Course Description

Introduction to the Undergraduate Information Technology Program

by

Author

Mohammad Hossein **Mohammadi**

Mohammadimh76.github.io

M.H.Mohammadimir2017@gmail.com

Github Repository

University: Islamic Azad University Najafabad Branch
Faculty: Computer Engineering
Field of Study: Information Technology (IT)

Licence

Curriculum and course descriptions for each lesson are provided in English, translated according to the Ministry of Science program.

Hello! Some universities like TUM and ETH require you to provide syllabi for the courses you have taken during your undergraduate or graduate studies when applying. This document is a translation of the Ministry of Science file.
Please refer to your own file for each major.
If you don't see a course in this list, please submit a request on **Github** to have it added, so that your classmates' and friends' time won't be wasted unnecessarily.
This file was initially prepared for the Computer Engineering major, Information Technology branch, at Islamic Azad University, Najafabad Branch (**IAUN**).

The main idea for creating this document came from my friend **Hossein**'s github repository. Some of the courses are also taken from Hossein's github repository. If you are in the Computer Engineering major with a software specialization, you can refer to **Hossein's repository**.

Description

Introduction to the Undergraduate Information Technology Program at Islamic Azad University, Najaf Abad Branch

Welcome to the detailed guide for the Undergraduate Information Technology Program offered exclusively at the Islamic Azad University, Najaf Abad branch. This document serves as a comprehensive resource for students, faculty, and academic advisors, offering a deep dive into the curriculum, courses, and educational objectives of the Information Technology undergraduate program.

Structured to provide a robust foundation in both theoretical concepts and practical applications, the curriculum is divided into four key categories: Specialized Courses, Main Courses, Basic Courses, and General Courses. This segmentation ensures a holistic educational approach, balancing deep technical expertise with essential foundational knowledge.

- **Specialized Courses** offer in-depth knowledge on cutting-edge topics and technologies within the Information Technology sphere, allowing students to specialize in areas of their interest.
- **Main Courses** cover the core principles and practices essential to Information Technology, laying the groundwork for both specialized study and broad understanding.
- **Basic Courses** aim to solidify the students' grasp on fundamental scientific and mathematical concepts, critical for success in technology-related endeavors.
- **General Courses** are designed to enhance students' broader skills such as critical thinking, ethics, and communication, which are indispensable in the professional world.

Each course category is complemented by detailed descriptions, which include the course objectives, the topics that will be covered, and the recommended references, providing a clear roadmap for students. This meticulous documentation ensures that every student has the information necessary to navigate their academic and professional journey successfully.
This guide articulates the essence of the Information Technology undergraduate program at the Islamic Azad University, Najaf Abad branch, reflecting our commitment to offering a curriculum that is both comprehensive and adaptable to the evolving needs of the technology sector.

Contents

1

Specialized Courses

1.1. System Analysis and Design

The purpose of this course is to acquaint students with the concepts of analysis and design of software systems. In this course, students will be introduced to different information systems, software product life cycle and development, different software creation methods, analysis, and structured design, and project management concepts.

Name of Course	System Analysis and Design
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Advanced Programming
Reference books	[1] L. D. Bentley and J. L. Whitten, Systems Analysis and Design for the Global Enterprise. 7th Edition, McGraw-Hill, 2007. [2] C. Larman, Applying UML and Patterns: An Introduction to Ohject-Oriented Analysis and Design and Iterative Development. Addison Wesley, 2004 [3] Roger S. Pressman, Software Engineering: a Practitioner's Approuch. Mc-Graw Hill Inc.,7th Edition, 2011.

At the end of the module students are able to:

- Application of software development in software production
- Selection and application of software development tools
- Documenting software products using UML language
- Analysis and design of systems by the object-oriented method
- Importance of software testing and test design

Course Objectives:

1. Introduction to software engineering and its challenges
2. Software development process models and their differences

3. Software analysis and design methods
4. Requirements engineering and system analysis
5. System design and software architecture
6. Build software
7. Introduction to software testing
8. Introductory introduction to project management and planning

1.2. Principles of Database Design

Students can apply the essential concepts of relational database systems and can use and evaluate them systematically and in a qualified manner. The students have the expertise to systematically use a database system, starting from the conceptual design to the implementation design to the physical design. They can formulate even complex queries in SQL and have a basic understanding of logical and physical optimization based on relational algebra. Furthermore, they know how to safeguard a database application concerning recovery, concurrency control, and authorization. The course also covers database design methodology. main course topics are data models, the architecture of database systems, data definition and manipulation languages, database design methods, and the theory of data dependencies and relational normalization.

Name of Course	Principles of Database Design
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Data Structures
Reference books	[1] R. Ramakrishnan and J. Gehrke, Database Management Systems, 3rd Edition. McGraw-Hill Inc., 2003. [2] H. Garcia-Molina, et al., Database Systems: The Complete Book 2nd Edition, Pearson Prentice Hall, 2009. [3] J.D. Ullman and J. Widom, A First Course in Database Systems. 3rd Edition, Pearson Prentice Hall Inc., 2008.

At the end of the module, students are able to:

- A general understanding of database systems and their architecture
- Drawing a chart of existence and relationship
- Convert existence and relationship diagrams to corresponding relationships
- Recognition of functional dependencies and normalization of relationships
- SQL QA
- How relationships and indexes are stored in a database system

Course Objectives:

1. Basic concepts
    - Relational model
    - Relational algebra and relational arithmetic
    - SQL
    - Normal descriptions
2. Storage and indexing

- Data storage
- Indexing with tree structure
- Indexing based on scrambling

3. Evaluate questions and answers

    - External sorting
    - Evaluate relationship operators
    - Optimize questions and answers

4. Transaction management

1.3. Fundamentals of Information Technology

Familiarity of students with the principles, definitions, concepts, applications, organizational and social effects, management concepts, foundations, and architecture of information technology.

The broad scope of the application of Rabaneh forms the framework of information technology discussions, and the preconceptions of this technology consider the scope, social, economic, and cultural effects to be influential. In this first lesson, IT students are introduced to the principles, definitions, concepts, applications, organizational and social effects of the management concepts of this technology, its foundations, and architecture. Because computer and information technology engineers are the creators and promoters of new solutions in this field, they should be aware of the latest concepts, achievements, and areas of application of this technology in the world and in Iran. The superficial breadth of the concepts of this lesson, due to the depth of each topic in the following lessons, forms its structural form.

Name of Course	Fundamentals of Information Technology
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Does not have
Reference books	[1] Linda Volonino, Efrain Turban, Information Technology for Management Improving Performance in the digital Economy. & Edition, Wiley, 2011. [2] Efraim Turban, Dorothy Leidner, Ephraim Mclean and James Wetherbe; information Technology. for Management Transforming Organizations in the Digital Economy, 5 Edition, John Wiley & Sons Inc, 2006 [3] E.Turban, R.K.Rainer, R.E.Potter, Introduction to Information Technology, Wiley, 3rd Edition, 2005. [4] Urs Birchler and Monika Butler, Information Economics, Routledge, 2007. [5] E.W.Martin, C.V.Brown, Managing Information Technology. Prentice Hall, 5th Edilion, 2004. [6] K.D.Willett, Information Assurance Architecture. CRC, 2008. [7] Thomas.H.Davenport and Laurence Prusak, Information Ecology: Mastering the Information and knowledge Environment. Oxford University Press, 1997.

Course Objectives:

1. Introduction
2. Background, definition, principles, framework and preconceptions
3. Data, information, and knowledge
4. Network Computing and Fa Management in a Fa-Based Organization in a Digital Economy
5. Ability to absorb Fa, electronic readiness, digital rankings, and digital gap criteria

6. E-commerce, business intelligence, and data warehousing
7. Wireless and mobile computing, pervasive, live and present and currency
8. Business, corporate, local and international systems, their features and integration
9. Support systems for management, supply chains, organizational resource planning, and linking to clients
10. Types of Internet structures, foundations, and Fa architecture
11. Combined applications of today's value-added Fa
12. Impacts, etiquette and security
13. Information society and e-government, e-services and foundations
14. National and International Information Technology

1.4. Principles of Information Technology Strategic Management and Planning

Familiarity of information skills with theoretical and practical topics of Fa-strategic studies in the fields of information technology management and planning in the organization and gaining knowledge of selecting the appropriate type of Wi-Fi strategic study for each organization according to its absorption capacity, using appropriate methods and producing solutions Transition with organizational methods using modifiable engineering patterns is one of the main objectives of this course. The second goal of the course is to understand the need to draw architectural plans and update them to move from the current situation to the desired conditions and integrate solution systems that are pioneers of national projects such as e-government. Improving the level of students' attitudes and gaining the ability to extract systemic solutions for the realization of organizational solutions will also become a necessary skill with practical exercises.

Name of Course	Principles of Information Technology Strategic Management and Planning
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Does not have
Reference books	[1] Inge Hanschke, Strategic IT Management, Springer, 2010. [2] Danny Greefhorst, Erik Proper, Architecture Principles, Springer, 2011. [3]Martin Op't Land, Enterprise Architecture Creating Value by informed Governance, Springer, 2009. [4] Mario Godincz, The Art of Enterprise Information Architecture, IBM Press, 2010.

Course Objectives:

1. Introduction
   - Fundamentals and principles of strategic management and planning
   - Define the foundation, architecture, management, and planning of the Fa
   - Concepts of organization, enterprise, mission, vision, plan, plan and project
2. Introducing the background of early organizational orthodox Romans
   - Information Engineering
   - Strategic business planning
   - Information resource management
3. Basic patterns of strategic planning and management
   - Performance measurement reference model
   - Application of models PEST, SWOT, SPACE & QSPM
   - Zakman Strategic Framework and its applications

- Strategic alignment models
- Mans's integrated architecture and management model and Nolan organizational life and maturity model

4. Detailed study of family-oriented strategic planning patterns
   - Strategic business planning
   - Strategic planning of information systems
   - Planning and managing the organization's information resources
   - Strategic planning of the organization's information technology
   - Organizational architecture planning
5. Organizational FI strategic planning bicycle
   - Knowledge of the organization
   - Alignment analysis
   - Analysis of the current situation
   - Optimal condition design
   - Develop a transition program
6. Organizational information architecture planning process model
   - Develop a policy
   - Draw maps of existing conditions
   - Drawing optimal status maps
   - Distance measurement of the current and favorable situation
   - Compile a change statement and schedule a transition
   - Develop operational plans
7. An example of organizational architecture methods and reference models
   - FEAF, NIST
8. Organizational architecture in Iran and the introduction of several large projects
   - Iranian organizational architecture reference models
   - Iran's position in the rankings of organizational architecture in the world
   - Several large national projects
9. Status and future of applications of Fa's strategic studies
   - Fa rule and required integration of e-government project
   - Management of organizational architecture and service-oriented organizational architecture
   - Applications of some standard Fa service management templates
   - Quality assurance of information and services and organizational knowledge architecture
   - Koenigsberg model
   - Landscape view of architectural studies
   - Minimalist view of architecture
   - A supermodel look at enterprise architecture
10. Documenting and updating strategic study plans and computer tools
    - Types and how to draw maps of each layer of architecture
    - A variety of tools for producing, maintaining and updating maps
    - How to select the minimum number of maps per layer
    - Fa-Strategic Studies Document Configuration Management

1.5. Information Technology Projects Management

In this course, students get acquainted with the principles and foundations of project management and control and how to use the techniques of this field in information technology projects.

Name of Course	Information Technology Projects Management
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Does not have
Reference books	[1] David L. Olson, Introduction to Information Systems Project Management with CD ROM Mandatory Package. McGraw-Hill, 2001. [2] Graham McLeod and Derek Smith, Managing Information Technology Project, Course Technology, 1996. [3] Chris Kemerer, Software Project Management: Readings and Cases, McGraw-Hill, 1997.

Course Objectives:

1. Challenges of Fa Management
2. Strategies for Fa projects
3. Start a project and define requirements
4. Forming the Fa Project Team
5. Project planning
6. Perform estimates
7. Execution and control of the project
8. Manage hardware and communication projects
9. Software project management
10. Integrated systems management

1.6. Enterprise Applications Integration

Familiarity with techniques, tools, and skills in integrating enterprise application systems, providing solutions for linking antique and new applications in network application environments. Data and process, the concept of enterprise service gateway, its structure, and application, the types of integration patterns in service architectures. Incident - drive and grid, familiarity with safety considerations in integration.

Name of Course	Enterprise Applications Integration
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	System Analysis and Design - Computer Networks
Reference books	[1] G.Schmutz, D.Liebhart,P.Welkenbach , Service-Oriented Architecture: An Integration Blueprint. Birmingham: Packt Publishing Ltd., 2010. [2] G. Hohpe and B. Woolf, Enterprise Integration Patterns. Addison-Wesley Inc., 2004. [3] David S. Linthicum, Enterprise Application Integration. Addison-Wesley Inc., 1999. [4] Amjad Umar, Enterprise and Inter-Enterprise Application Integration. NGE Solution, Inc., 2003.

Course Objectives:

1. Basic concepts and necessities of integration, technologies, and architectures
2. Types and meanings and levels of integration
3. Types of layered architectures, their applications, advantages, and facilities
4. Types of link middleware and their new types
5. A variety of integrated architectures
6. Integration and ESB
7. Patterns of data integration, service-oriented, event-driven and grid
8. Basic technologies: OSGI, JCA, JBI, SCA, SDO
9. Process modeling
10. Architectural integrity
11. Implementation scenarios of service, data, and event and network integration
12. Gates and environments connection and integration
13. Styles and methods of integration
14. Study of organizational architecture, productive and supportive of systems integrity
15. Legal considerations in connection and security considerations in integration

1.7. Fundamentals of Secure Computing

During this course, students, after understanding the need to establish security and the various threats and attacks in it, with issues related to how to establish and implement security using authentication methods, encryption access control, how to provide physical and environmental security, and methodology, familiarize themselves with security assessment standards.

Name of Course	Fundamentals of Secure Computing
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Computer Networks
Prerequisites	Operating Systems
Reference books	[1] Ross J. Anderson, Security Engineering: A Guide to Building Dependable Distributed Systems. 2nd Edition, Wiley, 2008. [2] Matt Bishop, Introduction to Computer Security. Addison-Wesley, 2004.

Course Objectives:

1. Basic concepts and definitions
   - Basic concepts in security
   - Types of threats and attacks
   - Vulnerabilities
   - Security mechanisms
   - Principles of Security Engineering
2. Authentication
   - Authentication methods
   - Identity Management Systems
   - Audit Challenges vs. Privacy
3. Security policy and access control models
   - DAS
   - ACL & Capability-based
   - Hidden channels, information flow control and forced models (MAC)
   - RBAC
4. Cryptography and its applications
   - Symmetric cryptography
   - Asymmetric encryption and digital signature
   - Tangler functions
   - Applications of cryptography in maintaining confidentiality, authentication and non-denial

5. Symmetric cryptography
    - Fistula code structure
    - Encryption algorithms based on fistula structure
    - Symmetric cryptographic standards
    - Working modes in symmetric code
6. Asymmetric cryptography
    - Introduction and necessity
    - Cryptographic and asymmetric standards
7. Digital key signature and public key infrastructure
    - Simple digit signature algorithms
    - RSA
    - Certificate management
8. Data security approaches
    - Transfer data security approaches
    - Processing data security approaches
    - Stored data security approaches
9. Physical and environmental security
    - Threats and physical attacks
    - Physical and electronic barriers and locks
    - Physical protection of communications
    - Sensors and alarm systems
10. Security and reassurance assessment
    - Security assessment and testing methodologies
    - Assessment and reassurance standards
11. Non-technical dimensions of secure computing
    - Ethical Dimensions in Secure Computing
    - Judicial Dimensions of Secure Computing
    - Social Dimensions of Secure Computing
12. Content responsibility in secure computing

1.8. Engineering Economy

Familiarity of engineering students with the techniques of economic evaluation of industrial and service plans and projects so that they can apply these techniques and methods in their field of specialization.

Name of Course	Engineering Economy
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Does not have
Reference books	[1] William G. Sullivan, Elin M. Wicks, C. Patric Koclling, Engineering Economy. 15th edition, Prentice Hall, 2011. [2] L.T. A. Blank, A. J. Tarquin, Engineering Economy. 6h edition, McGraw-Hill, New York, 2005.

Course Objectives:

1. Importance and necessity of engineering economics
2. Drawing liquidity flow and how to estimate the parameters affecting liquidity flow
3. Mathematical relations to determine the explanation of economic parameters
4. Using the table of factors in economic calculations
5. Liquidity flow with geometric slope and specific liquidity flow modes
6. Nominal and practical interest and their effect on the content
7. Economic evaluation techniques(IRR, B/C, NEUA, NPV)
8. Return on investment and projects with multiple rates of return
9. Economic decisions based on the decision horizon
10. Methods of calculating depreciation, calculating taxes and calculating tax savings
11. Conduct economic studies taking into account taxes and depreciation
12. Alternative studies in economic evaluations
13. Sensitivity analysis in economic studies
14. Investigating the effect of inflation in economic studies
15. Examples and applications of economic studies

1.9. Electronic Commerce

In this course, students are introduced to the main concepts of business and IT applications in e-commerce.

Name of Course	Electronic Commerce
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Computer Networks - Engineering Economy
Reference books	[1] R. Kalakota, A.B. Whinston, and T. Stone, Frontiers of Electronic Commerce, Addison-Wesley, 1996. [2] S. Solomon, Marketing. Student Edition, Prentice-Hall, 1996. [3] P. Koiler, and G. Amstrong, Principles of Marketing, & Edition, Prentice-Hall, 1998. [4] D. Kosiur, Understanding Electronic Commerce, Microsoft Press, 1997. [5] S.L. Huff, CASES in Electronic Comnrerce, 2 Edition. McGraw-Hill, 2002. [6] R. Reddy, and S. Reddy, Supply Chains to Virtual Integration, McGraw-Hill. 2002. [7] W. Raisch, The Marketplace: Strategies for Success in B2B eCommerce, McGraw-Hill, 2001.

Course Objectives:

1. Introduction to E-Commerce
2. The emergence of knowledge-based business
3. Value in a network economy
4. Factory and virtual organization
5. Product development in the digital economy
6. Marketing in the digital economy
7. Product management and trading services
8. Strategic planning and trading process
9. Security and e-commerce
10. E-commerce infrastructure
11. E-Commerce Software
12. Search strategies
13. Applications of software agents in the business

2

Main Courses

2.1. Fundamentals of Computer Programming

The purpose of this course is to acquaint students with the basics of computer and programming. After learning the basic structure of the machine and the basics of computing in hardware, students become familiar with organizing the components of a modern computer. Another emphasis of this course on programming in java is writing engineering code (tweed modular, clean code, commenting, spacing, and the ability to implement pseudocode).

Name of Course	Fundamentals of Computer Programming
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Does not have
Reference books	[1] P. Deitel and 11. Deitel, Java: How to Program. 6th Edition, Prentice Hall, 2009.

At the end of the module students are able to:

- Familiarize students with the basics of computer and programming
- Gain the ability to program in Java
- Gain the ability to write and implement pseudocode

Course Objectives:

1. Basic concepts
2. Calculations on the computer
3. Programming Basics
4. Output / input formatting

5. Familiarity with algorithms, flowcharts and pseudocodes
6. Orders
7. Functions
8. Familiarity with program testing and troubleshooting
9. Arrays
10. Characters and strings
11. Structures
12. Inputs and outputs with files

2.2. Advanced Computer Programming

The purpose of this lesson is to explore different ways to produce a quality program. In this regard, after covering the top-down design method for problem-solving, students are introduced to the concepts of object-oriented programming as a tool for complexity management in medium and large-sized applications. Throughout the lesson, you will focus on the proper functioning of the test and debugging program, which is achieved through methods such as unit tests, assertions, and pre-and multiple conditions. The emphasis of the lesson will be on methods rather than the advantages of a particular programming language. This lesson can be presented in any common object-oriented programming language, such as Java with C++.

Name of Course	Advanced Computer Programming
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Fundamentals of Computer Programming
Reference books	[1] H.M. Deitet and P.J. Deitel, C++ How to Program. Sth ed., Prentice-Hall Inc., 2011. [2] P. Deitel and H. Dcitel, Java: How to Program. 9th Edition, Prentice Hall Inc., 2011.

At the end of the module students are able to:

- Solve problems with the top-down design method
- Manages the complexity of the programming problem by defining appropriate classes
- Appropriate abstraction methods such as inheritance and polymorphism are used
- They use the essential features of programming language libraries
- Ability to use the necessary methods to test and debug the program

Course Objectives:

1. An overview of the basics of programming
2. Top-down design
3. Basic concepts of objectives
4. Basic structures of object-oriented programming
5. Inheritance and polymorphism
6. Memory Management
7. General programming
8. Handle errors

9. Input/output libraries
10. Data libraries of standard structures
11. Create a graphical user interface
12. Text processing and strings
13. Introduction to Simultaneous Programming
14. Test and debug the program

2.3. Discrete Mathematics

The purpose of this course is to acquaint students with the concepts, structures, and techniques of discrete mathematics that I Widely used in computer science and engineering. Develop basic skills, including understanding and building accurate mathematical proofs, creative thinking in problem-solving. Familiarity with the initial results in number theory, logical. Combinations and graph theory and the mathematical prerequisites required for many of the other courses offered in various computer engineering majors are among the objectives of this course.

Name of Course	Discrete Mathematics
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	General Mathematics I (Calculus I) - Fundamentals of Computer Programming
Prerequisites	Does no have
Reference books	[1] K. H. Rosen, Discrete Mathematics and its Applications. 6th Edition, McGraw-Hill Inc., 2007.

At the end of the module, students are able to:

- Mathematical reasoning and methods of using arguments to solve problems
- Combination methods and counting methods
- Graphs and trees

Course Objectives:

1. Fundamentals of Mathematical Logic
2. Theory of functions and sets
3. Number Theory
4. Induction
5. Counting
6. Reciprocal relations
7. Relationships
8. Partial order
9. Algebra Bull
10. Graphs
11. Trees
12. understand the elementary vocabulary of discrete mathematics and use logic algebraic and algorithmic calculations
13. solve combinatorics problems
14. model and solve problems using graph theory
15. do a quantitative analysis of the efficiency of algorithms

2.4. Signal and Systems

Students are familiar with describing signals and analyzing linear and time-invariant systems in the domains of time and frequency.

Name of Course	Signal and Systems
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does no have
Prerequisites	Engineering Mathematics
Reference books	[1] A.V. Oppenheim, A. S. Willsky, and S.II. Nawab, Signals and Systems. 2nd ed. Prentice-Hall, 1996. [2] R. E. Zicmer, W.11. Tránter, and D. R. Fannin, Signals and Systems , Continuous and Discrete. 4th., Prentice-Hall, 1998. [3] S. Haykin and B. Van Veen, Signals and Systems. 2nd ed. Wiley, 2003.

Course Objectives:

1. Introduction: Mathematical concepts and tools of signal processing and systems analysis Definitions, properties and general classifications
2. Continuous and discrete-time signals Mathematical representation, stroke, and step, alternating, mixed exponential, power, and energy
3. Linear and immutable systems with impact response time, convolution concept, impact response properties, description by differential equations
4. Fourier series intermittent signals: The importance of display in calculating the response of LTI systems Continuous-time signals, discrete-time signals, Fourier series convergence, Fourier series display properties
5. Fourier transform continuous definition time, convergence, properties, analysis of systems described by differential equations with constant coefficients
6. Discrete Fourier transform time: definition, convergence, properties, analysis of systems described by differential equations with constant coefficients
7. Sampling: Theorem of ideal and non-ideal forms, Hatifi interference phenomenon, reconstruction
8. Laplace transform definition, convergence, properties. LTI system conversion and analysis function (continuous-time)
9. Z Conversion: Definition, Convergence, Properties, Conversion Function and Analysis of LS Systems

2.5. Data Structures

Theoretical and practical exercises deepen the knowledge imparted to be able to design, analyze and implement data structures and algorithms yourself or to be able to use them sensibly in your programs. Students learn how the choice of a data structure affects the performance and usability of applications, e.g., in web browsing and searching, computer databases, data analysis, text processing, etc. Specific topics include lists, stacks, queues, sets, maps, priority queues, trees, and graphs, together with general algorithmic techniques, such as sorting, searching, and other transformations on data. Students who complete the course can use these tools to design and develop efficient programs for a wide variety of applications.

Name of Course	Data Structures
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	General Mathematics I (Calculus I) - Fundamentals of Computer Programming
Prerequisites	Does no have
Reference books	[1] T. Cormen, C. Leiscrson, and R. Rivest. Introduction to Algorithms. McGraw-Hill Inc., 2001.

At the end of the module, students are able to:

- Use existing wind buildings appropriately and appropriately
- Design different data as needed
- Design and implement various algorithms for accessing and processing data
- Analyze the data structures and algorithms in question from the dimensions of time and irony

Course Objectives:

1. Algorithm analysis methods: Growth function. Counting steps. Recursive relations and methods of solving them, conjecture and induction, repetition with placement and use of the main theorem), breaking analysis
2. Types of lists (one-way, two-way, general, hash and stack lists): Different actions on lists, use of actual and Andy pointers, implementation of various problems with lists (working with mathematical expressions, garbage collection, Integrated sorting)
3. Trees: basic definitions, phrase tree, different tree implementations, tree induction, navigation and structural induction trees of a binary tree, different actions on phrase tree, converting different versions of phrases together, tree, binary tree search

4. Interlocking method in interlocking global chains, open
5. Sorting and Statistical Ranking Bottom Line, Decision Tree, Linear Sorting (Counting, Enamel, and Bucket), Quick Sorting, Pyramid Sorting, Statistical Sorting, External Sorting
6. Advanced data structure: separate sets, red-black trees, statistical order tree, interval tree, B tree

2.6. Technical English

The purpose of this course is to develop the ability to read fluently and understand English texts in computer engineering and information technology and, to some extent, to understand scientific lectures in this field. This course tries to improve the student's skills by relying on continuous exercises in reading and writing and listening to scientific lectures in English.

Name of Course	Specialized English
Number of Credits	2
Number of Hours	32
Cross section	Bachelor's Degree
Needs	Does no have
Prerequisites	General Language
Reference books	[1] Selected short articles on Computer Engineering and Information Technology (from different authors) [2] TED group scientific lectures [3] EE Times [4] IEEE Spectrum Magazine

At the end of the module, students are able to:

- Ability to read a technical text of computer engineering with medium complexity fluently
- Relatively good understanding of the meaning of the text at the same time as reading the text
- Relative understanding of technical lectures in the field of computer engineering
- Ability to write simple technical texts with appropriate speed

Course Objectives:

1. Browser security
2. Cloud Computing
3. Network Security
4. What is FPGA?
5. What is Linux?
6. What is service-oriented architecture?
7. Data communication channel
8. Bit Torrent
9. How to optimize search engine
10. Operating Systems
11. Programming Languages

2.7. Electric Circuits

Objectives of the course are students' familiarity with the principles and theorems, and methods of analyzing electrical circuits and gaining the ability to analyze the dynamic behavior of the circuit as an energy system.

Name of Course	Electric Circuits
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does no have
Prerequisites	Differential Equations
Reference books	[1] L. O. Chua, C. A. Desoer, and E. S. Kuh, Linear and Nonlinear Circuits. Mc-Graw Hill, 1987. [2] J. W. Nilson, Electric Circuits. 4" Edition, Addison Wesicy, 1995. [3] RJ. Smith and R. C. Dorf, Circuits,Devices, and Systems, sih Edition, John Wiley, 1992.

Course Objectives:

1. Rules and definitions
   - Concepts of compact and wide circuits
   - Voltage and current laws
   - Elements of an ideal and real valve (resistor, capacitor, inductor and sources of voltage and constant current)
   - Two-valve elements (dependent voltage and current sources, transistor model and operational amplifier)
   - Concepts of power and energy
   - Concepts of active or inactive elements
   - Waveforms (stairs, pals, impact and sine)
2. General circuit analysis
   - Concepts of linearity and immutability with time
   - The concepts of zero state response and zero input response
   - Concepts of transient state response and permanent state response
   - Response concepts in the time domain and frequency domain
   - Methods of node and mesh analysis
3. Analysis of circuits in the time domain
   - Simple circuits
   - Ordinary circuits
   - A concept of step response and impact response
   - Second order circuits
   - Far higher circuits
4. Convolution theorem and users

- Convolution integral
- Zero mode response of linear circuits

5. Circuit analysis in the frequency domain
    - Laplace transform
    - How to use Laplace transforms in the analysis of electrical circuits
    - February series
    - Sinusoidal permanent state response
    - The concept of network function and its relationship with impact response
    - The concept of frequency response
6. Network theorems and their applications
    - The case of collecting works
    - Norton Theorems
    - Maximum power transfer theorem
7. Familiarity with spice simulator software and its user in electrical circuit analysis

2.8. Operating Systems Lab

This course aims to create a good understanding and learn the techniques of designing and implementing operating systems based on a .open source operating system. Students in this course will be introduced to the basic concepts of operating system design such as file management, process management, programming at the cartel level, process synchronization in a Mann Bar operating system, and the installation and management of an open-source operating system. (Laboratory and practices of Operating Systems course).

Name of Course	Operating Systems Lab
Number of Credits	1
Number of Hours	16
Cross section	Bachelor's Degree
Needs	Operating Systems
Prerequisites	Does no have
Reference books	[1] M. K. Dalheimer, T. Dawson, L. Kaufman, M. Welsh, Running Linux. O'Reilly, 2002 [2] K. Wall, M. Watson, and M. Whitis, Line Programming Unleashed. Sams PublishersInc., 1999.

At the end of the module students are able to:

- Install and manage an open-source operating system
- How to implement the basic concepts of operating system design in an open-source operating system
- Programming at the kernel level

Course Objectives:

1. Introduction: Overview of Linux including History, Versions, POSIX. Graphic user interface
2. Linux installation: disk partitioning, boot loading, using the application menu and system
3. Linux file system
4. Standard and advanced shell
5. Process and clause management
6. CPU scheduling in Linux
7. Sync and deadlock
8. Kernel programming and system services

2.9. Logic Circuits

Covers the design and application of digital logic circuits, including combination and sequential logic circuits. After taking this course, students will be able to recognize and use the following concepts, ideas, and tools: 1) Logic level models, including Boolean algebra, finite state machines, arithmetic circuits, and hardware description languages, 2) Logic gates, memory, including CMOS gates, flip-flops, arrays, and programmable logic, 3) Design tools, both manual and computerized, for design, optimization, and test of logic circuits, and 4) Design criteria, including area, speed, power consumption, and testability.

Name of Course	Logic Circuits
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Discrete Mathematics
Prerequisites	Does no have
Reference books	[1] S. Brown and Z. Vrancsic, Fundamentals of Digital Logic with Verilog Design. 3 Edition, McGraw-Ilill, 2009. [2] C.H. Roth and L.L. Kinncy, Fundamentals of Logic Design. 5° Edition, 2005. [3] J. Wakerly. Digital Design, Principles and Practices. 4" Edition. 2005. [4] Victor P. Nelson, II. Troy Nagle, Bill D. Carroll, and David Irwin, Digital Logic Circuit Analysis and Design. Prentice Hall, 1995.

At the end of the module, students are able to:

- Basic concepts of digital systems and circuits
- Design of digital systems
- Digital systems analysis
- Digital systems modeling

Course Objectives:

1. Introduction and basic concepts
    - History of digital systems
    - Application of digital systems in today's world
    - Basics of digital systems and their differences with analog systems
    - overview of digital circuits based on MOS transistors
2. Numerical systems
    - Theory of numbers and their representation
    - Computations in digital systems
    - Concepts of Carry and Overflow
    - LCD display systems

3. Algebra Bull
    - Principles of Bull Algebra
    - Functions, operators and logic gates
    - Bull algebra relations
    - Display Max term, Min term, Canonical and standard forms display logical functions
    - Learning hardware description language (VHDL or Verilog) at the structural level
4. Analysis and design of hybrid logic systems
    - Methods for simplifying combined circuits with Boolean algebra
    - Optimization of hybrid circuits with Carnot table and Queen-McCluskey algorithm and the concept of trivia
    - Glitch concept, Hazard Race concept
    - types of implementation of two-story circuits
    - The concept of delay
    - Coding, decoding, dividing, pod-sharing, seven-part circuits and their applications, especially as a basic block in logic circuit design
    - Design with universal kits (Universal)
    - Use of three-state buffers to create high impedance, open collector gate wires, belong to the wire, use of resistors as pull-up and pull-down
    - Programmable circuits
    - Introduction of standard hybrid chips
    - Diffusive collector circuits, comparators, collectors with high impedance conveyor number prediction
5. Analysis and design of sequential logic systems
    - Introducing memory elements, flashes and flip-flops
    - Release delay of memory elements, concept of start-up time and storage time, synchronous and asynchronous inputs
    - Sequential circuit analysis, excitation table, state diagram. Mode table
    - FSN design steps for fcaly and Moorc models and their differences
    - Design of sequential circuits with different types of flip-flops
    - Universal counters, registers and shifters and candies
    - Introduction of standard sequential chips
6. Basics of asynchronous circuit design

2.10. Logic Circuits and Computer Architecture Laboratory

Practical acquaintance with the concepts presented in logic circuit and computer architecture courses such as digital circuit design, how to analyze and debug them. Design the logical and arithmetic parts of a processor design memory units, input / output, and data path and control in a typical processor, use hard descriptive languages Software such as Verilog and VHDL to describe the simulation, synthesis and finally implementation in the section on FPGA boards at the gate and RTL abstraction level.

Name of Course	Logic Circuits and Computer Architecture Laboratory
Number of Credits	1
Number of Hours	16
Cross section	Bachelor's Degree
Needs	Computer Architecture
Prerequisites	Logic Circuits
Reference books	[1] S. Brown and Z. Vranesic, Fundamentals of Digital Logic with Verilog Design. McGraw-Hill, 2003. [2] B. Parhami, Computer Arithmetic - Algorithms and Hardware Designs, Oxford Univ.Press, 2000. [3] D.A. Patterson and J. L. llennessy, Computer Organization and Design: The Hardware, Software Interfacc, 4 Edition, Morgan Kaufman Publisher Inc., 2010. [4] J. L. Hennessy and D.A. Patterson, Computer Architecture, A Quantitative Approach,Prentice-Hall, 4th cdition. [5] D.M, Harris, Digital Design and Computer Architecture, 24 Edition, Morgan Kaufman Publisher Inc., 2012.

At the end of the module, students are able to:

- Design of digital systems using SSL chips and their debugging and debugging
- Familiarity with how to design different parts of a typical processor and its peripheral units

Course Objectives:

1. Work with schematic design software and simulation of sequential and hybrid digital circuits in the environment FPGA
2. Work with simulation software based on hardware description language and test the synthesis results of various structures in these languages
3. Design of hybrid circuits with SSI chips and analysis and implementation of Verilog equivalent with their VIDI
4. Familiarity with the performance of counters and registers and analysis and implementation of Verilog equivalent with their VHDL
5. Design of RAM, ROM and dual port memory

6. Design of summers and multipliers and analysis of their size and speed Design of sequential circuits and analysis of their performance and specifications and analysis and implementation of their Verilog or VHDL equivalent
7. Implementation of sending in synchronous texture in one-way and two-way passages
8. Implement the crossing arbitration mechanism
9. Implement the pipeline and review its acceleration
10. Implementing a binary divider in complementary logic 2
11. Design and implementation of a desktop computer
12. Memory hierarchy design
13. Measurement of CPI parameters IPC and MIPS in a typical processor

2.11. Computer Networks

This course examines the principles of design, implementation, and performance of computer networks. In this course, students will be introduced to the architecture and services of computer networks and the Layda model. This course focuses on Internet networks and the TCP / IP model to examine application layer protocols. It deals with the transport layer, the network layer and the data link layer.

Name of Course	Computer Networks
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does no have
Prerequisites	Operating Systems
Reference books	[1] James F. Kurose and Keith W. Ross, Computer Networking: 4 Top-DowwApproach. 5th edition, Addison-Wesley Inc., 2009. [2] Alberto Leon-Garcia and Indra Widjaja, Communication Networks. 20 edition.McGraw-Hill Inc., 2003.

Course Objectives:

1. Overview of computer network services (examples of network services, computer network definition, service definition and service quality, protocol definition)
2. Internet network and its components (definition of Internet network components (network edge and core), Client-Server model, access networks and physical media, packet switching and circuit switching service quality parameters in packet switching networks)
3. Layer architecture of computer networks (reference model 051. Unified view of layers, protocols and services of connection-oriented and offline service models, component manufacturing and reconstruction, multiplexing and multiplexing, TCP / IP model)
4. Application layer (principles of web network applications and HTTP protocol, file transfer, FTP protocol, e-mail, SMTP protocol, Internet directory service and DNS protocol, peer-to-peer applications, socket programming with TCP, socket programming with UDP)
5. Layer transport (introduction of transport layer services, non-connection service of transport layer and UDP, principles of secure data transfer (ARQ error control protocols), connection-oriented transport layer service and TCP, congestion control principles, congestion control in TCP)
6. Network layer (introduction of network layer tasks (routing and forwarding), data logger networks and virtual circuit router architecture, traffic management in packet switching networks (packet level traffic management, queue management and packet scheduling), flow level traffic management (Congestion control), traffic management at the level of flow aggregation (traffic engineering),

Internet protocol (ICMP IPv6 IPv4 ARP protocols), DHCP and Mobile IP protocols, routing algorithms, distance vector and link status algorithms), routing protocols On the Internet (definition of AS and IGP and EGP protocols and RIP protocol OSPF protocol BGP protocol), multipart and all-part routing)

7. Data link layer and local area networks (introduction of data link layer and its services, general methods of error detection and correction, general methods of media access control methods, general local area networks)

2.12. Microprocessors and Assembly Language

This course provides students with basic microprocessor and microcontroller knowledge. In a way that will enable them to design systems based on microprocessors and microcontrollers. A significant portion of this lesson is based on the 808688 microprocessor. Learning this microprocessor, in the simplicity of the structure compared to today's much more advanced processors, conveys important basic information of microprocessor design concepts to students and prepares them to learn more complex structures. At the same time, this course will introduce a brief overview of the more advanced Pentium microprocessor, such as the stability of addressing methods and the concepts of real and protected state operation, to show a brief overview of its differences with the 86 80 (optional) microprocessor. The course is dedicated to the introduction of AVR family microcontrollers so that students with the facilities built into microcontrollers and concepts in embedded systems, including the combination of memory types of programmable parallel ports, types of hardware and software interrupts and their prioritization Familiarize the timer for use in counting timing events and generate analog comparator PWM signals, analog-to-digital converters, and some types of serial communication protocols. In order to get acquainted with assembly language, in this course, while getting acquainted with the main concepts of assembly language, 8084 microprocessor assembly instructions and AVR family are introduced and assemblers and integrated development environments (IDE) will be introduced to them.

Name of Course	Microprocessors and Assembly Language
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does no have
Prerequisites	Computer Architecture
Reference books	[1] John Uffenbeck, The 8086/8088 Family: Design, Programming, and Interfacing, Prentice Hall, 3rd Edition, 2001. [2] ATmegal6 microcontroller datashcct. [3] James L. Antonakos, The Pentium Microprocessor, Prentice-Hall, 1998. [4] Holzner Steven Advanced Assembly Language, Prentice-Hall, 1995. [5] Intel Corporation, Intel Pentium Developer's Manual, Volume 3, 1995. [6] NASM Development Team, NASM-Netwide Assembler User Manual, 2012. [7] Richard H. Barnett, Sarah Cox, Larry O'Cull, Embedded C Programming and the Atmel AVR, Delnmar Cengage Learning Publishing, 2011. [8] CodeVision AVR C compiler, User manual, 2003. [9] AVR Assembler, Atmel, 2004. [10] Atmel Studio, Atmel. [11] Winavr User Guide.

At the end of the module students are able to:

- Describe the architecture and organization of a microprocessor
- Write assembly programs with a proper structure, with sufficient and under-

standable explanations
- Understand the methods of connecting and scheduling different types of main memory and input-output to the microprocessor
- Understand the handling methods between the microprocessor and the input-output
- Use of microcontroller components (counter start time, analog to digital converter, various serial communication methods, etc.) in applications of embedded systems
- Use of integrated development environments (IDEs) as software tools for the development of microprocessor-based systems and microcontrollers in various applications

2.13. Microprocessors Laboratory

The purpose of the microprocessor course laboratory is to acquaint students with practical issues such as setting up a reset circuit, production | Clock signal, work with parallel ports and their programming, keyboard connection, 7-piece display and display | Character provider, communication with different types of memory, working with internal and external keys and prioritizing them, working with timer counter and event counting, timing and generating PWM1 signals, working with analog comparator and analog-to-digital converter to establish Communicating with analog signals, working with serial communications such as SP1 USART and TWVT, working with assemblers and compilers of microprocessors and microcontrollers, and finally simulating a microprocessor-based project or microcontroller in a simulation environment (such as Proteus and schematic design and PCB It is, for example, by Alnium software), and the order, construction and testing of the relevant board as the final project of this laboratory.

Name of Course	Microprocessor Laboratory
Number of Credits	1
Number of Hours	16
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Microprocessors and Assembly Language
Reference books	[1] John Uffenbeck, The 8086/8088 Family: Design, Programming, and Interfacing, Prentice Hall, 3rd Edition, 2001. [2] ATmega16 microcontroller datasheet. [3] CodeVision AVR C compiler, User manual. [4] AVR Assembler, Atmel. [5] Atmel Studio, Atmel. [6] WinAVR user manual.

At the end of the module, students are able to:

- Practical familiarity and ability to work with microprocessors
- Design hardware and software and build systems based on microprocessors and microcontrollers

Course Objectives:

1. Familiarity with a simulation software (such as Proteus for simulation and testing, familiarity with a printed circuit board design software (such as Altium software) for schematic and PCB design, and finally doing a simple electronic project. It can design and simulate a flashing circuit with two LEDs (unstable multivibrator circuit) and its schematic and PCB design, rolling order and assembly of parts on PCB and soldering and testing it. The final laboratory project will be used.

2. Introduction to microcontroller assemblers and compilers (such as Atmel Studio and CodeVision)
3. Construction of reset circuit programming cable, fuse bit programming, clock signal generation for microcontroller
4. Prepare reset interrupt program, set stack pointer, work with ports, polling a base of a software delay generation port, and work with guard timer
5. working with external interrupts and using power saving modes
6. working with value reading ports specified by a four-bit Djp-Switch connected to a port Converting the read value to BCD Converting BCD digits to se2-7 and displaying the result by 2 7-piece displays
7. Working with a matrix keyboard (designing the keyboard and displaying the numbers read from the keyboard by 7-piece displays)
8. Working with LCD (connecting LCD to microcontroller and displaying the information received from the keyboard by it) 9. Writing and reading data from EEPROM microcontroller memory
9. Working with counter timer: Microcontroller in normal operation mode and CTC (flashing or two LEDs connected to two pins of one port that turn on and off alternately)
10. Working with counter timers 0 and 1 (making digital frequency meter)
11. Work with timer 2 in PWM mode (adjust LED brightness by adjusting motor speed by PWM wave)
12. Working with a microcontroller analog comparator (turning on and off an LED by increasing or decreasing the voltage of one of the analog comparator inputs relative to the other input)
13. Working with analog to digital microcontroller converter (temperature measurement by making a voltmeter with resistance meter and display on LCD)
14. Working with the USART microcontroller interface (communication between two microcontrollers by communicating between a microcontroller and a computer via RS232 communication using the USART interface on the microcontroller side and the USART Serial Connection.NET Component or Hyper Terminal on the computer side)
15. Working with SPI microcontroller interface (communication between two microcontrollers via SPI interface or writing and reading data in SD RAM)

2.14. Digital Systems Design

Familiarity with automated methods of designing and debugging digital circuits and systems with the approach of using tools | Automatic design of integrated circuits.

Name of Course	Digital Systems Design
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Computer Architecture
Reference books	[1] S. Palnitkar, Verilog HDL: A Guide to Digital Design and Synthesis. SunSoft Press, 2nd ed. 2003. [2] V. A. Pedroni, Circuit Design with VHDL, MIT Press, 2011. [3] C. Maxfield, The Design Warrior's Guide to FPGAs: Devices, Tools and Flows.Elsevier Pub., 2004.

At the end of the module, students are able to:

- Ability to describe and design large circuits and hardware systems with hardware description languages with emphasis on complexity control skills
- Familiarity with digital circuit design tools
- Familiarity with programmable chips (FPGA and CPLD), interior architecture and their useful features for professional design

Course Objectives:

1. Introduction and basic concepts
   - History of digital systems
   - investigating the growth trend of digital systems design industry
   - automated hardware design tools and languages
   - ASIC and FPGA design cycles and their comparison
   - Hardware design styles
   - Abstract levels of hardware design
2. Hardware description languages
   - Reasons for the need for hardware description languages versus schematic methods
   - Key features of a hardware description language
   - Companion as a distinctive feature of hardware description languages
   - Conventional hardware description languages and their comparison
   - Verilog / VIIDL language features
   - Compare VIIDL / Verilog with other hardware description languages
   - Hardware simulation methods

3. VHDL/Verilog descriptive language training
    - Delay model in the target language
    - Types of language data
    - Hardware description method at different levels of behavior, data flow and structure)
    - Specific features of the desired description language
    - Test bench design method
    - Design of sequential and functional blocks with the desired language
    - Parametric design with generic
    - Large hardware complexity management methods
    - Describe organizing techniques
    - Top-down design method and bottom-up design method
    - Types of methods for describing state machines with hardware description language and methods for binary state coding, coding, etc.
    - Pipeline design and how to describe it at the level of stability transfer
4. Hardware synthesis
    - Concepts of behavioral, logical and physical synthesis
    - Steps to perform logical synthesis
    - The concept of a centrifugal subset and the necessary considerations in describing a synthesize
    - Simulation and testing after synthesis
    - Design based on constraints
    - Static time analysis (STA) methods and introduction of Slack parameter
    - How to optimize the design criteria of area speed and power consumption using tools
    - An Overview of High Speed and Low Power Circuit Design Techniques
    - An overview of test circuit design techniques

2.15. Design of Algorithms

The purpose of this course is to teach methods of analysis and design of algorithms. In this lesson, students learn how to analyze a problem and find a variety of possible algorithms to solve it. Then find algorithmic solutions based on each type, analyze and compare them in terms of computational complexity, and select the best one for a specific engineering application based on the size and input characteristics of the problem. In this course, basic algorithms for solving practical and common problems will be presented to students.

Name of Course	Design of Algorithms
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Data Structures
Reference books	[1] T. Cormen, C. Leiserson, and R. Rivest. Introduction to Algorithms. McGraw-HillInc., 2001.

At the end of the module, students are able to:

- Have a general understanding of algorithmic problem-solving methods
- Familiarize yourself with NP-complete issues and prove that an issue is NP-complete
- Familiarize themselves with basic graph algorithms
- calculate the temporal complexity of an algorithm
- Understand common and essential algorithms and compare their different solutions in terms of complexity, and know where to use the algorithm.
- Use existing library functions for standard algorithms

Course Objectives:

1. Different types of problem-solving methods
2. Dynamic programming
3. Backpack issues
4. Greedy methods
5. Interval scheduling issues
6. Breaking Analysis
7. Advanced data structures
8. Separate collections
9. The shortest path between all graph nodes
10. Most flow
11. String matching

2.16. Artificial Intelligence and Expert Systems

Artificial intelligence is one of the essential branches of computer science; the ultimate goal is to model humans' intelligence and brilliant performance. AI itself includes a wide range of sub-disciplines, including general topics. Leaper focuses on machine learning and various concepts to more specific topics such as playing chess, automatic proof of mathematical theorems, and diagnosis of diseases. In this course, while presenting the basic concepts of artificial intelligence, including various methods of searching and displaying knowledge, some sub-branches of artificial intelligence are also briefly introduced.

Name of Course	Artificial Intelligence Expert Systems
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Data Structures
Reference books	[1] S. Russel and P. Norvig Artificial Intelligence: A Modern Approach. 3rd edition, Prentice Hall, 2010. [2] C. S. Krishnamoorthy and S. Rajecy, Artificial Intelligence and Expert Systems for Engineers. CRC Press, 1996.

At the end of the module, students are able to:

- Understanding the intelligent agent and the structure of an intelligent agent
- Recognizing the solution of artificial intelligence problems by search method
- To understand the concept of revelation (initiative) in solving artificial intelligence problems
- Knowledge of knowledge-based factors
- Familiarity with first-order logic as a language of knowledge representation in knowledge-based agents
- Familiarity with planning issues
- Familiarity with solving artificial intelligence problems in uncertain environments
- Familiarity with the concept of learning to use observations
- Familiarity with the concept of the robot, the concept of pCTccpt, inference, and execution by robots

Course Objectives:

1. Introduction and History
2. Intelligent agents
3. Solve problems by searching
4. Types of smart and non-smart searches

5. Solve problems with constraints by searching
6. Minimum search Maximum
7. Logic factors, resolution proposition logic, forward and backward chains
8. Assign the first rank and the types of inference in it
9. Systems based on logical inference
10. Uncertainties and systems based on possible inference
11. Decision systems
12. Building a knowledge base, knowledge presentation methods

2.17. Computer Architecture

The purpose of this course is to acquaint computer engineering students with the architecture and organization of processors, which includes familiarity with instructional architecture and the internal structure of the processor. Students will also become familiar with computer calculations used in general-purpose processors, including displaying numbers and basic operations including addition, subtraction, multiplication, and division in various numerical systems. The following is the hierarchy of memory in processing systems. Since modeling and testing of different architectures, which is one of the lesson's objectives, is possible by using hardware description languages, the use of Verilog language and recall of its basic concepts while teaching the class is recommended.

Name of Course	Computer Architecture
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Logic Circuits
Reference books	[1] D. A. Patterson and J. L. Hennessy, Computer Organization and Design: The Hardware/Software Interface. 4 Edition, Morgan Kaufmann Publishers Inc., 2010.

At the end of the module, students are able to:

- Various low-order / predestined architectures
- Processor performance analysis
- Design and implementation of processors
- Computer account algorithms in processors
- Design of peripherals and their connection to the processor
- Familiarity with the language of Or Bella and simulating the fundamental structures of computer architecture with it

Course Objectives:

1. Introduction
2. Introduction and basic concepts
3. Introduction of Verilog hardware modeling language
4. Processor design
5. Introduction of pipeline mechanism
6. Memory hierarchy
7. Computer account
8. CPU peripherals
9. Introduction of multicore processors

2.18. Operating Systems

This course aims to provide a proper understanding of the relationship between application software and hardware and resource management methods and algorithms for computer engineering students. In addition, students in this course will learn robust methods for managing low-level computer systems. In addition, designing such systems taking into account the characteristics and limitations of hardware and software, and improving the program's quality is another goal of this course. This course is a combination of theoretical and practical topics.

Name of Course	Operating Systems
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Logic Circuits - Computer Architecture
Reference books	[1] P. Silberschatz, B. Galvin, and G. Gagne, Operating System Concepts. 81h Edition,John Wiley Inc., 2010.

At the end of the module, students are able to:

- Types of computer systems and their applications
- Design, build and manage software systems
- Ability to discover the reasons for the decline in the efficiency of computer systems and solve their problems
- Create resource management policies based on system conditions

Course Objectives:

1. Introduction and operating system structures
2. Process management
3. Strings
4. CPU scheduling
5. Sync processes
6. Deadlock management
7. Main memory management
8. Secondary memory management
9. Input-output management
10. Mass memory structures

2.19. The Theory of Formal Languages and Automata

This course is about the theoretical aspects of computer engineering and identifies the relationship between problems and languages. Topics include different computational models, computational ability of these models, formal expression of models and grammars, their computational properties, and their applications. Other topics include the concepts of computability, decision making, and Church and Turing theses on algorithms. This course covers essential knowledge for compiler courses in algorithm design, computational theory, and courses related to the formal description and modeling of computer systems.

Name of Course	The Theory of Formal Languages and Automata
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Data Structures
Reference books	[1] P. Linz, An introduction to formal languages and automata, 5th Edition, Joncs and Barlett Publishers, 2011. [2] M. Sipser, Introduction to the theory of computation. 2nd Edition, PWS Publishing Company, 2006.

At the end of the module, students are able to:

- Basic knowledge for identifying solvable and indecisive issues
- Basic knowledge for identifying solvable and indecisive issues
- Write grammar to solve different problems (different languages)
- Design machines to distinguish language strings from different classes

Course Objectives:

1. Introductory topics: propositional logic, belonging to the document, proof system, Russell parallax set theory, numerical and innumerable sets, languages, and grammars. Uncertainty theory.
2. Regular languages: definite finite acceptors, indefinite finite acceptors, conversion of indefinite to definite finite acceptors, definite finite acceptors of K-min, native languages, regular expressions, linear right-hand grammars, linear left-hand grammars, regular grammars, language hospitalization Regular, decision-making and regular languages, Irregular languages, Lamp temping for regular languages
3. Text-independent languages: text-independent grammars, text-independent languages, chord derivation, right-derivative derivation, derivation tree, ambiguous grammars, anonymous grammars, inherently ambiguous languages, obscure languages, simplification of text-independent grammars, independent

grammars Chomsky natural text, independent text grammars Graybach natural, membership issue, CYK algorithm, push-through machines, the equivalence of push-pull machines and text-independent grammars, definite push-through machines, definite text-independent languages, languages Non-text independent, Lump Pumping for text-independent languages, Hospitalization features and decision-making of text-independent languages

4. Text-sensitive languages, linear bounding machines, and text-sensitive grammars
5. Unlimited languages, Turing machines, and their types and unlimited grammars
6. Hierarchy of official languages
7. Inoculability

2.20. Computer Networks Laboratory

This derby aims to create a correct understanding and practical experience of the basic concepts of computer networks. For this purpose, while introducing the tools and equipment used to establish computer networks, experiments that have included theoretical topics should be performed.

Name of Course	Computer Networks Laboratory
Number of Credits	1
Number of Hours	16
Cross section	Bachelor's Degree
Needs	Computer Networks
Prerequisites	Does not have
Reference books	[1] S. Panwar, S. Mao, J. Ryoo , Y. Li , TCP/IP Essentials: A Lab-Based Approach. Cambridge University Press, 2004.

At the end of the module, students are able to:

- Ability to work with Troubleshooting tools
- Ability to work with Packet Capturing tools
- Ability to work with Packet Crafting tools
- Ability to configure network equipment
- Ability to set up a LAN with multiple VLAN
- Ability to establish connections between multiple LAN using dynamic static routing protocols
- Provide test results in the form of technical reports

Course Objectives:

1. An overview of computer networking lessons
2. Introducing Troubleshooting tools such as Ping, Traceroute, Arpping
3. Introducing and using Packet Capturing tools such as TCPDump and Wireshark
4. Create and send third- and fourth-layer packages using Packet Generator tools and change field values with the Stay tool
5. Set up a Client Server connection
6. Initial configuration of VLAN and Trunk configuration switches and routers
7. Static and dynamic routing configuration (OSPF, RIP)
8. Setting up a Domain Name Server (DNS)
9. Set up a DHCP server

2.21. Engineering Mathematics

The class covers basic mathematical methods for engineers, including differentiation and integration, Taylor's expansion, linear system's resolution, matrix formalism, partial differential equations, Laplace, Fourier, and Legendre transforms statistics and probability. The class will be mainly oriented towards exercises and problem-solving.

Name of Course	Engineering Mathematics
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Calculus II - Differential Equations
Reference books	[1] E. Krcyszig, Advanced Engineering Mathematics. 10 cd., Wiley, 2011. [2] C.R. Wylie, Advanced Engineering Mathematics, 6h ed., McGraw-Hill, 1995.

Course Objectives:

1. Fourier series and Fourier transform: Fourier series definition, O's formula, half-domain regression, Toussaint induced Fourier integral, Laplace transform: Gibbs theorem and Fourier analysis constraints, orthogonal and complete functions of series convergence and Fourier transform, application of Quarry analysis in problem-solving Engineering, symmetric properties of Fourier transform and Fourier series, short Fourier transform mosaic
2. Partial Equations: Univariate Wave Equation, Variable Separation Method, Dalamir's Answer for Wave Equation, Heat Emission Equation, Wave Equation, Laplace Equation in Cartesian, Spherical and Polar Equations, Elliptic, Parabolic and Hyperbolic Equations Partial, Solving Partial Equations Using the Fourier Integral, Extracting Telegraphic Equations
3. Analytical and mapping functions of coherence and complex integrals: limit and continuity, derivative of complex functions, exponential and multiplicative functions. Hyperbolic and logarithmic, inverted and exponential trigonometry with a mixed view, coherence mapping of Dirichlet boundary condition under coherence conversion, energy conservation under coherence conversion, application of coherence conversion in electrical capacity calculations
4. Linear integrals in the mixed plane: Cauchy integral theorem, calculation of linear integrals by indeterminate integrals, Cauchy formula, tabular and McLaren extensions, calculation of integrals by residuals, residual theorem, calculation of true fractional integrals, calculation of real trigonometric integrals

2.22. Principle of Compiler Design

Designing and building compilers is one of the basic concepts of computer science. Although the methods of making compilers are slightly varied, they can be used to make interpreters and translators from a wide variety of languages and machines. Becomes. After an initial introduction to the components of a compiler and the types of grammars, the various stages of translation such as lexical analysis, semantics and semantics, and code generation and payment are described.

Name of Course	Principle of Compiler Design
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Data Structures
Reference books	[1] Alfred V. Aho, Ravi Sethi, and Jeffrey D. Ullman, Compilers: Principles, Techniques, and Tools. Second Edition, Boston: Addison-Wesley, 2007.

At the end of the module, students are able to:

- Familiarity with compiler implementation and various techniques for their implementation
- Understand the execution of commands in programming languages
- Gain skills in producing optimal programs and fixing programming errors
- Familiarity and application of automated tools in compiler production

Course Objectives:

1. Introduction
2. Types of languages and grammars
3. Verbal analysis and correction of verbal errors
4. Analysis
5. Top-down decomposition methods
6. Bottom-up decomposition methods
7. Operator precedence
8. Simple precedence
9. Decomposition (1) LR includes (1) LALR (1) SLR And (1) CLR
10. Semantic analysis
11. Manage the symptom table
12. Runtime memory allocation methods
13. Generate code
14. Payment and code optimization
15. Automatic production of compilers

2.23. Methods of Research and Presentation

The purpose of this course is to get acquainted with and acquire skills in the principles and methods of conducting research, the principles of preparing various types of written presentations, issues raised in the implementation of various types of speech donations, and familiarity with relevant tools.

Name of Course	Method of Research and Presentation
Number of Credits	2
Number of Hours	32
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Technical English
Software	Word, PowerPoint, OneNote, Project, EndNote, .

At the end of the module, students are able to:

- How to choose a topic for research
- How to do research
- Research results in the form of reports with articles
- Delivering a technical lecture by observing the necessary principles
- Familiarity with useful software in conducting research and presentation

Course Objectives:

1. Basic definitions and concepts related to research
2. Defining the research topic and choosing the title
3. Develop a research plan
4. Review subject records, study, and take notes
5. Methods and essential points in conducting the experimental practical part of the research
6. Important principles in writing engineering reports, preparing final design reports
7. Special points for different types of written presentations
8. Principles in speech presentations
9. Special tips for different types of speech presentations

3

Basic Course

3.1. General Mathematics I (Calculus I)

This calculus course covers the differentiation and integration of functions of one variable and concludes with a brief discussion of infinite series. Calculus is fundamental to many scientific disciplines, including physics, engineering, and economics.

Name of Course	General Mathematics I (Calculus I)
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Does not have
Reference books	[1] James Stewart, Single Variable Calculus: Concepts and Contexts. 4th edition, Cengage Learning, 2009. [2] George Simmons, Calculus with Analytic Geometry, 2nd edition, McGraw-Hill Science/Engineering/Math, 1996. [3] Tom Apostol, Calculus, Vol. 1: One-Variable Calculus, with an Introduction to Linear Algebra, Wiley: 2nd edition, 1991.

Course Objectives:

1. Cartesian coordinates and polar coordinates
2. Complex numbers, addition and multiplication, and root and geometric representation of complex numbers, innate representation of complex numbers
3. Algebra Functions
4. Derivative commands, inverse function, and its derivative, the derivative of trigonometric functions and their inverse functions Relay theorem, mean factor

5. Geometric and physical applications of the derivative, curves, and acceleration in polar coordinates, application of the derivative in approximating the roots of equations
6. Definition of integrals of continuous functions and fragments, basic theorems of differential and integral calculus, primary function. Approximate methods of integral estimation
7. Application of integral in calculating the area, volume, length of a curve, torque and center of gravity and work - (in Cartesian and polar coordinates)
8. Logarithms and exponential functions and their derivatives, hyperbolic functions
9. Integration methods such as variable and fractional substitution and fractional decomposition
10. Some substitution of special variables, sequence and numerical series and related theorems, power series, and Taylor theorem with the remainder of the Taylor expansion

3.2. General Mathematics II (Calculus II)

Calculus II is a continuation of Calculus I, covering integration and infinite series. It is designed for students working on a degree in science, mathematics, computer science, and those planning on certain types of graduate work.

Name of Course	General Mathematics II (Calculus II)
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Calculus I
Reference books	[1] James Stewart, Multivariable Calculus. Cengage Leaming, 7th edition, 2011. [2] Tom Apostol, Calculus, Vol. 2: Multi-Variable Calculus and Linear Algebra with Applications to Differential Equations and Probability. Wiley, 1969. [3] George Simmons, Calculus with Analytic Geometry, 2nd edition, McGraw-Hill Science/Engineering/Math, 1996. [4] Ron Larson and Bruce Edwards, Calculus Multivariable, Cengage Learning; 9th edition. 2009.

Course Objectives:

1. Parametric equations
2. Spatial coordinates
3. Vector in space and types of multiplication vectors
4. 3*3 matrices, three-dimensional linear equation set, inverse matrix, equation set solution, linear independence, base in R3□,R2□ linear conversion, Determinant 3*3. Value and special vector
5. Second-order line, plane, and procedure equations
6. Vector and its derivative function, speed, and acceleration. Curvatures and vertical vectors of any curve
7. Multivariate function. General and partial exercises, tangent plane and gradient vertical line, chain rule for partial derivative, full differential
8. Dual and triple integrals and their application in Handy-C and physical problems, variable change in internalization without an exact proof (cylindrical and spherical coordinates)
9. Vector field, linear curve integral. Integral procedures of divergence, Laplace in, potential, Green and divergence theorems and Snox

3.3. Differential Equations

First-order ordinary differential equations and initial value problems; higher-order differential equations; vector spaces, matrices, determinants, eigenvectors, and eigenvalues; applications to systems of first order equations; Laplace transforms.

Name of Course	Differential Equations
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Calculus I
Reference books	[1] Yunus Cengel and William Palm, Differential Equations for Engineers and Scientists. McGraw-Hill Science Engineering Math, First edition, 2012.

Course Objectives:

1. The nature of differential equations and their solutions
2. The family of curves and vertical spouses
3. Detachable equation
4. First-order linear differential equation, homogeneous equation
5. Second-order linear equation, homogeneous equation with constant coefficients Method of indeterminate coefficients, method of changing parameters
6. Application of second-order equations in physics and mechanics
7. Solve the differential equation with series, Bessel, and gamma functions
8. Legendary polynomial
9. Introduction to the system of differential equations
10. Laplace transform and its application in solving differential equations

3.4. Engineering Statistics and Probability

After successful completion of the module: 1) Participants are familiar with essential concepts of discrete and continuous probability spaces and stochastic processes and can in large part deduce them themselves, 2) master calculation rules for the determination and estimation of probabilities, expected values, and variances, 3) can map real problems to abstract probability spaces, and 4) can easily apply simple statistical tests.

Name of Course	Engineering Statistics and Probability
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Calculus II
Reference books	[1] Alberto Lcon-Garcia, Probability, Statistics, and Random Processes for Electrical Engineering, Prentice Hall, 3rd edition, 2008. [2] Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, and Keying E.Ye. Probability and Statistics for Engineers and Scientists. Pearson, 9th edition, 2011.

Course Objectives:

1. A reference to the tenor of collections, samples and their tabular representation along with mean, view, median, and variance
2. Conversion and combination of related probabilities and theorems
3. Continuous and discrete stochastic variables
4. Mean, mean and variance of binomial distributions, Boasen, supra-geometric, exponential, normal
5. Common distribution of stochastic variables, correlation, and independence of variables
6. Conditional distribution
7. Characteristic and torque generating functions
8. The central limit theorem
9. My Markov Inequality
10. Functions of random variables
11. Refers to the topic of statistics

3.5. Physics I (Heat and Mechanics)

The principles and applications of classical mechanics, including harmonic motion, physical systems, and thermodynamics, are studied with problem-solving. Performance of basic laboratory experiments supporting theoretical physics principles and applications of classical mechanics, including harmonic motion, physical systems, and thermodynamics.

Name of Course	Physics I
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Calculus I
Reference books	[1] D. Halliday, R. Resnick, and J. Walker, Frondamentals of Physics. 9"edition, Wiley,2010.

Course Objectives:

1. Equilibrium: Equilibrium conditions under the influence of forces and torques, relevant laws
2. Moving in one and two dimensions: speed and acceleration, types of motion, ground motion and similar gravity
3. Work and Energy: Introduction, Work, Kinetic Energy, Elastic Potential Energy, Stable and Unstable Forces, Internal Work, Internal Potential Energy, Power and Speed, Mechanical Energy Conservation
4. Impact, the law of momentum conservation, the laws of collision
5. Rotation: Angular velocity, Angular acceleration, Rotation with constant and variable angular acceleration, Interface between angular velocity and velocity and Angular acceleration and velocity, Force torque, Rotational inertia Rotational kinetic energy Beck torque, Angular momentum conservation Rotation around fixed and moving axes of rolling motion
6. Temperature, heat and the first law of thermodynamics, the zero law of thermodynamics, temperature measurement
7. Kinetic Theory of Gases: Complete Gases, Transitional Kinetic Energy, Mean Free Scan, Degree of Release, and Molar Specific Heat
8. Entropy and the second law of thermodynamics: One-way process, change in entropy, second law of thermodynamics

3.6. Physics II (Electricity and Magnetism)

Electric charge and Coulomb's law, the electric field, Gauss' law - Electric Potential Energy and Potential, the electric properties of materials, Capacitance and Capacitors, DC Circuits - the magnetic field and magnetic field of a current, Faraday's law in Induction, magnetic properties of materials, inductance, AC circuits , Ampere's law.

Name of Course	Physics II
Number of Credits	3
Number of Hours	48
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Calculus I
Reference books	[1] D. Halliday, R. Resnick, and J. Walker, Fundamentals of Physics. 9th edition, Wiley,2010.

Course Objectives:

1. Electric charge: electric charge, Coulomb law, electric charge conservation
2. Electric field: electric field of field lines, the electric field of a point charge, the bipolar electric field of charged line, the motion of point charge in an electric field, a bipolar period in an electric field
3. Gaussian law: electric flux, the flux of an electric field, the relationship between Gaussian law and the charge-conducting coulomb of a mollusk, applications of Gaussian law
4. Electric potential: Electric potential energy, electric potential, point charge potential, charge particle group potential, the potential of a continuous distribution, Milan calculation of potential
5. Capacity: Capacitive capacity, calculation of energy storage capacity in an electric field, capacitor or dielectric, dielectrics, and Gaussian law
6. Electric current and electrical resistance: current, resistance, ohm law of power in an electrical circuit
7. Magnetic fields The magnetic field strength of orthogonal fields (Hall effect). Magnetic force on a moving load Magnetic force on a current-carrying wire, torque on a current loop, amplitude magnetic field of electric current generated by a current: calculation of magnetic field by the current of ampere law, the magnetic field of winding and torque, carrier tangle as Magnetic dipoles
8. Magnetic field from electric current: Calculation of magnetic field from current, Amir's law, Sampi and torque magnetic field, carrier tensile as a magnetic dipole
9. Induction: Faraday's law, induction law, and energy transfer, induction electric fields, torches, self-induction, RL circuits of energy stored in the magnetic field, the energy density of the magnetic field of mutual induction

10. Electromagnetic oscillations and alternating current: LC oscillations, alternating current, power RLC circuit in alternating current circuits, Maxwell equations of displacement current
11. Electromagnetic waves: moving electromagnetic wave, energy transfer and pointing vector
12. Familiarity with the duality of wave and particle. Light interference, light diffraction, relativity, hydrogen atom radiation

3.7. Computer Workshop

Laboratory and practices of Computer Basic Course.

Name of Course	Computer Workshop
Number of Credits	1
Number of Hours	16
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Fundamentals of Computer Programming

Course Objectives:

1. Familiarity with the principles of safety and health of workshops and how to use the tools and equipment used
2. Familiarity with basic topics in practical work with computers
3. Familiarity with basic topics in practical work with computer systems hardware
4. Familiarity with basic topics in practical work with computer systems software
5. Familiarity with the basic topics in information security in computer systems
6. Familiarity with basic topics in software security, hardware, and e-commerce systems
7. Macro writing in software management and computer systems
8. Connect, use, and program various software in a smart work environment and wireless communication Bluetooth connections. Zigby and LAN
9. Practical work with connections and network cards and microcontrollers in a network environment
10. Writing simple web applications and designing web pages

3.8. Physics Laboratory II (Electricity and Magnetism)

Practical Experiments Related to physics II Course.

Name of Course	Physics Laboratory II
Number of Credits	1
Number of Hours	16
Cross section	Bachelor's Degree
Needs	Does not have
Prerequisites	Physics II
Reference books	[1] D. Halliday, R. Resnick, and J. Walker, Fundamentals of Physics. 9th edition, Wiley,2010.

Course Objectives:

1. Investigation of Ohm and Kirchhoff Laws in Direct Current Circuits Introduction to Meter-Ammeter-Ohm Meter)
2. Investigation of direct current measurement bridges
3. Investigation of capacitor charge and discharge in RC circuit with step excitation
4. Stable response of RC circuit to sinusoidal excitation
5. Stable response of RL circuit to sinusoidal stimulation
6. Magnetism and electric current (study of lens and Faraday laws) Familiarity with the oscilloscope
7. Check the basis of the work of transformers
8. Ferromagnetic Investigation of the effect of the ferromagnetic nucleus on the magnetic field)
9. DC generator with dynamo

4

General Courses

4.1. Islamic Ideology I
Focused on Islamic Laws and Regulations.

4.2. Islamic Ideology II
Based on Rational and Philosophical proofs of God and other Islamic Concepts.

4.3. Islamic Revolution of Iran
Studies focused on Islamic Revolution happened in 1979 and the upcoming events. Also, what leads to this revolution.

4.4. The History of Islamic Culture and Civilization
Focused on Iran History and Islamic Civilization in Arab world.

4.5. Thematic Interpretation of Qu'ran
Interpretation of Qu’ran.

4.6. General Persian
Mainly focused on Persian literature, academic, and formal writing.

4.7. General English Language
Focused on four English Skills: Writing, Speaking, Listening and Reading.

4.8. Physical Education
General Excreting and Aerobic and Core Muscle Exercises.

4.9. Sports 1
Specialized in one of the following sports: Football, Volleyball, Basketball, Marathon Running, Table Tennis or Badminton.

4.10. Family and Population Knowledge

Mainly Focused on Marriage and Sexual Education.