1503741	Software Quality Engineering	---

Advanced Topics in graph algorithms, application-driven treatment of graph theory, evolutionary algorithms, and efficient algorithms with emphasis on algorithms with real world applications. Examples drawn from: computational geometry, biology, scientific computation, image processing, combinatorial optimization, game theory, computer vision and operations research.

1501751	Applied Algorithms	---

Advanced Topics in graph algorithms, application-driven treatment of graph theory, evolutionary algorithms, and efficient algorithms with emphasis on algorithms with real world applications. Examples drawn from: computational geometry, biology, scientific computation, image processing, combinatorial optimization, game theory, computer vision and operations research.

1501730	Cloud Computing	---

The student studies cloud systems as the newest distributed systems and the impact it caused in the world of computing, Its components, characteristics, uses, problems and future.

1501731	Advanced Database systems	---

Overview of database management systems implementation and introduction to emerging database technologies. The topics include: data storage structures, query processing and optimization, database security, transaction management, distributed database systems, Data warehousing and database system architectures.

1503730	Big Data curation	---

The course provides an advanced description of large computer systems, and explains how it can be identified, their design and implementation. Systems Analysis is offered as a

means to gather and organize information so that the required specifications closely correspond to the requirements of users. Then the course provides an explanation of design and converts requirements specification to the form that can be implemented. This is done through the use of tools that are available to manage and develop large projects and data acquisition.

1503735	Advanced Software Modeling	---

Conventional methodologies of software modeling, OO paradigm; OO design methodologies (OMT, Brooch's, Yourdon's methodologies): a comparative study; conversion from imperative to OO designs. Evaluation of OO designs and software metrics. Use of OO methodologies in various application domains such as multimedia, hypermedia, etc. Automation of the methodologies; OO design evaluation, UML. Case Study.

1501732	Advanced Operating Systems	---

This course provides an in-depth examination of the principles of distributed systems in general, and distributed operating systems in particular. Covered topics include processes and threads, concurrent programming, distributed inter-process communication, distributed process scheduling, virtualization, distributed file systems, security in distributed systems, distributed middleware and applications such as the web and peer-to-peer systems.

Concurrency and distributed communication, fault-tolerance, availability, persistence, operating system (OS) structure; principles used in the design of distributed operating systems, case studies, and related research, topics include: compare centralized OS to distributed OS, the state-of-the-art in distributed systems, Identify issues involved in distributed systems, Investigate novel research ideas in distributed systems.

1503745	Component Based Software Design	---

The definition and nature of components, Components and interfaces, Interfaces as contracts and the benefits of components. Basic techniques: Component design and assembly, Relationship with the client-server model and with patterns, Objects and object lifecycle services, Use of object brokers and Marshalling. Applications (including the use of mobile components). Architecture of component-based systems, Component-oriented design. Event handling: detection, notification, and response. Middleware: O.O. paradigm, Object request brokers, Transaction processing monitors, Workflow systems and State-of-the-art tools.

1503751	Software Architecture and Design Patterns	---

Architectural design of complex software systems. commonly-used software system structures، techniques for designing and implementing structures، models and formal notations for characterizing and reasoning about architectures، tools for generating specific instances of an architecture، and case studies of actual system architectures. Architectural design methods for Large-scale software systems، quality design attributes، component-level design، design patterns، aspect-oriented software development، and service oriented architecture.

1503785	Advanced Topics in Software Engineering	---

This course allows the department to cover one of the recent research topics in the field of software engineering. The content of this course may be change depends on the recent topics like cloud computing, data mining, big data analysis.

1501785	Data Mining	---

Methods for identifying valid, novel, useful and understandable patterns in data. The topics include: Data warehousing, induction of predictive models from data (classification, regression

and probability estimation); clustering, association rules and knowledge discovery. Mining sequential patterns, temporal data, spatial data and graph patterns.

1503711	Research Methodology	---

This course aims to provide students with the basic concepts of scientific research: its concept, nature, purpose , scope, , tools, requirements, and statistical methods, as this course will focus on providing students with basic skills in scientific research including the identification of the problem, develop plan, data collection and analysis, writing the final report, selection of the proper approach, apply statistical methods in dealing with the problems, the study of relations, and examine hypotheses.

1503712	Software Verification and Validation	---

This course will address topics in the verification and validation (V&V) of software. Verification addresses issues related to whether the system is correct (with respect to some specification); validation addresses the question whether the right system was built. Topics include the depth study of verification and validation strategies and techniques as they apply to the development of quality, software test planning and management, testing of

software throughout its life cycle. Unit and system level testing. Designing of test cases Black box and white box testing techniques. Measuring test effectiveness. Regressing testing and Integration testing. The relationship of testing to other quality assurance activities as well as the integration of verification and validation into the overall software development process are also discussed.

1503755	Software Maintenance and Evolution	---

The principles of generating maintainable software، Theory and practice of maintaining large scale software، problems in maintaining software systems، building software in view of the maintenance problems، reverse engineering، the principles of software evolution، mainly the evolution of open source projects، program comprehension، software Evolution Process Models.

1503761	Requirements Engineering	---

Requirements methods، tools، and techniques. Software Requirements Fundamentals. Product vs. process، functional vs. non-functional، emergent properties. Requirements Process، process models، process actors، process support and management، and process quality and improvement. Requirements Elicitation. Requirements Analysis، detect and resolve conflicts between requirements، discover the bounds of the software. Requirements Specification. Requirements Validation.

1503731	Software Project Management	---

Skills necessary to lead a project team، understand the relationship of software development to overall product engineering، estimate time and costs، and understand the software process. Advanced topics related to life cycle models، requirements elicitation، configuration، to control environments، quality assurance، and leadership، advanced issues of risk analysis، schedule، costs، team organization، resources، and technical approach، Capability Maturity Model and the technology and practices associated with each and a variety of software standards.

1501755	Practical aspects of Information Security	---

Exploration of modern cryptography techniques (algorithms and protocols) and its application to real-world problems such as: secure and validate electronic documents, messages and e-commerce. The course also illustrates how to design a secure system, controlling access and malware protection such as: computer viruses, worms, spyware, key-loggers, buffer overflow and similar attacks. Web security: Cross-site scripting and SQL injection attacks. Network security: protocols (TCP, DNS, SMTP, SSH, TLS).

1503765	Advanced Formal methods in SE	---

Formal specification languages. Executable and non-executable specifications. Pre and post assertions. Formal verification techniques in the context of software validation and testing. Using a common formal specification language, formulate the specification of a simple software system and demonstrate.

1503791	**Research Project**	---

In this course student make a research in one of software engineering topics.