Communicate effectively with the computing community and with society at large about complex computing activities by being able to comprehend and write effective reports, Design documentation, make effective presentations, and give and understand clear instructions. |
| 8 | Computing Professionalism and Society | Understand and assess societal, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice. |
| 9 | Ethics | Understand and commit to professional ethics, responsibilities, and norms of professional computing Practice. |
| 10 | Life-long Learning | Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional. |

7. Curriculum Model for Bachelor of Science in Computer Science

The generic structure for computing degree program given before is mapped with the BSCS (Post ADP) program in the following tables.

Generic Structure for Computing Disciplines:

| Areas | Credit Hours | Courses |
|-----------------|--------------|-----------|
| Computing Core | 12 | 4 |
| Domain Core | 18 | 6 |
| Domain Elective | 21 | 7 |
| Allied Courses | 9 | 3 |
| Internship | 3 | 1 |
| Totals | 63 | 21 |

Mapping of BSCS (Post ADP) Program on the Generic Structure:

| # | Code | Pre-Reqs | Course Title | Dom | Cr Hr |
|-------------------------------|----------|----------------------------|---|-------------|---------|
| Computing Core Courses | | | | | |
| 1 | COMP2117 | | Data Structures (Prerequisite) | Core | 4 (3+1) |
| 2 | COMP3142 | Data Structures | Operating Systems | Core | 3 (2+1) |
| 3 | COMP4121 | Data Structures | Analysis of Algorithms | Core | 3 (3+0) |
| 4 | COMP4115 | | CS- Final Year Project – I | Core | 3 (0+3) |
| 5 | COMP4127 | CS- Final Year Project – I | CS- Final Year Project – II | Core | 3 (0+3) |
| 6 | COMP2111 | PF | Object Oriented Programming (Prerequisite) | Core | 4 (3+1) |
| 7 | COMP2119 | | Computer Networks (Prerequisite) | Core | 3 (2+1) |
| 8 | COMP2112 | | Software Engineering (Prerequisite) | Core | 3 (3+0) |
| Domain Core Courses | | | | | |
| 9 | COMP3145 | | HCI & Computer Graphics | Domain Core | 3 (2+1) |
| 10 | COMP3146 | DB | Advance Database Management Systems | Domain Core | 3 (2+1) |
| 11 | COMP3147 | COAL | Computer Architecture | Domain Core | 3 (2+1) |
| 12 | COMP3148 | | Theory of Automata | Domain Core | 3 (3+0) |

| | | | | | |
|--------------------------------|----------|----|--------------------------------------|-----------------|---------|
| 13 | COMP3149 | TA | Compiler Construction | Domain Core | 3 (2+1) |
| 14 | COMP4122 | OS | Parallel & Distributed Computing | Domain Core | 3 (2+1) |
| Domain Elective Courses | | | | | |
| 15 | COMP3144 | | Web Technologies | Domain Elective | 3 (2+1) |
| 16 | COMP4124 | | Mobile Application Development 1 | Domain Elective | 3 (2+1) |
| 17 | COMP3114 | | Advanced Computer Programming | Domain Elective | 3 (2+1) |
| 18 | ITEC3111 | | Web Engineering | Domain Elective | 3 (2+1) |
| 19 | COMP3143 | | Cyber Security | Domain Elective | 3 (2+1) |
| 20 | COMP4125 | | Software Testing & Quality Assurance | Domain Elective | 3 (2+1) |
| 21 | ITEC4133 | | IOS Application Development | Domain Elective | 3(2+1) |
| 22 | COMP4123 | | Cloud Computing | Domain Elective | 3 (2+1) |
| 23 | ITEC4128 | | Computer Graphics | Domain Elective | 3 (2+1) |
| 24 | COMP3150 | | Object Oriented Analysis and Design | Domain Elective | 3 (2+1) |
| Allied Courses | | | | | |
| 25 | ITEC4152 | | Technical and Business Writing | Allied | 3 (3+0) |
| 26 | STAT2115 | | Introduction to Statistics | Allied | 3 (3+0) |
| 27 | BUSA1113 | | Fundamentals of Accounting | Allied | 3 (3+0) |
| | | | | | |

8. Semester/Study Plan for BSCS (Post ADP)

Course Codes will be assigned considering codes generated in University of Education Lahore Information System (UE, Lahore). The course code for remaining courses will be generated after the approval of Academic Council.

| Semester – I | | | | | |
|-----------------------|-----------------------|-------------------------------|--|-----------------|---------------------|
| Sr # | Code | Prerequisite | Course Title | | Credit Hours |
| 1 | COMP3142 | Data Structures (COMP2117) | Operating Systems | Core | 3(2+1) |
| 2 | COMP3145 | | HCI & Computer Graphics | Domain Core | 3(2+1) |
| 3 | COMP3146 | | Advance Database Management Systems | Domain Core | 3(2+1) |
| 4 | STAT2115 | | Introduction to Statistics | Allied | 3(3+0) |
| 5 | COMP3147 | | Computer Architecture | Domain Core | 3(2+1) |
| 6 | COMP3148 | | Theory of Automata | Domain Core | 3(3+0) |
| | | | Total | | 18(14+4) |
| Semester – II | | | | | |
| Sr # | Code | Prerequisite | Course Title | Domain | Credit Hours |
| 7 | COMP3149 | Theory of Automata (COMP3148) | Compiler Construction | Domain Core | 3(2+1) |
| 8 | COMPXXXX | | Domain Elective-I | Domain Elective | 3 (2+1) |
| 9 | COMPXXXX | | Domain Elective-II | Domain Elective | 3 (2+1) |
| 10 | COMPXXXX | | Domain Elective-III | Domain Elective | 3 (2+1) |
| 11 | COMPXXXX | | Domain Elective-IV | Domain Elective | 3 (2+1) |
| | ISLA1120/ ISLA1121 | | ترجمہ قرآن کورس / وحدت ادیان اور مذاہب عالم | | Non-Credited |
| | | | Total | | 15(10+5) |
| Semester – III | | | | | |
| Sr # | Code | Prerequisite | Course Title | Domain | Credit |

| | | | | | Hours |
|----------------------|-----------------------|--|--|-----------------|---------------------|
| 37 | COMP4121 | Data Structures (COMP2117) | Analysis of Algorithms | Core | 3 (3+0) |
| 38 | COMPXXXX | | Domain Elective-V | Domain Elective | 3 (2+1) |
| 39 | COMPXXXX | | Domain Elective-VI | Domain Elective | 3 (2+1) |
| 40 | COMP4115 | | CS- Final Year Project – I | Core | 3 (0+3) |
| 41 | ITEC4152 | | Technical and Business Writing | Allied | 3 (3+0) |
| | INTN6112 | | Internship | | 3(0+3) |
| | | | Total | | 18(10+8) |
| Semester – IV | | | | | |
| Sr # | Code | Prerequisite | Course Title | Domain | Credit Hours |
| 42 | COMP4122 | Operating Systems (COMP3142) | Parallel & Distributed Computing | Domain Core | 3(2+1) |
| 43 | COMPXXXX | | Domain Elective-VII | Domain Elective | 3(2+1) |
| 44 | COMP4127 | CS- Final Year Project – I (COMP4115) | CS- Final Year Project – II | Core | 3(0+3) |
| 45 | BUSA1113 | | Fundamentals of Accounting | Allied | 3(3+0) |
| | ISLA1120/ ISLA1121 | | ترجمہ قرآن کورس / وحدت ادیان اور مذاہب عالم | | Non-Credited |
| | | | Total | | 12(7+5) |

Note:

The Internship of six to eight weeks is mandatory requirement for the award of degree. Students will take an Internship 3(0+3) course during summer vacations after second semester, and the result of this course will be added in the third semester.

Note: One credit of lab means 3 contact hours, which means students will spend three hours in the lab for each credit earned.

9. Eligibility Criteria, Duration of the Program and Award of Degree:

Admission and degree requirements for Candidates who completed their 14-years of education in the annual system.

- Candidates are required to have completed 14 years of education with Mathematics (200 marks) and Computer Science (200 marks) for admission.
- Candidates are required to have aggregate marks $\geq 45\%$ in the terminal degree for admission.
- The minimum duration for completion of BSCS (Post ADP) is two years (4-regular semester) and the maximum period is three years (6 semesters).
- The following prerequisite courses are compulsory to be completed before completion of BSCS (Post ADP). However, if the candidate has already completed the following courses, they will be exempted.

| Sr# | Course Title | Domain | Credit Hours |
|-----|-----------------------------|--------|--------------|
| 1 | Data Structures | Core | 4(3+1) |
| 2 | Object Oriented Programming | Core | 4(3+1) |
| 3 | Computer Networks | Core | 3 (2+1) |
| 4 | Software Engineering | Core | 3(3+0) |

Admission and degree requirements for candidates who completed their 14-years of education in the semester system.

- Candidates are required to have 14 years of education in the semester system, with 15 Credit Hours in Computer Science and 8 Credit Hours in Mathematics, for admission.
- Candidates with a CGPA less than 2.00 or aggregate marks less than 60% in the semester system are not eligible to apply.
- If a candidate has completed only 6 credit hours of mathematics along with 15 credit hours of computer science in a semester system, they are conditionally eligible. However, they will need to take an additional 3-credit course of mathematics during their BSCS (Post ADP) degree.
- The minimum duration for completion of BSCS (Post ADP) is two years (4-regular semester) and the maximum period is three years (6 semesters).

- The following prerequisite courses are compulsory to be completed before completion of BSCS (Post ADP). However, if the candidate has already completed the following courses, they will be exempted.

| Sr# | Course Title | Domain | Credit Hours |
|-----|-----------------------------|--------|--------------|
| 1 | Data Structures | Core | 4(3+1) |
| 2 | Object Oriented Programming | Core | 4(3+1) |
| 3 | Computer Networks | Core | 3 (2+1) |
| 4 | Software Engineering | Core | 3(3+0) |

Note: All deficiencies and prerequisite courses will be considered as non-credited courses.

The Internship of six to eight weeks is mandatory requirement for the award of degree. Students will take an Internship 3(0+3) course during summer vacations after second semester, and the result of this course will be added in the third semester.

10.Course Outlines

| | |
|------------------------|------------------------------------|
| Course Name: | <i>Object Oriented Programming</i> |
| Course Code: | COMP2111 |
| Credit Hours: | 4 (3+1) |
| Contact Hours: | 3+1 |
| Pre-requisites: | Programming Fundamentals |

Course Introduction:

The course aims to focus on object-oriented concepts, analysis and software development. The basic concept of OOP is covered in this course.

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|--|-----------------------|
| CLO-1 | Understand principles of object oriented paradigm. | C2 (Understand) |
| CLO-2 | Identify the objects & their relationships to build object oriented solution | C3 (Identify) |
| CLO-3 | Model a solution for a given problem using object oriented principles | C3 (Apply) |
| CLO-4 | Examine an object oriented solution | C4 (Examine) |

Course Outline:

Introduction to object oriented design, history and advantages of object oriented design, introduction to object oriented programming concepts, classes, objects, data encapsulation, constructors, destructors, access modifiers, const vs non-const functions, static data members & functions, function overloading, operator overloading, identification of classes and their relationships, composition, aggregation, inheritance, multiple inheritance, polymorphism, abstract classes and interfaces, generic programming concepts, function & class templates, standard template library, object streams, data and object serialization using object streams, exception handling.

Reference Materials: (or use any other standard and latest books)

1. Java: How to Program, 9th Edition by Paul Deitel
 2. Beginning Java 2, 7th Edition by Ivor Horton
 3. An Introduction to Object Oriented Programming with Java, 5th Edition by C. Thomas Wu
 4. Starting Out with C++ from Control Structures to Objects, 9th Edition, Tony Gaddis
 5. C++ How to Program, 10th Edition, Deitel & Deitel.
 6. Object Oriented Programming in C++, 3rd Edition by Robert Lafore
-

Course Name: *Data Structures*
Course Code: COMP2117
Credit Hours: 4 (3+1)
Contact Hours: 3+1
Pre-requisites: Programming Fundamentals

Course Introduction:

The course is designed to teach students structures and schemes, which allow them to write programmer to efficiently manipulate, store, and retrieve data. Students are exposed to the concepts of time and space complexity of computer programs.

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|---|-----------------------|
| CLO-1 | Implement various data structures and their algorithms and apply them in implementing simple applications | C3 (Apply) |
| CLO-2 | Analyze simple algorithms and determine their complexities. | C5 (Analyze) |
| CLO-3 | Apply the knowledge of data structure to other application domains. | C3 (Apply) |
| CLO-4 | Design new data structures and algorithms to solve problems. | C6 (Design) |

Course Outline:

Abstract data types, complexity analysis, Big Oh notation, Stacks (linked lists and array implementations), Recursion and analyzing recursive algorithms, divide and conquer algorithms, Sorting algorithms (selection, insertion, merge, quick, bubble, heap, shell, radix, bucket), queue, dequeuer, priority queues (linked and array implementations of queues), linked list & its various types, sorted linked list, searching an unsorted array, binary search for sorted arrays, hashing and indexing, open addressing and chaining, trees and tree traversals, binary search trees, heaps, M-way tress, balanced trees, graphs, breadth-first and depth-first traversal, topological order, shortest path, adjacency matrix and adjacency list implementations, memory management and garbage collection.

Reference Materials: (or use any other standard and latest books)

1. Data Structures and Algorithm Analysis in Java by Mark A. Weiss
 2. Data Structures and Abstractions with Java by Frank M. Carrano & Timothy M. Henry
 3. Data Structures and Algorithms in C++ by Adam Drozdek
 4. Data Structures and Algorithm Analysis in C++ by Mark Allen Weiss
Java Software Structures: Designing and Using Data Structures by John Lewis and Joseph Chase
-

Course Name: *Operating Systems*
Course Code: COMP3142
Credit Hours: 3 (2+1)
Contact Hours: 2+1
Pre-requisites: Data Structures

Course Introduction:

To help students gain a general understanding of the principles and concepts governing the functions of operating systems and acquaint students with the layered approach that makes design, implementation and operation of the complex OS possible.

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|---------|---|------------------|
| CLO-1 | Understand the characteristics of different structures of the Operating Systems and identify the core functions of the Operating Systems | C2 (Understand) |
| CLO-2 | Analyze and evaluate the algorithms of the core functions of the Operating Systems and explain the major performance issues with regard to the core functions | C5 (Evaluate) |
| CLO-3 | Demonstrate the knowledge in applying system software and tools available in modern operating systems. | C3 (Demonstrate) |

Course Outline:

Operating systems basics, system calls, process concept and scheduling, inter-process communication, multithreaded programming, multithreading models, threading issues, process scheduling algorithms, thread scheduling, multiple-processor scheduling, synchronization, critical section, synchronization hardware, synchronization problems, deadlocks, detecting and recovering from deadlocks, memory management, swapping, contiguous memory allocation, segmentation & paging, virtual memory management, demand paging, thrashing, memory-mapped files, file systems, file concept, directory and disk structure, directory implementation, free space management, disk structure and scheduling, swap space management, system protection, virtual machines, operating system security

Reference Materials: (or use any other standard and latest books)

1. Operating Systems Concepts, 9th edition by Abraham Silberschatz
2. Modern Operating Systems, 4th edition by Andrew S. Tanenbaum
3. Operating Systems, Internals and Design Principles, 9th edition by William Stallings Wu

Course Name: *Analysis of Algorithms*
Course Code: COMP4121
Credit Hours: 3 (3+0)
Contact Hours: 3+0
Pre-requisites: Data Structures

Course Introduction:

Detailed study of the basic notions of the design of algorithms and the underlying data structures. Several measures of complexity are introduced. Emphasis on the structure, complexity, and efficiency of algorithms.

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|---|-----------------------|
| CLO-1 | Explain what is meant by “best”, “expected”, and “worst” case behavior of an algorithm | |
| CLO-2 | Identify the characteristics of data and/or other conditions or assumptions that lead to different behaviors. | |
| CLO-3 | Determine informally the time and space complexity of simple algorithms | |
| CLO-4 | List and contrast standard complexity classes | |
| CLO-5 | Use big O, Omega, Theta notation formally to give asymptotic upper bounds on time and space complexity of algorithms | |
| CLO-6 | Use of the strategies(brute-force, greedy, divide-and-conquer, and dynamic programming) to solve an appropriate problem | |
| CLO-7 | Solve problems using graph algorithms, including single-source and all-pairs shortest paths, and at least one minimum spanning tree algorithm | |
| CLO-8 | Trace and/or implement a string-matching algorithm | |

Course Outline:

Introduction; role of algorithms in computing, Analysis on nature of input and size of input Asymptotic notations; Big-O, Big Ω , Big Θ , little-o, little- ω , Sorting Algorithm analysis, loop invariants, Recursion and recurrence relations; Algorithm Design Techniques, Brute Force Approach, Divide-and-conquer approach; Merge, Quick Sort, Greedy approach; Dynamic programming; Elements of Dynamic Programming, Search trees; Heaps; Hashing; Graph algorithms, shortest paths, sparse graphs, String matching; Introduction to complexity classes.

Reference Materials: (or use any other standard and latest books)

1. Introduction to Algorithms (3rd edition) by Thomas H. Corman, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein
2. Algorithm Design, (1st edition, 2013/2014), Jon Kleinberg, Eva Tardos,
3. Algorithms, (4th edition, 2011), Robert Sedgewick, Kevin Wayne

| | |
|------------------------|---|
| Course Name: | <i>Computer Architecture</i> |
| Course Code: | COMP3147 |
| Credit Hours: | 3 (2+1) |
| Contact Hours: | 2+1 |
| Pre-requisites: | Computer Organization and Assembly Language |

Course Introduction:

The course aims to develop an understanding of design of computer systems and components, Processor design, instruction set design, and addressing; control structures and microprogramming; memory management, caches, and memory hierarchies; and interrupts and I/O structures, Pipelining of processor Issues and Hurdles, exception handling, Parallelism, Multiprocessor Systems.

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|---|-----------------------|
| CLO-1 | Understand functionality of major components of a computer system like CPU, control unit, memory, I/O and storage. | C2 (Explain) |
| CLO-2 | Understand principles of instruction set design including RISC architectures and basic assembly programming | C3 (Discuss) |
| CLO-3 | Understand pipelining and parallelism features applied in single processor, multiple processors and multicore architectures | C4 (Apply) |
| CLO-3 | Solve the problems related to computer architecture | C3 (Apply) |

Course Outline:

Digital Hardware Design: Transistors, Digital logic, Hardware description languages (Verilog). Instruction Set Architecture: Instruction types and mixes, Addressing, RISC vs. CISC, Exceptions. Scalar Pipelines: Data dependencies, Static scheduling, Performance. VLIW Pipelines: Local scheduling, Loop unrolling, Software pipelining, Trace scheduling, Deferred exceptions, Predicated execution, IA64. Dynamic Pipelines: Dynamical scheduling, Register renaming, Speculative execution, Trace cache. Thread-Level Parallelism: Cache coherency, Sequential consistency, Multithreading, Symmetric multiprocessing, Transactional memory. Data-Level Parallelism: GPU programming.

Reference Materials:

1. Hennessy, Patterson, Morgan & Kauffman (2006) Computer Architecture: A Quantitative Approach by Series. (4th. Edition).
2. Dubois et al. (2012). Parallel Computer Organization and Design, Cambridge University Press.
3. Shen, J.P. & Lipasti. H.M. (2005). Modern Processor Design: Fundamentals of Superscalar Processors. (1st edition). McGraw-Hill.
4. Patterson & Hennessy, Morgan & Kauffman Series (2008). Computer Organization Design. (4th. Edition)

Course Name: *Compiler Construction*
Course Code: COMP3149
Credit Hours: 3 (2+1)
Contact Hours: 2+1
Pre-requisites: Theory of Automata

Course Introduction:

Introduction, Lexical Analysis, Parsing, Syntax-Directed Translation, Type Checking, Intermediate Code Generation, Run-Time Organization, Code Generation, Optimization

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|---|-----------------------|
| CLO-1 | Describe the architecture of a compiler, and function of its components | C1(Knowledge) |
| CLO-2 | Choose and apply appropriate formal notations to define a programming language. | C3(Apply) |
| CLO-3 | Design and implement lexical and syntax analyzers by using various algorithms. | C3(Apply) |
| CLO-4 | Select and use appropriate code generation and optimization techniques. | C4(Analyse) |

Course Outline:

Introduction to interpreter and compiler, Structure of a Compiler, Phases of a Compiler, Lexical Analyzer, Input Buffering, Specifications of Tokens, Recognitions of Tokens, Regular Expressions, Finite Automata, Transition Table, Transition Graph. Definitions of Grammars, Derivations, Parse trees, Ambiguity, Associativity and Precedence of Operators, Syntax Analysis, Role of the Parser, Eliminating Ambiguity, Eliminating Left Recursion, Left Factoring. Parsing: Top-Down Parsing, Recursive-Decent Parsing, First and Follow Sets, LL (1) Grammars, Non-recursive Predictive Parsing, Bottom-Up Parsing, Reductions, Shift-Reduce Parsing, LR-Parsing, LR (0) Parsers, LR (0) Automaton and Parsing Table, Shift-Reduce Conflicts, SLR (1) Parsers: SLR (1) Automaton and Parsing Table, LR (1) Parsers: LR (1) Automaton and Parsing Table, LALR Parsing: LALR Automaton and Parsing Table. Semantic Analysis: Intermediate Code Generation, Three Address Code, Tasks of semantic analyzer, Type of Errors. Type Checking and Environments (book2), Type Conversions (implicit vs explicit) (book2), Back patching, Switch Statements Storage Organization, Stack Allocation of Space, Management and Optimization: Heap Management, Code Generation, Design of a code Generator, Target Language, Addresses in the Target Code, Basic Blocks and Flow Graphs Optimization of Basic Blocks

Reference Materials:

1. Compilers: Principles, Techniques & Tools 2nd Ed. By Alfred B. Aho, Monica S. Lam, Ravi Seth
2. Modern Compiler Design by David Gales.

Course Name: *Parallel & Distributed Computing*
Course Code: COMP4122
Credit Hours: 3 (2+1)
Contact Hours: 2+1
Pre-requisites: Operating Systems

Course Introduction:

This is a theory course directed at non-theory students with the standard undergrad background. The goal is to survey the key theory topics that every computer science graduate student should know. In about two weeks for each selected topic, we will gain insights into the basics and study one two example in depth.

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|---------|---|----------------|
| CLO-1 | Learn about parallel and distributed computers. | - |
| CLO-2 | Write portable programs for parallel or distributed architectures using Message-Passing Interface (MPI) library | - |
| CLO-3 | Analyze complex problems with shared memory programming with openMP. | - |

Course Outline:

Asynchronous/synchronous computation/communication, concurrency control, fault tolerance, GPU architecture and programming, heterogeneity, interconnection topologies, load balancing, memory consistency model, memory hierarchies, Message passing interface (MPI), MIMD/SIMD, multithreaded programming, parallel algorithms & architectures, parallel I/O, performance analysis and tuning, power, programming models (data parallel, task parallel, process-centric, shared/distributed memory), scalability and performance studies, scheduling, storage systems, synchronization, and tools (Cuda, Swift, Globus, Condor, Amazon AWS, OpenStack, Cilk, gdb, threads, MPICH, OpenMP, Hadoop, FUSE).

Reference Materials:

1. Distributed Systems: Principles and Paradigms, A. S. Tanenbaum and M. V. Steen, Prentice Hall, 2nd Edition, 2007
2. Distributed and Cloud Computing: Clusters, Grids, Clouds, and the Future Internet, K Hwang, J Dongarra and GC. C. Fox, Elsevier, 1st Ed.

| | |
|------------------------|--------------------------------|
| Course Name: | <i>Web Technologies</i> |
| Course Code: | COMP3144 |
| Credit Hours: | 3 (2+1) |
| Contact Hours: | 2+1 |
| Pre-requisites: | None |

Course Introduction:

In this course, we investigate the origins of hypermedia and the World Wide Web and discuss current and future developments on the Web. The architecture of the Internet and various protocols such as the Hypertext Transfer Protocol (HTTP, client and server-side issues in web architectures, Web 2.0 and Rich Internet Applications (RIAs), latest HTML5, XML technologies and applications are presented. JavaScript is introduced and CSS3, the Semantic Web, web search, security and privacy.

| CLO No. Course Learning Outcomes | Bloom Taxonomy |
|---|-----------------------|
| CLO-1 Sound understanding of basic web technology architectures | C2(Understand) |
| CLO-2 Application of syntax and semantics of several Markup languages for use in information communication. | C3(Apply) |

Course Outline:

Introduction to Web Applications, TCP/IP Application Services. Web Servers: Basic Operation, Virtual hosting, Chunked transfers, Caching support, Extensibility. SGML, HTML5, CSS3. XML Languages and Applications: Core XML, XHTML, XHTML MP. Web Service: SOAP, REST, WML, XSL. Web Services: Operations, Processing HTTP Requests, Processing HTTP Responses, Cookie Coordination, Privacy and P3P, Complex HTTP Interactions, Dynamic Content Delivery. Server Configuration. Server Security. Web Browsers Architecture and Processes. Active Browser Pages: JavaScript, DHTML, AJAX. JSON, Approaches to Web Application Development. Programming in any Scripting language. Search Technologies. Search Engine Optimization. XML Query Language, Semantic Web, Future Web Application Framework.

Reference Materials:

- 1 Learning PHP, MySQL, JavaScript, and CSS, A Step-by-Step Guide to Creating Dynamic Websites By Robin Nixon, O'Reilly Media; Second Edition edition (September 3, 2012). ISBN-10: 1449319262
- 2 Web Technologies: A Computer Science Perspective by Jeffrey C. Jackson, Prentice Hall; 1st Edition (August 27, 2006). ISBN-10: 0131856030
- 3 Web Technologies by Uttam Kumar Roy, Oxford University Press, USA (June 13, 2011). ISBN-10: 0198066228
- 4 Web Application Architecture: Principles, protocols and practices by Leon Shklar and Richard Rosen, Wiley; 2nd Edition (May 5, 2009). ISBN-10: 047051860X.

| | |
|------------------------|--|
| Course Name: | <i>Mobile Application Development 1</i> |
| Course Code: | COMP4124 |
| Credit Hours: | 3 (2+1) |
| Contact Hours: | 2+1 |
| Pre-requisites: | Object Oriented Programming |

Course Introduction:

This course discusses the principles and issues associated with mobile application development using Android as the development platform. The course covers all the basic functionality to get started, as well as experienced knowledge of unique features of Android to enhance existing products or create innovative new ones. Topics covered will include Android application components, UI design, data storage and latest SDK features. Students will develop their own apps in Java using Android Studio in their semester-long projects. Prior knowledge of object oriented programming is required

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|--|-----------------------|
| CLO-1 | Discuss different architectures & framework for Mobile Application development | C1(Discuss) |
| CLO-2 | Develop mobile applications using current software development environments. | C3(Develop) |
| CLO-3 | Compare the different performance tradeoffs in mobile application development. | C3(Compare) |

Course Outline:

Mobiles Application Development Platform; HTML5 for Mobiles; Android OS: Architecture, Framework and Application Development; iOS: Architecture, Framework; Application Development with Windows Mobile; Eclipse; Fragments; Calling Built-in Applications using Intents; Displaying Notifications; Components of a Screen; Adapting to Display Orientation; Managing Changes to Screen Orientation; Utilizing the Action Bar; Creating the User Interface; Listening for UI Notifications; Views; User Preferences; Persisting Data; Sharing Data; Sending SMS Messages; Getting Feedback; Sending E- mail; Displaying Maps; Consuming Web Services Using HTTP; Web Services: Accessing and Creating; Threading; Publishing, Android Applications; Deployment on App Stores; Mobile Programming Languages; Challenges with Mobility and Wireless Communication; Location-aware Applications; Performance/Power Tradeoffs; Mobile Platform Constraints; Emerging Technologies.

Reference Materials:

- 1 Professional Android application development, Reto Meier, Wrox Programmer to Programmer, 2015.
- 2 iOS Programming: The Big Nerd Ranch Guide, Conway, J., Hillegass, A., & Keur, C., 5th Edition, 2014.
- 3 Android Programming: The Big Nerd Ranch Guides, Phillips, B. & Hardy, B., 2nd Edition, 2014.

| | |
|------------------------|--------------------------------------|
| Course Name: | <i>Advanced Computer Programming</i> |
| Course Code: | COMP3114 |
| Credit Hours: | 3 (2+1) |
| Contact Hours: | 2+1 |
| Pre-requisites: | Object Oriented Programming |

Course Introduction:

The course covers the techniques of Java network Programming, advanced graphical user interfaces, event handling techniques and security of applications.

| CLO No. Course Learning Outcomes | Bloom Taxonomy |
|---|-----------------------|
| CLO-1 Use the different elements of a programming language as building blocks to develop correct, coherent programs. | C1(Discuss) |
| CLO-2 Program using the fundamental software development process, including design, coding, documentation, testing, and debugging | C3(Develop) |
| CLO-3 Analyze problems, develop conceptual designs that solve those problems, and transform those designs to Visual Programs. | C4(Analyze) |

Course Outline:

Java API: Abstract classes, Interfaces, Packages, Exception handling, Advance issues of GUI and event handling, Applets and swing, Network Programming Concepts: JDBC, Multithreading, Building Client/ Server, implementation of protocols in Client and Server Programs, RMI. Applets and Network Programming: Java secure socket extension, Secure sockets layer (SSL); SSL socket and SSL server socket classes; Client and Server Authentication: HTTPS, Developing TCP/IP client and server and testing with standard window telnet client and telnet server.

Reference Materials:

- 1 Derek Hamner, Conrad Hughes (2002), Java 2 The Complete Reference, 5th ed. Herbert Schildt, ISBN: 0072224207.
 - 2 Merlin Hughes, Michael Shoffner, Derek Hamner, Conrad Hughes (1999) Java Network Programming: A Complete Guide to Networking, Streams, and Distributed Computing, ISBN: L-884777-49-X.
 - 3 Floyd Marinescu (2002), EJB Design Patterns: Advanced Patterns, Processes, and Idioms, John Wiley & Sons, ISBN: 0471208310.
-

| | |
|------------------------|------------------------|
| Course Name: | Web Engineering |
| Course Code: | ITEC3111 |
| Credit Hours: | 3 (2+1) |
| Contact Hours: | 2+1 |
| Pre-requisites: | Web Technologies |

Course Introduction:

This course introduces the methods and techniques used in Web-based applications development develops an understanding of the concepts, principles, strategies, methodologies and processes involve in web applications development.

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|--|-----------------------|
| CLO-1 | Discuss how web standards impact software development. | C1(Describe) |
| CLO-2 | Describe the constraints that the web puts on developers. | C2(Understand) |
| CLO-3 | Design and Implement a simple web application. | C4(Apply) |
| CLO-4 | Review an existing web application against a current web standard. | C4(Apply) |

Course Outline:

Web programming languages (e.g., HTML5, CSS 3, Java Script, PHP/JSP/ASP.Net), Design principles of Web based applications, Web platform constraints, Software as a Service (SaaS), Web standards, Responsive Web Design, Web Applications, Browser/Server Communication, Storage Tier, Cookies and Sessions, Input Validation, Full stack state management, Web App Security - Browser Isolation, Network Attacks, Session Attacks, Large scale applications, Performance of Web Applications, Data Centers, Web Testing and Web Maintenance.

Reference Materials:

1. Web Engineering, Rajiv Chopra, Prentice-Hall of India, 2016
 2. Web Engineering, Emilia Mendes and Nile Mosley, Springer Verlag, 2010.
 3. Web Engineering: A Practitioners' Approach, Roger S. Pressman, McGraw Hill, 2008.
 4. Dynamic HTML: The Definitive Reference: A Comprehensive Resource for XHTML, CSS, DOM, JavaScript 3rd Edition, O'Reilly Media 2007.
 5. JavaScript: The Definitive Guide, 8th Edition, David Flanagan. O'Reilly Media. 2014.
-

Course Name: *Cyber Security*
Course Code: COMP3143
Credit Hours: 3 (2+1)
Contact Hours: 2+1
Pre-requisites: Information Security

Course Introduction:

This course provides students an introduction to common cyber security threats, vulnerabilities, and risks related to web applications, networks, software and mobile applications. The course provides basic concepts and terminology used in the information and cyber security fields. Moreover, it will also enable students to differentiate between the various forms of malware and how they affect computers and networks.

CLO No. Course Learning Outcomes Bloom Taxonomy

| | | |
|-------|---|----------------|
| CLO-1 | To be able to identify computer system threats | C2(Understand) |
| CLO-2 | To be able to identify Malware attacks, and understand the stages of attack and payloads. | C2(Understand) |
| CLO-3 | Implement various cryptographic techniques and simulate attack scenarios | C3(Apply) |

Course Outline:

Introduction to Cyber security; Networks and the Internet; cyber threat landscape; understanding security; information security Principles (Confidentiality, Integrity, Availability); Information Security Terminology; Who are the attackers; Advanced Persistent Threat (APT); Malware, types of malware; Attacks using malware; Malware Attack Lifecycle: Stages of Attack; Social engineering attacks; types of payload; Industrial Espionage in Cyberspace; Basic cryptography; Web application attacks; Database security; Cyber kill chain; Privacy and anonymity; Network security; Software security; Mobile device security; Mobile app security; Cyber Terrorism and Information Warfare; Introduction to Digital Forensics; Digital Forensics Categories.

Reference Materials:

- 1 Computer Security Fundamentals by Chuck Easttom, 4th edition or latest.
 - 2 Security+ Guide to Network Security Fundamentals, by Mark Ciampa, 5th Edition.
 - 3 Security in Computing by C.P. Pfleeger, Prentice-Hall, 4th Edition or Latest
-

| | |
|------------------------|---|
| Course Name: | <i>Software Testing & Quality Assurance</i> |
| Course Code: | COMP4125 |
| Credit Hours: | 3 (2+1) |
| Contact Hours: | 2+1 |
| Pre-requisites: | None |

Course Introduction:

This course will provide the software testing and software quality assurance principles. Students will learn to prepare test case and test suites for completely testing all aspects of a system under test (SUT) They can analyze which of the software testing techniques are relevant for a particular case and know software reliability analysis tools and techniques.

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|---|-----------------------|
| CLO-1 | Conduct effective and efficient inspections , quality assurance plans, and configuration management plans | C4(Apply) |
| CLO-2 | Design and implement comprehensive test plans | C6(Design) |
| CLO-3 | Assess software process to evaluate how effective it is at promoting quality | C3(Apply) |

Course Outline:

Testing techniques. Black Box testing, White Box and Grey Box testing techniques. Quality Assurance planning and execution. Automated testing topics include constructing a framework, scripting techniques, generating a test data, generating test

Reference Materials:

1. *Software Quality Assurance: Integrating Testing, Security, and Audit* (Internal Audit and IT Audit), Abu Sayed Mahfuz, Auerbach Publications, 2016.
 2. *Practical Model-Based Testing: A Tools Approach*, Mark Utting and Bruno Legeard, Morgan Kaufmann Publishers Inc., San Francisco, CA, 2006.
 3. *Software Quality Engineering, Testing, Quality Assurance, and Quantifiable improvements*, Jeff Tian, IEEE Computer Society, 2005.
 4. *Introduction to Software Engineering*, P Ammann and J Offutt, Cambridge University Press, 2008.
-

| | |
|------------------------|------------------------|
| Course Name: | <i>Cloud Computing</i> |
| Course Code: | COMP4123 |
| Credit Hours: | 3 (2+1) |
| Contact Hours: | 2+1 |
| Pre-requisites: | None |

Course Introduction:

The overall aim of this module is to introduce students to the theory, practice, and advance techniques associated with implementing large-scale distributed computing systems in Service-Oriented Architectures (SOA).

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|--|-----------------------|
| CLO-1 | Explain the core issues of cloud computing such as security, privacy, consistency and interoperability | C1(Describe) |
| CLO-2 | Develop and deploy cloud application using popular cloud platforms | C3(Apply) |
| CLO-3 | Compare the key trade-offs between multiple design approaches used for cloud systems. | C3(Compare) |

Course Outline:

Introduction to cloud computing, Cloud benefits and challenges, Cloud service providers and cloud ecosystem. Concurrency in the cloud, Parallel and distributed systems , Cloud access and cloud interconnection networks , Cloud data storage , Cloud applications , Cloud hardware , Cloud software , Cloud resource management and scheduling ,Cloud security ,Privacy and compliance issues ,Portability and interoperability issues , Big Data, Data streaming and Mobile cloud.

Reference Materials:

1. Cloud Computing: Theory and Practice, Dan C. Marinescu, latest Edition, Morgan Kaufmann.
 2. Cloud Computing, Sandeep Bhowmik, Cambridge University Press, latest edition
-

Course Name: *Object Oriented Analysis and Design*
Course Code: COMP3150
Credit Hours: 3 (2+1)
Contact Hours: 2+1

Pre-requisites:

Course Introduction:

Object-oriented analysis and design (OOAD) is a technical approach for analyzing, designing a system by applying the object-orientated concepts, and develops a set of graphical system models during the development life cycle of the software. OOAD in modern software engineering is typically conducted in an iterative and incremental way. In this course, the students will learn how to produce detailed object models and designs from system requirements; use the modeling concepts provided by UML; identify use cases and expand into full behavioral designs; expand the analysis into a design ready for implementation and construct designs that are reliable.

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|---|-----------------------|
| CLO-1 | Describe Object Oriented Analysis and Design concepts and apply them to solve problems | C1 (Describe) |
| CLO-2 | Prepare Object Oriented Analysis and Design documents for a given problem using Unified Modeling Language | C3 (Apply) |

Course Outline:

Principles of Object Technology. OOP Review. Principles of Modeling. OOA&D Overview. OO Development Process. Requirements Engineering, Analysis, and Specification: Requirements Engineering, Use Cases, Prototyping, Class Models. Interaction Diagrams. Verification and Validation. Architectural and Detailed Design. Class Diagrams. Interaction Diagrams. State Machines and Diagrams. Implementation, Package Diagrams. Activity Diagrams. OO Patterns, Verification and Validation. Note: Students may also be introduced to Object Diagram, Component Diagram, Package Diagram, Deployment Diagram, Network Diagram.

Reference Materials:

1. Applying UML and patterns: An introduction to Object-Oriented Analysis and Design and Iterative Development by Craig Larman, Prentice Hall; 3rd Edition (October 30, 2004). ISBN-10: 0131489062
 2. Using UML: Software Engineering with Objects and Components by Perdita Stevens, Addison-Wesley; 2nd Edition (February 13, 2006). ISBN-10: 0321269675
 3. Fundamental of Object-Oriented Design in UML by Meiler Page-Jones, Addison Wesley, 2000. ISBN: 020169946X.
 4. The Unified Modeling Language User Guide by G. Booch, J. Rumbaugh and I. Jakobson, Addison-Wesley Professional; 2nd Edition (2005). ISBN- 10: 0321267974.
-

Course Name: Mobile Application Development 2
Course Code: COMP4126
Credit Hours: 3 (2+1)
Contact Hours: 2+1
Pre-requisites:

Course Introduction:

The main objectives of this course is to Build iOS native applications for iPhone and iPad using different frameworks

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|---------|---|----------------|
| CLO-1 | Discuss architecture & framework for Mobile Application development for iPhones | C1(Discuss) |
| CLO-2 | Develop mobile applications using current software development environments. | C3(Develop) |
| CLO-3 | Compare the different performance tradeoffs in mobile application development. | C3(Compare) |

Course Outline:

Creating an iOS App: Understanding Xcode, Using the Xcode interface builder, Using the Xcode objects library, Understanding view hierarchy, Creating a custom app icon. Outlets, Actions, and Views: Understanding outlets and Actions, using text fields, Buttons, Labels, Web Views, and page controllers, Using Views along with sub Views, Creating Views using code. Using View Controllers: Working with the single View template, Exploring the app delegate, Adding new View Controllers, and Transitioning between multiple View, Controllers, and Using animations. Application Templates: Working with the tabbar application, Template, Understanding the master detail, Application template. The iOS Keyboard: Customizing the iOS keyboard for different inputs, Adjusting text field behaviors, Methods for dismissing the keyboard, Detect keyboard activities with the notification center, Using the scroll view, Responding to keyboard activities programmatically using scrolling views. Working with Different Ios Devices; iPhone & iPad: Programmatically detecting device hardware, Dynamically adjusting graphical layouts, Working with multiple devices, Creating apps that work universally. Using Table Views :Understanding the UI Table View, Using UI Table View Cell Classes, Working with UI Table View data source and delegate, Using Table View and Table View Cell, The master detail template, Creating drill-down menus, Navigation to other views, Using Property lists for data persistence, Creating multi section tables. Supporting Screen Rotations: Portrait & landscape modes, Handling device rotation setting preferred device orientation, forcing specific orientation using no Rotations, Dynamically adjusting graphical layouts based upon rotation. Working with Databases: Importing the sqlite3 library. Creating a database, writing tables and inserting records into tables, Bundling a database with your app. Checking for database existence. Reading and displaying database data. Using Animations & Video: Using the NS Timer Class. Animating objects on the screen using transformation, Rotation, Scaling and Translation, animating image arrays Playing video within an app. Accessing integrated iOS Apps: Using the email app accessing Safari. Sending SMS text messages, Working with the camera, Using with the Photo Library. Using Web Services within an iOS app: Consuming a Web Service, Parsing XML

consuming and parsing JSON, Web Services, Integrating common Twitter and Facebook with iOS apps. Working with iOS Maps and Location Services: Using the Map Kit and UI Map View getting and displaying user location, Getting directional information, Displaying map annotations. Displaying disclosure buttons on annotations performing reverse Geo coding. Working with iCloud, Working with the Accelerometer: Using the Gyroscope, Using the Accelerometer, Outputting Sensor Data, Using the Shake API.

Reference Materials:

- 1 Keur, C. &Hillegass, A., iOS Programming:The Big Nerd Ranch,(5th ed.)
 - 2 Nahavandipoor, V.,iOS 7 Programming Cookbook,(1st, ed.), Vandad.
-

| | |
|------------------------|-----------------------------------|
| Course Name: | <i>Fundamentals of Accounting</i> |
| Course Code: | BUSA1113 |
| Credit Hours: | 3 (3+0) |
| Pre-requisites: | None |

Learning Outcomes:

After studying this course, the students will be able to:

- Understand the significance, objectives, scope and use of accounting information
- Explore conceptual issues related to accounting and ethics,
- Comprehend the basic accounting concepts and conventions and qualitative aspect of financial report
- Develop an understanding of what the basic financial reports communicate to its readers and how they are prepared in business

Course Outline:

Introduction to Accounting and Business:

Nature of Business and Accounting, Types of Businesses, Types of Business Organization, Users of Accounting Information, Role of Ethics in Business, Role of Accounting in Business, Profession of Accounting

Fundamental Accounting Concepts, Principles and Policies:

The Business Entity Concept, The Reliability (or Objectivity) principle, Historical Cost Convention, Substance Over Form, The Fair Value Principle, The Going Concern Assumptions, The Realization Principle, The Matching Principle, Money Measurement (Stable Dollar Assumption), Materiality

Financial Statements:

Business Transactions and The Accounting Equation, Effects of Business Transactions on Accounting Elements, Set of Financial Statements, Definition of Income Statement, Components of Income Statement : Revenues, Expenses, Gains and Losses, Accounting for Revenues and Expenses, Statement of Owner's Equity,Definition of Balance Sheet, Components of Balance Sheet: Assets, Liabilities, Equity, Statement of Cash Flows,

Operating, Investing and Financing Activities, Direct Method, Interrelationships Among Financial Statements

The Recording Process:

Accrual Basis and Cash Basis of Accounting, Chart of Accounts, Phases in Accounting Cycle, Account and its Recording Process, Types of Accounts – Permanent and Temporary, Double Entry Book Keeping System, Rules of Debit and Credit, Introduction to General Journal and Special Journals, T-Ledger, Posting Journal Entries to Accounts, Analyzing and Summarizing Transactions in Accounts, Normal Balances of Accounts, Need and Objective of Trial Balance, Preparing Trial Balance

The Adjusting Process:

Types of Adjusting Entries, Recording Adjusting Entries – Deferrals, Recording Adjusting Entries – Accruals, Depreciation Expense, Summary of Adjustment Process, Nature, Purpose and Preparation of Adjusted Trial Balance Completing the Accounting Cycle: Flow of Accounting Information, Journalizing and Posting, Closing Entries, PostClosing Trial Balance, Adequate Disclosure and Types of Information to be Disclosed, Income Statement, Statement of Owner's Equity, Balance Sheet, Illustrations and Questions

Accounting for Merchandising Operations:

Service Companies and Merchandising Companies, Operating Cycle of Merchandising Companies, Income Statement of Merchandising Companies, Perpetual Inventory System, Periodic Inventory System, Merchandising Transactions, Transactions related to Sales and Purchase, Evaluating the Performance of Merchandising Companies

Reference Materials:

1. Financial & Managerial Accounting, Williams , Haka , Bettner, Prentice Hall
2. Business Accounting, I, Fank Woods
3. Financial Accounting and Reporting, Barry Elliot ,Jame Elliot, Prentice Hall
4. Financial Accounting, A. Mukherjee and M. Hanif

| | |
|------------------------|---|
| Course Name: | <i>Introduction to Entrepreneurship</i> |
| Course Code: | BUSA1114 |
| Credit Hours: | 2 (2+0) |
| Pre-requisites: | None |

Course Description

This two-credit hour course is designed for students from diverse disciplines such as information technology, physical sciences, education, and other non-business backgrounds. The course aims to introduce students to the fundamental concepts of entrepreneurship, innovation, and business development, providing them with the necessary skills and knowledge to explore entrepreneurial opportunities and create innovative ventures. Instructors are encouraged to quote examples and case studies relevant to the disciplines they are teaching.

Learning Outcomes

1. Understand the key concepts and importance of entrepreneurship in non-business contexts.
2. Demonstrate an entrepreneurial mindset by embracing creativity, innovation, and risk-taking.
3. Apply entrepreneurial principles to identify opportunities and create business plans tailored to their disciplines.
4. Analyze and discuss relevant examples and case studies from various fields to enhance their understanding of entrepreneurship in practice.

Course Outlines

Introduction to Entrepreneurship: Defining entrepreneurship and its importance, Identifying common misconceptions about entrepreneurship, Entrepreneur vs a businessperson

Entrepreneurial Mind Set: Cultivating an entrepreneurial mindset, How entrepreneur think? Entrepreneur background and personality, Embracing creativity and innovation

Creativity and Innovation: Importance of creativity and innovation in entrepreneurship, Techniques for generating innovative ideas

Product Planning: Developing new product ideas based on trends, consumer preferences, and research, Stages of product development

Opportunity in domestic market: Market research on potential target markets, analyzing customer needs and preferences, Identifying potential competitors

Opportunity in International Market: Motivations to go global, Methods of entry into international markets, Overview of trade agreements

Protecting Idea and Innovation: What is intellectual property? Trademarks, patents, copyrights and licensing

The Business Plan: Importance of a business plan for stakeholders and as a roadmap, Skills required to write a business plan, Outline of a general business plan

Marketing and Sales Strategies: Understanding marketing mix, Various means of marketing and communication (e.g. digital marketing)

Organizational Plan: Developing the managing team, Legal forms of the business

Sources of capital for entrepreneurs (emphasis on sources in Pakistan)

Importance of budgeting and break-even analysis,

Common challenges faced by entrepreneurs in Pakistan, developing resilience in the face of failure, Learning from setbacks

Final Presentations and Reflection, Students present their business idea and plans

Recommended Books:

1. Hisrich, R. D., Peters, M. P., & Shepherd, D. A. *Entrepreneurship* (Latest edition.). Boston: McGraw-Hill/Irwin.
2. Scarborough, N. and Cornwall, J. (2016). *Essentials of entrepreneurship and small business management*. 8th ed. Harlow: Pearson Education Limited.
3. Naqvi, S.M., Entrepreneurship

Course Name: *Computer Networks*
Course Code: COMP2119
Credit Hours: 3 (2+1)
Pre-requisites: None

Course Introduction:

This course introduces the basic concept of computer network to the students. Network layers, Network models (OSI, TCP/IP) and protocol standards are part of the course.

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|--|-----------------------|
| CLO-1 | Describe the key terminologies and technologies of computer networks | C2 (Describe) |
| CLO-2 | Explain the services and functions provided by each layer in the Internet protocol stack. | C2 (Explain) |
| CLO-3 | Identify various internetworking devices and protocols and their functions in a networking | C4 (Identify) |
| CLO-4 | Analyze working and performance of key technologies, algorithms and protocols | C4 (Analyze) |
| CLO-5 | Build Computer Network on various Topologies | P3 (Build) |

Course Outline:

Introduction and protocols architecture, basic concepts of networking, network topologies, layered architecture, physical layer functionality, data link layer functionality, multiple access techniques, circuit switching and packet switching, LAN technologies, wireless networks, MAC addressing, networking devices, network layer protocols, IPv4 and IPv6, IP addressing, sub netting, CIDR, routing protocols, transport layer protocols, ports and sockets, connection establishment, flow and congestion control, application layer protocols, latest trends in computer networks.

Reference Materials: (or use any other standard and latest books)

1. Computer Networking: A Top-Down Approach Featuring the Internet, 6th edition by James F. Kurose and Keith W. Ross
2. Computer Networks, 5th Edition by Andrew S. Tanenbaum
3. Data and Computer Communications, 10th Edition by William Stallings
4. Data Communication and Computer Networks, 5th Edition by Behrouz A. Forouzan

Course Name: *Software Engineering*
Course Code: COMP2112
Credit Hours: 3 (3+0)
Contact Hours: 3+0
Pre-requisites: None

Course Introduction:

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|--|-----------------------|
| CLO-1 | Describe various software engineering processes and activates | C1 (Describe) |
| CLO-2 | Apply the system modeling techniques to model a medium size software systems | C3 (Apply) |
| CLO-3 | Apply software quality assurance and testing principles to medium size software systems | C4 (Apply) |
| CLO-4 | Discuss key principles and common methods for software project management such as scheduling, size estimation, cost estimation and risk analysis | C2 (Discuss) |

Course Outline:

Nature of Software, Overview of Software Engineering, Professional software development, Software engineering practice, Software process structure, Software process models, Agile software Development, Agile process models, Agile development techniques, Requirements engineering process, Functional and non-functional requirements, Context models, Interaction models, Structural models, behavioral models, model driven engineering, Architectural design, Design and implementation, UML diagrams, Design patterns, Software testing and quality assurance, Software evolution, Project management and project planning, configuration management, Software Process improvement

Reference Materials: (or use any other standard and latest books)

1. Software Engineering, Sommerville I., 10th Edition, Pearson Inc., 2014
2. Software Engineering, A Practitioner's Approach, Pressman R. S.& Maxim B. R., 8th Edition, McGraw-Hill, 2015.

Course Name: *Advance Database Management Systems*
Course Code: COMP3146
Credit Hours: 3 (2+1)
Pre-requisites: Database Systems

Course Introduction:

Advanced Database Management Systems is an extension to “Database Systems” course. The aim of the course is to enhance the previous knowledge of database systems by deepening the understanding of the theoretical and practical aspects of the database technologies, and showing the need for distributed database technology to tackle deficiencies of the centralized database systems. Moreover, it focuses to introduce the basic principles and implementation techniques of distributed database systems, and expose emerging research issues in database systems and application development.

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|---|-----------------------|
| CLO-1 | Understanding advance data models, technologies and approaches for building distributed database systems. | C2 (Understand) |
| CLO-2 | Applying the models and approaches in order to become enabled to select and apply appropriate methods for a particular case | C3 (Apply) |
| CLO-3 | To develop a database solution for a given scenario/ challenging problem in the domain of distributed database systems. | C3 (Apply) |

Course Outline:

Introduction to advance data models such as object relational, object oriented. File organizations concepts, Transactional processing and Concurrency control techniques, Recovery techniques, Query processing and optimization, Database Programming (PL/SQL, T-SQL or similar technology), Integrity and security, Database Administration (Role management, managing database access, views), Physical database design and tuning, Distributed database systems, Emerging research trends in database systems, MONGO DB, NO SQL (or similar technologies)

Reference Materials:

1. Database Systems: A Practical Approach to Design, Implementation, and Management, 6th Edition by Thomas Connolly and Carolyn Begg
2. Database Management Systems, 3rd Edition by Raghuram Ramakrishnan, Johannes Gehrke
3. Database System Concepts, 6th Edition by Avi Silberschatz, Henry F. Korth and S. Sudarshan.
4. Database Systems: The Complete Book, 2nd Edition by Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom

Course Name: *Introduction to Statistics*
Course Code: **STAT2115**
Credit Hours: 3 (3+0)
Pre-requisites: None

Course Objectives:

- To Develop Understanding for the Basic Concepts of Statistics.
- To Prepare the Students for the Advanced Courses in the Field of Statistics Like Statistical Inference, Sampling, Hypothesis Testing etc.
- At the End of Course, Students Should Have Capability of Critical Thinking about the Data and which Techniques Could be used for Analysis.

Course Content:

Scope of Statistics, Introduction to Basic Concepts of Statistics Like Descriptive and Inferential Statistics, Population, Sample, Parameter, Statistic, Types of Data and the Scales of Measurement, Frequency Distribution and Graphical Representation of Data (Bar Chart, Pie Chart, Histogram, Frequency Polygon and Frequency Curve, Cumulative Frequency Polygon), Measures of Central Tendency, Quantiles, Absolute and Relative Measures of Dispersion, Moments, Skewness and Kurtosis, Basic Concepts of Probability, Counting Rules (Multiplication Principle, Permutation and Combination), Probability Spaces, Laws of Probability, Conditional Probability, Bayes' Theorem, Discrete and Continuous Random Variables, Probability Distributions of Random Variables (Binomial Distribution, Poisson Distribution, Hypergeometric Distribution, Uniform Distribution, Exponential Distribution, Normal Distribution Etc.), Overview of Sampling Like Sample Design, Sampling Frame, Sampling and Non-Sampling Errors, Sampling Distributions for Mean, Proportion, Difference of Means and Difference of Proportions, Overview of Hypothesis Testing and Regression Analysis.

Recommended Books

1. Bluman, A. G. *Elementary Statistics: A Step-by-Step Approach*, McGraw-Hill, 2012.
2. Mood, A. M., Graybill, F. A. & Boes, D.C. *Introduction to The Theory of Statistics*, McGraw-Hill, New York (1997)

| | |
|------------------------|---------------------------|
| Course Name: | <i>Theory of Automata</i> |
| Course Code: | COMP3148 |
| Credit Hours: | 3 (3+0) |
| Pre-requisites: | None |

Course Introduction:

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|---|-----------------------|
| CLO-1 | Explain and manipulate the different concepts in automata theory and formal languages such as formal proofs, automata, regular expressions, Turing machines etc. | C2 (Understand) |
| CLO-2 | Prove properties of languages, grammars and automata with rigorously formal mathematical methods | C2 (Understand) |
| CLO-3 | Design of automata, RE and CFG | C3 (Apply) |
| CLO-4 | Transform between equivalent NFAs, DFAs and REs | C3 (Apply) |
| CLO-5 | Define Turing machines performing simple tasks | C2 (Understand) |
| CLO-6 | Differentiate and manipulate formal descriptions of languages, automata and grammars with focus on regular and context-free languages, finite automata and regular expressions. | C3 (Apply) |

Course Outline:

Finite State Models: Language definitions preliminaries, Regular expressions/Regular languages, Finite automata (FAs), Transition graphs (TGs), NFAs, Kleene's theorem, Transducers (automata with output), Pumping lemma and non-regular language Grammars and PDA: CFGs, Derivations, derivation trees and ambiguity, Simplifying CFLs, Normal form grammars and parsing, Decidability, Context sensitive languages, grammars and linear bounded automata (LBA), Chomsky's hierarchy of grammars Turing Machines Theory: Turing machines, Post machine, Variations on TM, TM encoding, Universal Turing Machine, Defining Computers by TMs.

Reference Materials:

- 1 Introduction to computer theory, Daniel I. A. Cohen, 2nd Edition
- 2 Automata, Computability and Complexity: Theory and Applications, by Elaine Rich, 2011
- 3 An Introduction to Formal Languages and Automata, by Peter Linz, 4th edition, Jones & Bartlett Publishers, 2006
- 4 Theory of Automata, Formal Languages and Computation, by S. P. Eugene, Kavier, 2005, New Age Publishers

| | |
|------------------------|------------------------------------|
| Course Name: | <i>HCI & Computer Graphics</i> |
| Course Code: | COMP3145 |
| Credit Hours: | 3 (2+1) |
| Pre-requisites: | None |

Course Introduction:

This course introduces the fundamental concepts of human computer interaction and computer graphics. It describes the implications of human understanding on the usability of computer systems and the importance of understanding the context of use. This course will also develop design and problem solving skills with applications to computer graphics.

| CLO No. | Course Learning Outcomes | Bloom Taxonomy |
|----------------|---|-----------------------|
| CLO-1 | Describe the concepts of human understanding on the usability of computer systems and the importance of understanding the context of use. | C2 (Describe) |
| CLO-2 | Explain the basic principles of implementing computer graphics fundamentals | C2 (Explain) |
| CLO-3 | Analyze and identify usability issues in User interfaces according to the standards. | C4 (Analysis) |
| CLO-4 | Develop the design and problem solving skills with applications to computer graphics | C3(Develop) |
| CLO-5 | Construct interactive computer graphics programs | C3 (Apply) |

Course Outline:

The Human: Input-output channels, Human memory, Thinking, Reasoning, Problem solving, Emotions, Individual differences, Psychology and design of interacting systems. **The Computer:** Introduction, Text entry devices, Positioning, Pointing, and drawing, Display devices, Devices for virtual reality and 3D interaction, Physical controls, Sensors and special devices, Paper printing and scanning, Memory, Processing and networks. **The Interaction:** Models of interaction, Frameworks and HCI, Ergonomics, Interaction styles, Elements of the WIMP interfaces, Interactivity, Context of interaction, Experience. **Usability Paradigm and Principles:** Introduction, Paradigms for interaction. **Interaction Design Basics:** Introduction, What is design, Process of design, User focus, Navigation design, Screen design and layout, Iteration and prototyping. **HCI in Software Process:** Introduction, Software life cycle, Usability engineering, Iterative design and prototyping, Design rationale. Design rules, Prototyping, Evaluation techniques, Task analysis, Universal design and User support and Computer Supported Cooperative Work: Guidelines, Golden rules and heuristics, HCI patterns, Choosing an evaluation method, Requirements of user support, Applications, Design user support systems. **Introduction to Groupware, Pervasive and Ubiquitous Applications:** Introduction, Groupware systems, Implementation of synchronous groupware, Ubiquitous computing. **History of Computer Graphics,** Graphics architectures and software, Imaging, Pinhole camera, Human vision, Synthetic camera, Modeling vs. rendering. **OpenGL:** Architecture, Displaying simple two-dimensional geometric objects, Positioning systems, Working in a windowed environment. **Color:** Color perception, color models (RGB, CMY, HLS), color transformations. Color in OpenGL. RGB and indexed color. **Input:** Working in a network environment, client-server computing; input measure, event, sample and request input, using callbacks, picking. **Geometric**

transformations: Affine transformations (translation, rotation, scaling, shear), homogeneous coordinates, concatenation, current transformation and matrix stacks. Three Dimensional Graphics: Classical three dimensional viewing, Specifying views, Affine transformation in 3D, Projective transformations. Ray tracing. Shading: Illumination and surface modeling, Phong shading model, Polygon shading. Rasterization: Line drawing via Bresenham's algorithm, clipping, polygonal fill, BitBlt. Introduction to hidden surface removal (z buffer). Discrete Techniques: Buffers, reading and writing bitmaps and pixel maps, texture mapping, compositing.

Reference Materials:

1. Janet E. Finlay, Leeds Metropolitan. (2000). Human-Computer Interaction, Alan Dix, Computing Dept, Lancaster University, Birmingham Publisher: PrenticeHall.
2. Ben Shneiderman, University of Maryland Catherine Plaisant. (2010). Designing the User Interface: Strategies for Effective Human-Computer Interaction, 4/E, University Maryland. Publisher: Addison-Wesley.
3. Computer Graphics with Open GL (4th Edition) by Donald D. Hearn, Prentice Hall, 2010, ISBN-10: 0136053580.
4. Foundations of 3D Computer Graphics by S. J. Gortler, The MIT press, 2012.
5. Fundamentals of Computer Graphics, 3rd Edition, A K Peters, 2009.
6. Computer Graphics: Principles and Practice, 3rd Edition

Course Name: *Computer Graphics*
Course Code: ITEC4128
Credit Hours: 3 (2+1)
Pre-requisites: None

Course Content:

Introduction to Computer Graphics, Graphics Systems, Point, Line Drawing Techniques, Circle Drawing Techniques, Ellipse and Other Curves, 2D Transformations, Clipping, 3D Concepts, 3D Transformations, Perspective Projection, Triangles & Planes, Triangle Rasterization, Lighting, Introduction to OpenGL, Animations.

Reference Material:

1 Computer Graphics using OpenGL Author: F.S. Hill Jr. & Stephen M. Kelley Jr. 3rd Edition
ISBN: 0023548568

2. Computer Graphics with OpenGL by Donald Hearn and M. Pauline Baker 2nd Edition

| | |
|------------------------|--|
| Course Name: | <i>Technical and Business Writing</i> |
| Course Code: | ITEC4152 |
| Credit Hours: | 3 (3+0) |
| Pre-requisites: | None |

Course Introduction

Students in the senior level needs good technical writing skills not only for writing project report but also useful for them to communicate their resume and get place in the market. This is a high level course which provide useful knowledge to the students for writing proposals etc. Further, the course aims at augmenting students' proficiency in technical writing in order to sensitize them to the dynamics, challenges, and needs of the modern world characterized by technologically advanced social, cultural, and corporate settings. It will focus on students' ability to effectively convey and exchange information in cross-cultural, international, and multinational milieu necessitated by the emergence of global society.

Course Outline:

Overview of technical reporting, use of library and information gathering, administering questionnaires, reviewing the gathered information; Technical exposition; topical arrangement, exemplification, definition, classification and division, casual analysis, effective exposition, technical narration, description and argumentation, persuasive strategy, Organizing information and generation solution: brainstorming, organizing material, construction of the formal outline, outlining conventions, electronic communication, generation solutions. Polishing style: paragraphs, listening sentence structure, clarity, length and order, pomposity, empty words, pompous vocabulary, document design: document structure, preamble, summaries, abstracts, table of contents, footnotes, glossaries, cross referencing, plagiarism, citation and bibliography, glossaries, index, appendices, typesetting systems, creating the professional report; elements, mechanical elements and graphical elements. Reports: Proposals, progress reports, Leaflets, brochures, handbooks, magazines articles, research papers, feasibility reports, project reports, technical research reports, manuals and documentation, thesis. Electronic documents, Linear verses hierarchical structure documents.

Reference Material

1. Technical Report Writing, by Pauley and Riordan, Houghton Mifflin Company, 8th Edition.
2. Effective Technical Communication by Ashraf Rizvi, Tata McGraw-Hill

Course Name: *Internship*
Course Code: INTN6112
Credit Hours: 3 (0+3)
Pre-requisites: None

Internship Policy

| | |
|-------------------------|--|
| Internship Requirement | UE includes an internship, as a mandatory part for BSCS/BSIT/BSCS Post ADP/ BSIT Post ADP programs. Students may need to complete a minimum duration of practical training in relevant field. |
| Internship Placement | UE has a Student Counseling Center (SCC) that assists students in finding Internship opportunities. The SCC may collaborate with local industries, Companies or organizations for the student's internship placements. |
| Internship Duration | Minimum duration is 6 weeks. |
| Academic Credit | Its mandatory for degree with credit hours 3(0+3) |
| Evaluation & Assessment | Upon completion of internship, students are required to submit internship report and completion certificate that reflects experiences in relevant field. |
| Eligibility Criteria | As per university policy. |
| Internship Supervision | Faculty members will be assigned a group of students and he/she will visit Internee students once during internship period and shall take written feedback from Industry regarding skill set of the students and submit a written report to the coordinator regarding his/her visit. |

Evaluation Criteria

| Evaluation | Contribution |
|---|--------------|
| Site Supervisor Evaluations | 40% |
| Students Reports | 30% |
| Faculty Supervisor Evaluation (Student internship report and completion certificate) | 25% |
| Student's activity log completion | 05% |

Seerat of the Holy Prophet Muhammad (SAW)

| Title | Description |
|--------------------------|--|
| Semester | Student can opt this course in any Semester |
| Nature of Course | Regular |
| Course Code | ISLA2211 |
| No. of Cr. Hrs. | 2 (2+0) |
| Total Teaching weeks | 16 Weeks |
| Objectives of the Course | <p>۱۔ طلباء کو مطالعہ سیرۃ طیبہ کی ضرورت و اہمیت سے آگاہ کرنا</p> <p>۲۔ تعمیر شخصیت میں مطالعہ سیرۃ طیبہ کے کردار کو واضح کرنا</p> <p>۳۔ بعثت نبوی کے موقع پر اقوام عالم کی عمومی صورت حال سے آگاہ کرنا</p> <p>۴۔ طلباء کو عہد نبوی کی معاشرت، سیاست، معیشت سے آگاہ کرنا</p> |

Course Description

| S.No. | Title | Description |
|-------|--|---|
| 1 | سیرۃ النبی صلی اللہ علیہ وسلم: تعارف و اہمیت | <p>۱۔ مطالعہ سیرت کی معاشرتی و سماجی اہمیت</p> <p>۲۔ مطالعہ سیرت کی معاشی و سیاسی اہمیت</p> <p>۳۔ مطالعہ سیرت کی عقلی و منطقی اور استدلالی حوالے سے اہمیت</p> <p>۴۔ مطالعہ سیرت کے شخصیت و کردار پر اثرات</p> |
| 2 | پیغمبر اسلام صلی اللہ علیہ وسلم کی زندگی کا مطالعہ کیوں کیا جائے؟ | <p>۱۔ مادی اور بنیادی ذرائع</p> <p>۲۔ ماحول اور حالات</p> <p>۳۔ اعلیٰ خدائی مشن کے لیے محمد صلی اللہ علیہ وسلم کا انتخاب</p> <p>۴۔ حرب فجار اور حلف الفضول</p> |
| 3 | اللہ تعالیٰ کے پیغام کی تبلیغ و اشاعت | <p>۱۔ ہجرت حبشہ</p> <p>۲۔ معاشرتی بائیکاٹ</p> <p>۳۔ معراج اور معجزات</p> |
| 4 | اسلام میں خواتین کا کردار | <p>۱۔ اسلام میں خواتین کا کردار قبل از ہجرت</p> <p>۲۔ اسلام میں خواتین کا کردار بعد از ہجرت</p> <p>۳۔ قومی شیرازہ بندی میں عورت کا کردار</p> |
| 5 | ہجرت مدینہ اور انصار و یہود | <p>۱۔ مؤاخات مدینہ کی عصر حاضر میں اہمیت</p> <p>۲۔ میثاق مدینہ کے تناظر میں یہود و نصاریٰ کے ساتھ</p> |

| | | معاشرتی تعلقات |
|----|--|---|
| 6 | غزوات نبوی کے اثرات | ۱۔ غزوات نبوی کے سیاسی اور دفاعی اثرات ۲۔ غزوات نبوی کے معاشی اثرات ۳۔ غزوات نبوی کے سماجی اور دعوتی اثرات |
| 7 | حضور صلی اللہ علیہ وسلم کے داخلی سیاسی اقدامات | ۱۔ یہود سے تعلقات ۲۔ مشرکین اور منافقین سے تعلقات ۳۔ قبائل عرب سے تعلقات |
| 8 | حضور صلی اللہ علیہ وسلم کے خارجی تعلقات | ۱۔ غیر ملکی سربراہوں سے حضور صلی اللہ علیہ وسلم کی مراسلات ۲۔ غیر ملکیوں کے ساتھ وفود کا تبادلہ ۳۔ فتح مکہ۔ دعوت اسلامی کا فروغ ۴۔ حضور صلی اللہ علیہ وسلم کی خارجی سیاست کے اثرات |
| 9 | غیر مسلم اقلیت اور سیرت طیبہ: ایک مطالعہ | ۱۔ غیر مسلم اقلیت کا تعارف ۲۔ اقلیتوں کے حقوق کا سیرتی مطالعہ ۳۔ اسلامی ریاست میں غیر مسلموں کے حقوق و فرائض ۴۔ اقلیتوں کے حقوق اور اسلاموفوبیا |
| 10 | عہد نبوی۔ تہذیبی مطالعہ | ۱۔ عہد نبوی کا معاشرتی و معاشی نظام ۲۔ عہد نبوی کا مذہبی نظام ۳۔ عہد نبوی کا نظام حکومت |
| 11 | نبی کریم ﷺ کی تعلیمی جدوجہد | ۱۔ نشر علم کے لیے ایک جماعت کا قیام ۲۔ خواتین کے لیے تعلیم کا اہتمام ۳۔ نسل نو کے لیے تعلیم کا انتظام ۴۔ تشکیل و فود برائے توسیع و استحکام تعلیم |

نصابی کتب

| نمبر شمار | نام مؤلف | نام کتاب |
|-----------|------------------------------------|-------------------------------|
| 1 | ابن ہشام | السیرة النبویة |
| 2 | مولانا شبلی نعمانی، سید سلمان ندوی | سیرة النبی صلی اللہ علیہ وسلم |
| 3 | قاضی محمد سلیمان سلمان منصور پوری | رحمة للعالمین |
| 4 | مولانا سید ابوالحسن علی ندوی | نبی رحمت صلی اللہ علیہ وسلم |
| 5 | ڈاکٹر یسین مظہر صدیقی | عہد نبوی کا نظام حکومت |
| 6 | ڈاکٹر خالد علوی | انسان کامل |

حوالہ جاتی کتب

| نام کتاب | نام مؤلف | نمبر شمار |
|-----------------------------------|-----------------------------|-----------|
| السيرة النبوية الصحيحة | ڈاکٹر اکرم الضیاء العمری | 1 |
| اصح السیر | مولانا عبدالرؤف داناپوری | 2 |
| الرحیق المختوم | مولانا صفی الرحمن مبارکپوری | 3 |
| ضیاء النبی صلی اللہ علیہ وسلم | پیر محمد کرم شاہ الازہری | 4 |
| سیرت سرور عالم صلی اللہ علیہ وسلم | سید ابوالاعلیٰ مودودی | 5 |
| آنحضور ﷺ کی تعلیمی جدو جہد | پروفیسر رب نواز | 6 |

11. FYP Deliverable 1 and Deliverable 2 Guidelines

| Name of Deliverable | Contents | Remarks |
|-----------------------|--|--|
| Deliverable-I | <ul style="list-style-type: none"> • Project Proposal • Chapter-1: Gathering & Analyzing Information • Chapter-2: Software Requirement Specification (Sample Attached) • Chapter-3: Analysis • Chapter-4: Design | Deliverable I should be submitted in the second last week of the 7 th /3 rd Semester of BS/ BS (Post ADP)/MSC. |
| Deliverable-II | <ul style="list-style-type: none"> • Chapter-5: Graphical User Interfaces • Chapter-6: Testing • Chapter-7: Conclusion and Future Work • Complete Running Application | Deliverable II should be submitted in the second last week of 8 th /4 th Semester of BS/ BS (Post ADP)/M.Sc. |

TABLE OF CONTENTS

(Required Chapters of Project Documentation)

TITLE AND DESCRIPTION:

- Inner Title Page
- Statement of Submission
- Declaration
- Plagiarism undertaken
- Acknowledgment
- Abstract

CHAPTER NO. 1: Gathering & Analyzing Information

- Introduction
- Problem Statement
- Goal & Objectives
- Research Questions
- Methodology
 - Available Methodologies
 - Chosen Methodology
 - Reasons for Chosen Methodology
- Definitions, Acronyms, and Abbreviations

CHAPTER NO. 2: Software Requirement Specification

- Stakeholders Characteristics
- Domain Requirements
- Functional Requirements
- Non-Functional Requirements

CHAPTER 3: Analysis [Use Case Description and Use Case Model]**CHAPTER 4: Design [with Description of each diagram]**

- Architecture Diagram
- ERD
- Data Flow diagram (Levels 0 and 1)
- Class Diagram
- Sequence Diagram

CHAPTER 5: Graphical User Interfaces

- (Mock-ups of the working software application)

CHAPTER 6: Testing

- Introduction
- Test Scenario (Sample Attached)
- Test Plan
- Definition of Test Cases
 - Test Cases Specifications
 - Test Cases Results for:
 - Black Box Test Cases
 - White Box Test Cases.

CHAPTER 7: Conclusion and Future Work**References (APA 6 Edition Style)**