

Course Description

11000	<p>Computer Skills Placement Test Pre-requisites: -</p> <p>The test MUST include all topics mentioned in 2100: Computer skills (s. table below). Student MUST pass this test to be able to go directly to 2103: Introduction to Computer Science. If the student fails in the test then she/he MUST (mandatory) take 2100 before she/he can take 2103.</p>
11100	<p>Computer Skills (Remedial) Pre-requisites: -</p> <p>Introduction to computers: historical, components, functionality. Introduction to computer software and hardware. Programming Languages. Introduction to operating systems. Introduction to word processing with practical applications in preparing homework and reports. Spreadsheets. Computer graphics. Presentation design. Using Internet E-mails.</p>
11102	<p>Introduction to Computer Science Pre -Requisites: -</p> <p>Introduction to computer science. Components of PC and Data representation. Low level data representations (Binary, hexa, octal, conversions, Binary Arithmetic). Introduction to programming computers. Evolution of programming languages and techniques. Problem solving by computers. Flowcharts. Problem solving through analysis and then through an introduction to programming language (Basic program structure, main function, I/O control structures, Functions, Arrays and Structures).</p>
11103	<p>Structured Programming Pre-requisites: 11102</p>

	<p>Concepts of structured programming. Structured programming languages. Program design, development, running, and testing, and debugging programs. Syntax and semantics of the programming language C++. Basic elements of the language: variables, constants, and data types. Basic input/output functions. Conditional and iterative control structures. Concept of procedural programming, Top/down design. Structured decomposition. Functions and parameter passing. Recursive functions. Pointers and dynamic variables. Basic data structures: one and two-dimensional arrays, string manipulation, structures. Input / Output Files, Concepts of OOP and Classes.</p>
11151	<p>Structured Programming Lab Co-requisite: 11103</p> <p>Laboratory sessions on the different aspects and topics of the structured programming using C++.</p>
11201	<p>Geographic Information Systems (GIS)*** Pre-requisites:-</p> <p>The goal of this course is to prepare students to introduce to the students basics of GIS. Specific features of the course include: Coverage of basic geographic, cartographic, and GIS concepts. Topics Include:</p> <ol style="list-style-type: none"> 1) Computer representation of physical, political, statistical, and social aspects of space using vector-based maps. 2) Graphic design principles for using shape, hue, size, and patterns in mapping 3) Attribute-based and graphic feature-based queries for spatial analysis. <p>Geographic Information Systems (GIS) allow individuals and organizations to pose, explore and answer a variety of public- and private-sector questions using spatial data.</p>
11206	<p>Object Oriented Programming Prerequisite: 11103, 11151</p> <p>Object oriented programming concepts and paradigms. Review of control structures, data types and functions. Data</p>

	<p>Abstraction ADT. Encapsulation and information hiding. Classes, attributes and methods (operations). Objects, instantiation, and constructors. Software reuse and reengineering. Inheritance. Overloading. Polymorphism. Templates. Graphical User Interface.</p>
12343	<p>Visual Programming Prerequisite:11206 This course introduces computer programming using a modern visual programming language with object-oriented programming principles. Emphasis is on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger. <i>Learning objectives:</i> Upon completion of this course, the student will be able to: Build programs that use a modern visual programming environment Write object-oriented programs emphasizing object reusability Build state-of-the-art user interfaces for their programs Work in the context of event-driven programming. Work and design: text boxes, labels, forms, buttons, frames, message boxes, input boxes, check boxes, combo boxes, and list boxes. Design, code, test and debug visual programs Write programs with client/server capabilities that interact as clients and servers with respect to database</p>
11212	<p>Data Structures and Introduction to Algorithms Prerequisite: 20134, 11206, 11253 Basics of algorithm design and analysis. Asymptotic analysis of upper and average complexity bounds: best, average, and worst case behaviors. Big "O" notation. Searching and sorting algorithms. Recursion. Data abstraction and review of object oriented concepts. Basic data structures. Sequential and linked representation of data structures. List, Ordered List, Sets, Stack, Queue, tree, Binary trees, graph and network.</p>
12241	<p>Webpage Design and Internet programming Prerequisite:11206 This course focuses on how to design and maintain interactive and dynamic web sites using HTML, Cascading Style Sheets (CSS) and client–side scripting with JavaScript. The students</p>

	<p>will also learn basic Web Page design principles. The goal is to develop effective, pleasing and useful Web sites. In the JavaScript part of the course students will develop real-world projects to learn JavaScript programming, the JavaScript Object Model, JavaScript event handlers, and how to integrate JavaScript programs in a HTML document. Other client-side technologies (Ajax and XML) will be introduced.</p>
12242	<p>Webpage Design and Internet Programming Lab Co-Requisite: 12241</p> <p>Laboratory sessions on how to design interactive and dynamic WebPages. Programming tools: HTML, JavaScript, Ajax and XML.</p>
11253	<p>Object Oriented Programming Lab Co-requisite: 11206</p> <p>Laboratory sessions on the different aspects and topics of object oriented programming.</p>
11313	<p>Algorithm Design and Analysis New Pre-requisites: 11212 3 credit hours</p> <p>Formal techniques of the design and analysis of algorithms. Asymptotic analysis of upper and average complexity bounds. Empirical measurements of performance; time and space tradeoffs in algorithms. Correctness and finiteness of algorithms. Algorithmic strategies: Brute-force, greedy, divide-and-conquer, backtracking, branch-and-bound, heuristics, pattern matching and string/text algorithms. Implementation strategies for Graph, Network and Tree algorithms.</p>
11323	<p>Database Systems Pre-requisites: 11212</p> <p>Basic concepts of databases. DBMS components. Transaction managements. Data modeling. Entity relationships diagrams. Relational databases. Database integrity constraints. Relational Algebra. Query languages. Dependencies, schema designs normalization and redundancy elimination.</p>

13435	<p>Database Management Systems Pre-requisites: 11323 Advanced Topics in DB Systems: Query Processing, Transaction Processing, Recovery System, Concurrency Control, OODB, Distributed Database, Security and privacy, and writing a technical report about a selected topic and in-class presentation.</p>
11335	<p>Operating Systems Pre-requisites: 11212 Introduction to Operating Systems. Processes. Threads. CPU Scheduling. Process Synchronization. Dead-Locks. Memory management. Virtual memory. File System. Mass Storage management. Case Study: UNIX.</p>
11347	<p>Electronic Business Pre-requisites: 11241 Basics of Internet programming. Electronic business. E-Commerce Terms and Concepts. Business models: (B2B), (B2C), and (C2C). Creation of E-Commerce Site: Overview of the technological infrastructure. Software technologies for e-Business. Database Solutions. Payment methods. Security issues. A project is required.</p>
11354	<p>Database Systems Lab. Co-requisite: 11323 How to design and implement a complete database application using a modern relational database system: It covers relations, queries, forms, reports, objects, properties, data design, software design, and rapid application development tools.</p>
11355	<p>Operating Systems Lab. Co-requisite: 11335 This course will provide practical skills needed for using a UNIX type operating system. This will include LINUX installation, Vi environment and commands file and process management commands, email, shell programming, and system administration, in addition to implementing some of operating system concepts, such as memory management or CPU scheduling.</p>

11428	<p>Artificial Intelligence Pre-requisites: 11212</p> <p>Introduction to AI and its scope and applications. AI programming languages. Knowledge representation. Heuristic Search and Problem-Solving with different strategies for solving different types of problems. Introduction to knowledge based systems. Expert Systems. Natural Language processing. Machine learning. Other AI applications. A project is required.</p>
11435	<p>Data Communications and Computer Networks Pre-requisites: 11335</p> <p>Data Communication principles. Network Reference Model. Interfaces and Services. Protocols. Physical Layer. Communication Services. Data Link Layer. Synchronization. Flow Control. Socket Programming. MAN Protocol. Ethernet. Token Ring. FDDI. Wireless Communication. Protocols and Programming. TCP/IP Reference Model. Routing Algorithms. Network Layer. Network Security. Transport Layer. Data Encryption and Decryption. Application Layer.</p>
11436	<p>Distributed Systems Pre-requisites: 11435</p> <p>Concepts of distributed system: advantages, hardware, software; design issues, communication in distributed systems: layered protocols, asynchronous transfer mode networks, client-server model, remote procedure call, RMI, group communication; synchronous: clock, mutual exclusion, election algorithms, atomic transactions, deadlocks; processes and processors: threads, system models, allocation, scheduling; fault tolerance; real time; distributed shared memory: consistency, page, variables, object-oriented based; case studies.</p>

<p>11447</p>	<p>Wireless Networks and Applications Pre-requisites: 11435</p> <p>The purpose of the course is to introduce mobile technology as an educational option and help students to prepare for upcoming change that mobile technology may bring out in the future. In this course students will:</p> <ol style="list-style-type: none"> 1. learn characteristics of mobile learning, 2. learn how wireless application infrastructures and Java technologies might be used in mobile applications. 3. analyze previous mobile learning cases, 4. design lesson plans / instructions that use mobile technology as a critical component. 5. learn how wireless technologies can impact a company's business model and plan the steps to adopt a solution. 6. be introduced to the Java 2 Platform, Micro Edition (J2ME platform) configurations, profiles, and standard extensions. <p>At the completion of the course, participants will be able to understand and explain key concepts and issues of mobile learning.</p> <ol style="list-style-type: none"> 1. Acquire and apply frameworks to analyze mobile learning cases with various pedagogical perspectives. <p>Create mobile learning product for the target audience.</p>
<p>11449</p>	<p>Computer and Society Co-requisite: finish 90 credits</p> <p>Seminars on the social, ethical, and legal issues of computing. Social impact of computerization on local and global organizations. Public perception of computers and computer scientists. Intellectual property: copyrights, patents, trademarks, and commercial law. Computer crime. Economic issues in computing. Privacy and civil liberties. Professional and ethical responsibilities.</p>

11464	<p>Information Systems Security Pre-requisites: 11212</p> <p>This course explains Security protocols, authentication protocols, data integrity, digital signatures, intrusion detection, key management and distribution, viruses and other malicious codes, information flow, mobile code and agent security. Cryptographic algorithms: Secret Key Encryption (DES), Public Key Encryption (RSA), Message Digest Algorithm (MD5); Attacks and countermeasures: Packet sniffing, Spoofing and denial of service; Application layer security: HTTPS, secure email; Transport layer security: TLS, SSL; Network layer security: IP security (IPSec), AH protocol, ESP protocol; access control and Firewalls: Filter-based firewalls, Proxy-based firewalls; wireless networks security, security in IEEE 802.11, WEP protocol, EAP protocol.</p>
13211	<p>Introduction to Software Engineering Prerequisite: 11206</p> <p>Principles of software engineering: Requirements, design and testing. Review of principles of object orientation. Object oriented analysis using UML. Frameworks and APIs. Introduction to the client-server architecture. Analysis, design and programming of simple servers and clients. Introduction to user interface technology.</p> <p><i>Learning objectives</i></p> <p>Upon completion of this course, students will have the ability to:</p> <ul style="list-style-type: none"> • Develop clear, concise, and sufficiently formal requirements for extensions to an existing system, based on the true needs of users and other stakeholders • Apply design principles and patterns while designing and implementing simple distributed systems-based on reusable technology • Create UML class diagrams which model aspects of the domain and the software architecture • Create UML sequence diagrams and state machines that correctly model system behavior • Implement a simple graphical user interfaces for a system • Apply simple measurement techniques to software

	<ul style="list-style-type: none"> • Demonstrate an appreciation for the breadth of software engineering
13212	<p>Software Construction Prerequisite: 13211,11212</p> <p>General principles and techniques for disciplined low-level software design. BNF and basic theory of grammars and parsing. Use of parser generators. Basics of language and protocol design. Formal languages. State-transition and table-based software design. Formal methods for software construction. Techniques for handling concurrency and inter-process communication .</p> <p>Techniques for designing numerical software. Tools for model-driven construction. Introduction to Middleware. Hot-spot analysis and performance tuning .</p> <p>Prerequisite: (SE201 or SE200), CS103 and CS105 .</p> <p><i>Learning objectives:</i> Upon completion of this course, students will have the ability to:</p> <ul style="list-style-type: none"> • Apply a wide variety of software construction techniques and tools, including state-based and table-driven approaches to low-level design of software • Design simple languages and protocols suitable for a variety of applications • Generate code for simple languages and protocols using suitable tools • Create simple formal specifications of low-level software modules, check the validity of these specifications, and generate code from the specifications using appropriate tools • Design simple concurrent software • Analyze software to improve its efficiency, reliability, and maintainability

<p>13326</p>	<p>Software Engineering Approach to Human Computer Interaction Prerequisite:11206</p> <p>Psychological principles of human-computer interaction. Evaluation of user interfaces. Usability engineering. Task analysis, user-centered design, and prototyping. Conceptual models and metaphors. Software design rationale. Design of windows, menus, and commands. Voice and natural language I/O. Response time and feedback. Color, icons, and sound. Internationalization and localization. User interface architectures and APIs. Case studies and project.</p> <p><i>Learning objectives:</i> Upon completion of this course, students will have the ability to:</p> <ul style="list-style-type: none"> • Evaluate software user interfaces using heuristic evaluation and user observation techniques • Conduct simple formal experiments to evaluate usability hypotheses. • Apply user centered design and usability engineering principles as they design a wide variety of software user interfaces
<p>13327</p>	<p>Software Design and Architecture Prerequisites: 13212</p> <p>An in-depth look at software design. Continuation of the study of design patterns, frameworks, and architectures. Survey of current middleware architectures. Design of distributed systems using middleware. Component based design. Measurement theory and appropriate use of metrics in design. Designing for qualities such as performance, safety, security, reusability, reliability, etc. Measuring internal qualities and complexity of software. Evaluation and evolution of designs. Basics of software evolution, reengineering, and reverse engineering.</p> <p><i>Learning objectives:</i></p>

	<p>Upon completion of this course, students will have the ability to:</p> <ul style="list-style-type: none"> • Apply a wide variety of design patterns, frameworks, and architectures in designing a wide variety of software • Design and implement software using several different middleware technologies • Use sound quality metrics as objectives for designs, and then measure and assess designs to ensure the objectives have been met • Modify designs using sound change control approaches • Use reverse engineering techniques to recapture the design of software
<p>13324</p>	<p>Systems Analysis and Design Prerequisites: 13211</p> <p>Fundamental concepts. Notion of a system. Information system. System life cycle. Approaches to system analysis and design (classical, structured and object-oriented). Preliminary and Detailed Analysis. Workflow and Dataflow Diagrams. Structured English. Decision Tables etc. Criteria for software design and evaluation: module coupling, cohesion, modularity, portability. A project is required.</p>
<p>13325</p>	<p>Software Requirements Analysis Prerequisites: 13324</p> <p>Domain engineering. Techniques for discovering and eliciting requirements. Languages and models for representing requirements. Analysis and validation techniques, including need, goal, and use case analysis. Requirements in the context of system engineering. Specifying and measuring external qualities: performance, reliability, availability, safety, security, etc. Specifying and analyzing requirements for various types of systems: embedded systems, consumer systems, web-based systems, business systems, systems for scientists and other engineers. Resolving feature interactions. Requirements documentation standards. Traceability. Human factors. Requirements in the context of</p>

	<p>agile processes. Requirements management: Handling requirements changes.</p> <p><i>Learning objectives:</i></p> <p>Upon completion of this course, students will have the ability to:</p> <ul style="list-style-type: none"> • Discover or elicit requirements using a variety of techniques • Organize and prioritize requirements • Apply analysis techniques such as needs analysis, goal analysis, and use case analysis • Validate requirements according to criteria such as feasibility, clarity, freedom from ambiguity, etc. • Represent functional and non-functional requirements for different types of systems using formal and informal techniques • Specify and measure quality attributes • Negotiate among different stakeholders in order to agree on a set of requirements • Detect and resolve feature interactions
<p>13428</p>	<p>Software Quality Assurance and Testing Prerequisite:13212</p> <p>Quality: how to assure it and verify it, and the need for a culture of quality. Avoidance of errors and other quality problems. Inspections and reviews. Testing, verification and validation techniques. Process assurance vs. Product assurance. Quality process standards. Product and process assurance. Problem analysis and reporting. Statistical approaches to quality control.</p> <p><i>Learning objectives:</i></p> <p>Upon completion of this course, students will have the ability to:</p> <ul style="list-style-type: none"> • Conduct effective and efficient inspections • Design and implement comprehensive test plans • Apply a wide variety of testing techniques in an effective and efficient manner • Compute test coverage and yield, according to a variety of criteria

13391	<p>Practical Training Prerequisite: 90 Cr. Hrs.</p> <p>The student is required to do practical training in a well known software company for a period of (2) months, full-time training, with at least (5) hours per day, or 3 months part-time training with at least (4) hours per day. For the part-time training, the student is allowed to register not more than (6) credit hours in the first or the second semester, or (3) credit hours in the summer semester. The student is required to perform tasks that are related to his major, such as writing, developing, or learning some new software.</p>
13334	<p>Mobile Application Development Prerequisite:11206</p> <p>This course discusses mobile application programming for developing mobile native applications. A specific programming language along with the necessary libraries of reusable software components of a mobile platform will be introduced. This course explores the key theories, principles, concepts, tools, current issues, and best practices of designing and developing applications for mobile devices. Upon course completion, students are expected to produce an innovative mobile solution for a real life situation.</p>
13335	<p>Server Side Programming Prerequisite: 11323,11241,11206</p> <p>This course provides a solid foundation for the concepts of server-side programming, using a current server-side programming/scripting language(s). The course provides a comprehensive coverage of a server-side scripting language, sessions, security, and considerations related to implementing efficient and maintainable server-side applications. The course includes communicating with databases, file servers, and other types of servers.</p>
13434	<p>Secure Software Development Prerequisite: 11323,11241</p> <p>This course covers the security and safety of software development during all stages of a software life cycle including analysis, design, coding, testing, and maintenance. The courses introduces the students to</p>

	the main security principles to be followed in software development. Topics include threat modeling, defensive programming, web security, database security, and other security relevant issues.
13431	<p>Advanced Topics In Programming Prerequisite: Set by Dept.</p> <p>The courses presents advanced software engineering topics using a modern programming language. This include threads, distributed computing, database driven apps. In addition, it includes coverage for design patterns, and implementing different software qualities according to the best practices in industry.</p>
13393	<p>Special Topic in Software Engineering (1) Prerequisite: Set by Dept.</p> <p>The objective of this course is to introduce new topics in Software Engineering to be set by the department.</p>
13413	<p>Formal Methods in Software Engineering Prerequisite: 13324</p> <p>Review of mathematical foundations for formal methods. Formal languages and techniques for specification and design, including specifying syntax using grammars and finite state machines. Analysis and verification of specifications and designs. Use of assertions and proofs. Automated program and design transformation.</p> <p><i>Learning objectives:</i> Upon completion of this course, students will have the ability to:</p> <ul style="list-style-type: none"> • Create mathematically precise specifications and designs using languages such as OCL, Z, etc. • Analyze the properties of formal specifications and designs • Use tools to transform specifications and designs
13424	Software Re-Engineering

	<p>Prerequisite: 13212</p> <p>This course aims to discuss methods applied to the evolution of large industrial software systems. The course introduces the phases of the software life-cycle and then focuses on technical issues related to program understanding, program restructuring, and software migration. More specifically, topics include:</p> <ul style="list-style-type: none"> • The software life-cycle and examination of Software Process Models. The role of software maintenance and evolution in a product's life cycle. Examination of the factors that make software evolution a critical business and technical challenge. • Decision making and economics of software re-engineering. When and how to re-engineer or re-design. • Program analysis and program understanding. Source code representation at higher level of abstraction. Parsers and Abstract Syntax Trees, Control and Data Flow Graphs, Program Dependency Graphs, topics in programming language semantics. Discussion on CASE tools that are available and used for program analysis in industrial environments. • Data flow analysis techniques and applications to slicing, change/impact analysis, re-documentation and, application partitioning. • Software metrics and their use in software evolution (quality, reliability, defect filtering). • Software migration techniques. Techniques to migrate legacy code to new platforms, and new software architectures. Particular emphasis will be given to techniques used to migrate legacy systems to new Object-Oriented architectures. • Evaluation of the re-engineering product. Quality metrics, metrics for progress and productivity.
<p>13427</p>	<p>Software Documentation Prerequisites: 13324</p> <p>Course description: Essentials of oral, written, and graphical communication for software engineers. Principles of</p>

	<p>technical writing; types of documents and strategies for gathering information and writing documents, including presentations. Appropriate use of tables, graphics, and references. How to be convincing and how to express rationale for one's decisions or conclusions. Basics of how to work effectively with others; notion of what motivates people; concepts of group dynamics. Principles of effective oral communication, both at the interpersonal level and when making presentations to groups. Strategies for listening, persuasion, and negotiation.</p>
<p>13432</p>	<p>Software Project Management Prerequisites: 13324</p> <p>Project planning, cost estimation and scheduling. Project management tools. Factors influencing productivity and success. Productivity metrics. Analysis of options and risks. Planning for change. Management of expectations. Release and configuration management. Software process standards and process implementation. Software contracts and intellectual property. Approaches to maintenance and long-term software development. Case studies of real industrial projects.</p> <p><i>Learning objectives:</i></p> <ul style="list-style-type: none"> • Upon completion of this course, students will have the ability to: • Develop a comprehensive project plan for a significant development effort • Apply management techniques to projects that follow agile methodologies, as well as methodologies involve larger-scale iterations or releases • Effectively estimate costs for a project using several different techniques. • Apply function point measurement techniques • Measure project progress, productivity and other aspects of the software process • Apply earned-value analysis techniques • Perform risk management, dynamically adjusting project plans • Use configuration management tools effectively, and apply change management processes properly

	<ul style="list-style-type: none"> • Draft and evaluate basic software licenses, contracts, and intellectual property agreements, while recognizing the necessity of involving legal expertise • Use standards in project management, including ISO 10006 (project management quality) and ISO 12207 (software development process) along with the SEI's CMM model
<p>13433</p>	<p>Risk Management for Software intensive projects Prerequisite: 13324</p> <p>Over half of all software projects fail to meet initial requirements, cost, schedule and/or customer expectations due to problems that occur through the development life cycle. Some of these problems start out as risks that — if identified, analyzed and mitigated — have the potential of never becoming a problem. Despite its benefits, the practice of risk management in industry today is ad-hoc, inconsistent, and often ineffective, leading decision makers to make poor choices based on lack of or mis-communicated information. The goal of this course is to enable the student to become a risk manager on a software project, learning how to reason about risks and the underlying factors that influence them so as to make better project decisions. In the process students will practice existing tools, techniques and strategies that help identify, analyze and assist in mitigating or avoiding risks completely. To provide students with an engaging learning experience, the classes use a hands-on approach to learning about risk management where lectures and discussions are combined with in class exercises, case studies and real application of material taught on industry projects. Students will also be given a chance to conduct research in risk management areas in their chosen areas of interest.</p>
<p>13491</p>	<p>Graduation Project 1 Prerequisite: 90 Cr. Hrs.</p> <p>This course is the first of two courses for the development of significant software system and employing the knowledge gained from courses throughout the program. Includes development of requirements, design, implementation, and quality assurance. Students may follow any suitable process model, must pay attention to quality issues, and must manage</p>

	<p>the project themselves, following all appropriate project management techniques. Success of the project is determined in large part by whether students have adequately solved their customer's problem.</p> <p>This course is focused on the early stages of the software development process such requirements analysis and design.</p>
13493	<p>Graduation Project 2 Prerequisite: 13491</p> <p>This course is the second of two courses for the development of significant software system, employing the knowledge gained from courses throughout the program. Includes development of requirements, design, implementation, and quality assurance. Students may follow any suitable process model, must pay attention to quality issues, and must manage the project themselves, following all appropriate project management techniques. Success of the project is determined in large part by whether students have adequately solved their customer's problem.</p> <p>Prerequisites: Completion of the 100 credit hours.</p> <p><i>Sample deliverables:</i></p> <p>Students should be expected to deliver one or several iterations of a software system, along with all artifacts appropriate to the process model they are using. These would likely include a project plan (perhaps updated regularly, and containing cost estimations, risk analysis, division of the work into tasks, etc.), requirements (including use cases), architectural and design documents, test plans, source code, and installable system.</p>
13494	<p>Special Topic in Software Engineering (2) Prerequisite: Set by Dept.</p> <p>The objective of this course is to introduce advanced topics in Software Engineering to be set by the department.</p>
22342	<p>Computer Organization and Assembly Language New Pre-requisites: 22241 3 credit hours</p>

	<p>Register transfer and micro-operations. Basic computer organization and design. Design of arithmetic logic unit. Design of accumulator. Central processing unit. Hardwired control. Micro programmed control. Execution of instructions. Pipelining. Introduction to memory hierarchy. Microprocessor organization. Central processing unit. Addressing modes. Instruction set. Programming in assembly language. Software interrupts and interfacing with BIOS and DOS. A specific microprocessor will be studied in detail</p>
20132	<p>Calculus (1) Pre-requisites: -</p> <p>Functions, limits and continuity. Derivatives. Differentiation. Inverse functions. Trigonometric functions. Logarithmic and exponential functions. Hyperbolic functions. Integrals.</p>
20133	<p>Calculus (2) Pre-requisites: 31132</p> <p>Methods of integration. Applications of integration. Plane analytic geometry including polar coordinates. Sequences and series, including power series.</p>
20134	<p>Discrete Mathematics Pre-requisites: -</p> <p>Mathematical models. Proof methods. Program correction methods. Sets and operations. Relations and types. Charts and branches. Searching methods. Dividing. Functions and types. Algorithms. Counting methods.</p>
20233	<p>Linear Algebra Pre-requisites: 20133</p> <p>System of Linear Equations: Row-echelon Form, Gaussian Elimination, Gauss-Jordan Method. Matrices: Operations, Properties of Matrix Arithmetic, Matrix Transpose, Special Matrices.</p>

	<p>Determinants: Properties of Determinants, The Method of Cofactors, Adjoint Matrix and Inverse of a Matrix, Cramer's Rule.</p> <p>Euclidean n-space: Vectors, Dot Product, Cross Product, Euclidean n-space, Linear Transformations.</p> <p>Vector Spaces: Vector Spaces, Subspaces, Span, Basis and Dimensions, Fundamental Subspaces, Inner Product Spaces, Orthogonal and Orthonormal Basis, Least Squares, QR-decomposition, Orthogonal Matrices.</p> <p>Eigenvalues and Eigenvectors: Eigenvalues and Eigenvectors, Diagonalization.</p>
20233	<p>Statistical Methods Prerequisites: -</p> <p>Introduction to statistics. Elements of probability. Probability distributions (e.g., binomial, Poisson, geometric, hypergeometric, normal, t, F, and χ^2). Sampling. Simple linear regression. Correlation. Test of hypotheses. Analysis of variance.</p>
20332	<p>Operations Research Pre-requisites: 20133</p> <p>Introductory steps of OR. Linear programming. Graphic solutions. Simplex method. Dual problem. Special linear programming problems (transportation, assignment, and transshipment). Project scheduling (CPM and PERT).</p>
20333	<p>Numerical Analysis Pre-requisites: 20133</p> <p>Vectors and matrices. Determinants. System of linear algebraic equations. Cramer's rule and characteristic-value problem. Error analysis. Iterative methods for solving linear and nonlinear systems of equations. Interpolation and approximation. Introduction to numerical differentiation and integration.</p>
31010	<p>Arabic Language placement test Pre-requisites:-</p> <p>Testing basic grammars in Arabic.</p>

31019	<p>Arabic Language (Remedial) Pre-requisites:-</p> <p>The verb and noun. Grammars. Punctuation. Al Hamza. Applications.</p>
31020	<p>English placement test Pre-requisites:-</p> <p>Testing basic grammars in Arabic</p>
31029	<p>English Language (Remedial) Pre-requisites: -</p> <p>Reading. Writing. Speaking. Listening. Application.</p>
31111	<p>Arabic Language Pre-requisites: 31019</p> <p>Summarization. Punctuation. Spelling. Deletion. Displacement. Construction and inflection. Derivation. Substitution and the vowel system. Number. Indescribability. Sentence. Clause. Rhetoric issues and various applications</p>
31121	<p>English Language Pre-requisites: 31029</p> <p>Advanced reading. Advanced writing. Grammar. Speech and Translation.</p>
31151	<p>Jordan: History and Culture Pre –requisites: -</p> <p>Jordan: the land and the people. Jordan: our homeland. The Arab Nation. History of Jordan. Political system in Jordan. Jordanian Society. Major national institutions. Internal and external challenges facing Jordan. The role of local</p>

	institutions in achieving development in national awareness. The family, childhood and woman and its role in society.
31152	<p>Arabic and Islamic Civilization Pre-requisites: -</p> <p>Concept of Civilization. Stages of Development of Arabic Islamic Civilization. Principles and Grounds of Arabic Islamic Civilization. Areas of cultural creativity in Arabic Islamic Civilization. Linguistics, Theology, Islamic Jurisprudence, Philosophy, Natural and Social Sciences, Islamic Art and Music. Unity of the Arab and Islamic worlds.</p>
31161	<p>Introduction to Library Science Pre-requisites: -</p> <p>Information sources. Types of Cataloging. Types of Catalogs. Types of Classification. Information and Knowledge. Information Society. Information Services. Information Technology. Information Storage. Information Retrieval and Dissemination. Information and Internet.</p>
31171	<p>History of Science Pre-requisites: -</p> <p>Importance of understanding science as a socio – historic phenomenon. Science as industry. Science and development. Cognitive conditions of scientific production. Examples: Kepler and Planck. Science as social production. Socio – historic determinants of the social production of science. Main epochs of the history of natural science. Greek science. Hellenistic science Roman science. Arabic Islamic science. Modern European science. Science in the contemporary Arab world. Epistemological periodization of natural science. Roots of the philosophy of nature. The Ionians. Platonic project in astronomy. Aristotle. Ptolemy. Arabic Islamic astronomy. The 17th century Scientific Revolution.</p>
31211	<p>Arabic literature Pre-requisites: 31111</p>

	Developing students' taste of Arabic literature through the reading of literature essays. Analyzing. Introducing some literature aspects from different eras.
31251	<p>Military Science Pre-requisites: - Grade: Pass / Fail (for Jordanians only)</p> <p>History of the Jordanian Army. Jordanian peace forces. Preparing the Nation for defense and liberation. Genesis and development of the Hashemite Kingdom of Jordan.</p>
31261	<p>Introduction of politics and economy Pre-requisites: -</p> <p>The nature of Political Economy. The Issues of political Economy. The importance of the market. Market effects and political Economy. Three Ideologies of political Economy. The Dynamics of the international political Economy. The political Economy of Structural changes. Long-Term variations of Economic Growth and the effect of political hegemonic. The politics of International Trade. The political Economy of international Finance.</p>
31262	<p>Introduction to Educational Science Pre-requisites: -</p> <p>Education: Principles and Philosophy. Education and individuals. Education and Learning. Education and Society. Education and development. Educational courses and methods. Educational institutions and educational assessments. Education in the Arab and Islamic world.</p>
31263	<p>Technical Writing Pre-requisites: 31111, 31121</p> <p>Organization of the technical report. Layout and organization of the front page. Arrangement of information. Organization and layout of headings and sub-headings. Numbering systems. Order of arrangements of results and</p>

	recommendations. How to prepare an un detailed proposal on any technical and scientific work.
31271	<p>Environmental Science Pre-requisites:-</p> <p>The earth and natural hazards. Ecosystems. Biogeochemical Cycles. Man and the Environment. The Natural Resources in the Solid Earth System. Air Pollution. Water Resources. Management and Pollution. Solid Waste. Food and Health. Environmental Impact Assessment.</p>
31351	<p>Current Issues in the Arab World Pre-requisites: -</p> <p>Social and national fragmentation in the Arab East: Features and Roots. The Arab-Israeli conflict. Democracy and Civil Society in the Arab world. Pan- Arabism and Islam. The self and the other. Secularism. The Arabs and globalization. Human security in the Arab world.</p>
31352	<p>Al-Quds History and Facts Pre-requisites: -</p> <p>The geographic borders of AL- Quds through the history. A glance at the historical discoverers in AL-Quds. AL-Quds and the British occupation. The establishment of Israil The Israili plans to Jadeite AL-Quds. Importance of AL-Quds from the religion point of view. The infringement of AL-Aqsa mosque since 1967 by Israil.</p>
31361	<p>Introduction to Psychology Pre-requisites: -</p> <p>Definition of Philosophy. The relation between Philosophy and Science. The science-based and religion-based thoughts. A historical review of Philosophy and Science. A discussion of the most important fields in Philosophy.</p>
31371	<p>Health education Pre-requisites: -</p>

	<p>Understanding the responsibility we have for our own health. Skills for dealing with emergency cases. Personal safety and accident prevention. Mental health, mental illness, stress and mental health.</p>
--	--