**Course Description**

**General Mathematics I (Calculas I)**

This calculus course covers 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.

**Mechanical Physics**

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

**Physical Education**

General Excreting and Aerobic and Core Muscle Exercises

**General English Language**

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

**The History of Islamic Culture and Civilization**

Focused on Iran History and Islamic Civilization in Arab world.

**General Mathematics II (Calculas 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.

**Electrophysics and Magnetism**

Principles of electricity and magnetism, including circuits, electromagnetism, waves, sound, light, and optics are studied. Performance of basic laboratory experiments supporting theoretical physics principles and applications of electricity and magnetism, including circuits, electromagnetism, waves, sound, light, and optics. Also includes experimental design, data collection and analysis, and preparation of laboratory reports.

**Electrophysics and Magnetism Lab**

Laboratory and practicies of Electrophysics and Magnetism

**Islamic Ethics**

Based on different Religious references, mainly focused on Ethics in sociaty and family.

**General Persian**

Mainy focused on Persian literature, academic and formal writing.

**Physical Education II**

Speciallized in one of the following sports: Football, Voleyball, Basketball, Marathon Running, Table Tennis or Badminton.

**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.

**Public Workshop**

Students learn how to work with workshop tools like saws and metal lathes.

**Specialized Language**

Advance English focused on Engineering vocabulary and scientific papers grammer and style.

**Thematic Interpretation of Qu'ran**

Interpretation of Qu’ran.

**Electerical Circuits**

understand basic concepts of DC and AC circuit behavior, develop and solve mathematical representations for simple RLC circuits, understand the use of circuit analysis theorems and methods.

**Electronic Circuits**

The course introduces the fundamentals of the lumped circuit abstraction. Topics covered include: resistive elements and networks; independent and dependent sources; switches and MOS transistors; digital abstraction; amplifiers; energy storage elements; dynamics of first- and second-order networks; design in the time and frequency domains; and analog and digital circuits and applications.

**Data Storage and Retrieve**

It focuses on the File System core technology by studying topics such as storage systems (data layout, disk-based data structures, Long-lasting memories), indexing, query processing algorithms, query optimization, transactional concurrency control, logging and recovery.

**Circuits and Electrical Measurement Lab**

Laboratory and practicies of Electerical Circuits course.

**Islamic Thoughts**

Focued on Islamic Laws and Regulations.

**Technical and Scientific Contents Prenestations Method**

In this course, students will develop the scientific and technical reading, writing and presentation skills they need to understand and construct research articles and technical reports, PowerPoints and LaTex.

**Electronic Lab**

Laboratory and practicies of Electronic Circuits course

**Islamic Revoloution of Iran**

Studies focused on Islamic Recoloution happened in 1979 and the upcoming events. Also, what leads to this revoloution.

**Family and Population Knowledge**

Mainly Focused on Marriage and Sexaul Education.

**Humans in Islam (Islamic Thoughts II)**

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

**Engineering Mathematics**

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

**Principles of Engineering**

Mainly Focused on Enviromental Concerns, General Ethics, and general mathematics like matrixes.

**Computer Basics**

This course introduces software engineering students to elementary data structures, and to the functional programming paradigm. Students learn programming concepts by creating interactive visualizations, simple games, and image processing effects. They also explore fundamental imperative programming language concepts, such as variables, conditionals, loops, functions, events, and arrays.

**Computer Basics Lab**

Laboratory and practicies of Computer Basic Course

**Advanced Programming I**

This lecture gives an introduction to programming using the Java language. The basic principles of object-oriented programming (objects, classes, usage, inheritance) are presented and practiced.

**Discrete Mathematics**

On successful completion of the module, students will be able to 1) understand the elementary vocabulary of discrete mathematics and use logic algebraic und algorithmic calculios, 2) solve combinatoric problems, 3) model and solve problems using graph theory, and 4) do a quantitive analysis of the efficiency of algorithms.

**Engineering Probability and Statistics**

After successful completion of the module: 1) Participants are familiar with important 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) are able to map real problems to abstract probability spaces, and 4) can easily apply simple statistical tests.

**Theory of Languages and Machines**

After successfully completing this module, the students understand the core concepts of the theory of computation on a basic but scientific level. They know what regular expressions, contextfree grammars, the Chomsky hierarchy, finite automata and Turing machines are. They can define formal languages with the appropriate grammars or machines. They can prove that a given language cannot be defined with a given class of grammars or machines. They can prove that certain grammars and machines are equivalent and they can transform them into each other algorithmically. They can explain the basic concepts of complexity theory and can reduce decision problems algorithmically to each other under given complexity limitations.

**Data Structures**

Theoretical and practical exercises deepen the knowledge imparted with the aim of being able to design, analyze and implement data structures and algorithms yourself, or to be able to use them sensibly in your own 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 successfully complete the course can use these tools to design and develop efficient programs for a wide variety of applications.

**Machine Languages and System Programming**

This course introduces hardware and software concepts used in computer systems. Specific topics include machine-level programming, memory organization, and basic I/O mechanisms.

**Desiging Algorithms**

To study efficient algorithms, effective algorithm design techniques and approaches to handling situations in which no feasible algorithms are known. The course is intended to give the student experience in program design and to emphasize both pragmatic and mathematical aspects of program efficiency.

**Logic Circuits**

Covers the design and application of digital logic circuits, including combinational and sequential logic circuits. After taking this course students will be able to recognize and use the following concepts, ideas, and/or 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, and4) Design criteria, including area, speed, power consumption, and testability.

**Programming Languages Design and Implementation**

This course examines foundational issues in contemporary programming languages, including abstraction mechanisms, operational semantics, type systems, and extensibility.

**Object-Oriented Analysis and Design**

This course introduces students to the requirements definition phase of software development. It discusses processes, models, and notations, and processes for software requirements elicitation, identification, analysis, modeling, representation, specification, and validation. An important component is a group project: the software requirements specification of a large software system.

**Artificial Intelligence**

This course introduces students to the fundamental problems of artificial intelligence and the basic models and algorithms used to tackle these problems. Students examine frontier areas of computer science and gain knowledge that will allow them to further their studies in artificial intelligence.

**Database Design Principles**

Students are able to 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 are able to 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 safe-guard a database application with respect to recovery, concurrency control and authorization. The course also covers database design methodology. main course topics are data models, architecture of database systems, data definition and manipulation languages, database design methods, and the theory of data dependencies and relational normalization.

**Computer Architecture I**

This course enables students to develop an accurate mental model of the architecture and organization of physical computers and it introduces them to theoretical and practical concepts relevant to the structure and design of modern digital computers. The course also helps students understand and predict the behaviour and performance of computer programs executing on real machines.

**Software Engineering I**

To provide students with an appreciation for software engineering tools and methodologies in the construction of computer-based systems.

**Software Engineering Lab**

Laboratory and practicies of Sofware Engineering I.

**Compiler Design Principles**

This course covers the basic structure of compilers for Algol-like languages. A major part of the course involves implementing a compiler for a simplified Algol-like language. Students discover software tools and techniques that are applicable to both compilers and the implementation of system utility routines, command interpreters, etc.

**Logic Circuits Lab**

Laboratory and practicies of Logic Circuits course

**Operating Systems**

This course introduces operating systems, what they do, how they are used, and how they are implemented.

**Microprocessors I**

Upon completion of the module, students are able to understand and analyse: 1) the architecture of a microprocessor in general, 2) features that have been added over processor generations since the first microprocessor, 3) the state of the art in modern microprocessor architectures, 4) possible trends with regard to future processor architectures, and 5) techniques to program and optimise different architectures at assembly language level.

**Computer Networks**

This course introduces the fundamentals of network architectures and protocols and focuses on protocols used in the Internet.

**Computer Architecture Lab**

Laboratory and practicies of Computer Architecture course

**Database Lab**

Laboratory and practicies of Database Design Principles course

**Software Apprenticeship**

In the software engineering internship, our students develop software projects from A to Z. They work independently in groups. There is also a weekly meeting where you have to present various aspects of the projects (organizational, technical, etc.).

**Software Project**

This course provides exceptional students with the opportunity to do research under the supervision of a professor and to write a thesis summarizing results.

**Microprocessor Lab I**

Laboratory and practicies of Microprocessor course

**Operating Systems Lab**

Laboratory and practicies of Operating Systems course

**Security in Computer Systems (Optional)**

This course introduces students to security and privacy issues that affect various aspects of computing, including programs, operating systems, networks, databases, and Internet applications. The course examines the causes of security and privacy breaches and provides methods to help prevent them.

**Advanced Programming II (Optional)**

Mainly Focused on mobile application programming, Web Services and APIs, and more advanced topics.

**Concurrent Programming (Optional)**

This course introduces advanced control-structures with an emphasis on concurrency and writing concurrent programs at the programming-language level. Programming techniques and styles are examined to express complex forms of control flow, such as multi-level loop exit, exceptions, coroutines, and concurrency.

**Computer Simulation (Optional)**

In this course, students will learn how to do computer simulations of such phenomena as orbits (Kepler problem and N-body problem), epidemic and endemic disease (including evolution in response to the selective pressure of a malaria), and traffic flow in a city (with lights, breakdowns, and gridlock at corners). The simulations are based on mathematical models, numerical methods, and Matlab programming techniques that will be taught in class.

**Data Mining (Optional)**

At the end of the module students are able to:  
- 1. produce scripts that automatically generate data analysis report  
- 2. import data from various sources into R  
- 3. apply the concepts of tidy data to clean and organize a dataset  
- 4. decide which plot is appropriate for a given question about the data  
- 5. generate such plots  
- 6. know the methods of hierarchical clustering, k-means, PCA  
- 7. apply the above methods and interpret their outcome on real-life datasets  
- 8. know the concept of statistical testing  
- 9. devise and implement resampling procedures to assess statistical significance  
- 10. know the conditions of applications and how to perform in R the following statistical tests: Fisher test, Wilcoxon test, T-test.  
- 11. know the concept of regression and classification  
- 12 apply regression and classification algorithms in R  
- 13. know the concept of error in generalization, cross-validation  
- 14. implement in R a cross-validation scheme.