Descriptions of Courses

CS 200 Introduction to Computer Science 3:1:3(3)

This course provides students with an overall view of computer science. It introduces general concepts and widely used technical terms in operating systems, algorithms, assembly programming, software engineering, data structures, artificial intelligence, etc., in order to help students obtain a comprehensive view of computer science.

CS 202 Problem Solving 2:3:3(15)

This course is about methods for problem solving and algorithm development. Through various lab work, students learn good programming practice in design, coding, debugging, and documentation.

CS 204 Discrete Mathematics 3:0:3(8)

This course covers mathematical concepts that are frequently employed in computer science: sets, relations, propositional logic, predicative logic, graphs, trees, recurrences, recursion, and fundamental notions in abstract algebra such as groups and rings.

CS 206 Data Structure 3:0:3(6)

This course provides students with fundamental concepts in data structures and algorithms in a broad context of solving problems using computers.

CS 211 Digital System and Lab 3:3:4(10)

This course provides students with an understanding of digital systems as building blocks of modern digital computers. This course puts emphasis on providing students with hands-on experience on digital systems. The course includes both lecture and laboratory work on the topics of: boolean algebra, binary system, combinatorial logic, asynchronous sequential circuits, algorithmic state machine, asynchronous sequential circuits, and AHPL. The final oroject is to design a calculator.

CS 220 Programming Principles 3:0:3(6)

This course's goal is to provide students with programming principles and a good feel for the elements of style and the aesthetics of programming, which are necessary in controlling the intellectual complexity of large yet robust software systems. The covered topics include: induction and recursion, data abstraction and representation, values and applicative programming, objects and imperative programming, streams and demand-driven programming, modularity and hierarchy, exceptions and advanced control, and higher-order functions and continuations.

CS 230 System Programming 3:0:3(4)

This course's goal is to provide students with programming techniques necessary in dealing with "systems" development. The covered topics include low-level machine oriented programming, device-control programming, and other various programming techniques for computer operating environment.

CS 300 Algorithms 3:0:3(8)

This course introduces the basic concepts of design and analysis of computer algorithms: the basic principles and techniques of computational complexity (worst-case and average behavior, space usage, and lower bounds on the complexity of a problem), and algorithms for fundamental problems. It also introduces the areas of NP-completeness and parallel algorithms. (Prerequisite: CS204, CS206)

CS 310 Microprocessor and Lab 3:3:4(10)

In this course, students are exposed to the basic hardware architecture of microprocessors and microcomputer

systems. Through lab work, they learn about microprocessor CPUs, memory architectures, and types. They also analyze components of software that run on microcomputer systems. (Prerequisite: CS211)

CS 311 Computer Organization 3:0:3(3)
This course provides students with a basic understanding of computer organization and architecture. It is concerned mostly with the hardware aspects of computer systems: structural organization and hardware design of digital computer systems, underlying design principles and their impact on computer performance, and software impact on computer. (Prerequisite: CS211)

CS 320 Programming Languages 3:0:3(3)
This course provides students with the necessary underlying principles in the design and implementation of programming languages. Lectures use a variety of existing general-purpose programming languages from different programming paradigms: imperative, functional, logical, and object-oriented programming. (Prerequisite: CS220)

CS 322 Formal Languages and Automata 3:0:3(6)
This course covers various types of finite automata, properties of language classes recognizable by automata, context-free grammar, pushdown automata, the Turing machine, and computability. (Prerequisite: CS204)

CS 330 Operating Systems and Lab 3:3:4(12)
In this course, students learn about basic concepts of operating systems, with an emphasis on multi-tasking, and time-sharing. We choose one specific operating system, and study in detail its organization and functions. Students are also required to program a simple operating system, and to develop performance improvement techniques.

CS 350 Introduction to Software Engineering 3:0:3(2)
This course provides students with basic concepts in software engineering in order to develop high-quality software economically. Key concepts are life cycle models, development techniques, automation tools, project management skills, and software metrics.

CS 360 Introduction to Database 3:0:3(8)
This is an introductory-level course to database systems. Students learn about various models, such as E-R models, relational models, and object-oriented models; query languages, such as SQL, relational , and QBE; and file and indexing systems for data storage. Advanced topics, such as data inheritance, database design issues, databased security, and access rights, are also covered.(Prerequisite: CS206)

CS 362 File Structures 2:3:3(3)
The techniques of storing, maintaining, and accessing large databases to disk effectively is becoming very important. The file processing technique is a core technology that can be used for this purpose. This course introduces basic disk I/O concepts, blocking and buffering, and covers various file structures, such as sequential files, hash files, indexed sequential files, and multi-key files. Students are required to implement these file structures in programming languages such as C++ and build a simple DBMS.

CS 370 Symbolic Programming 2:3:3(6)
Students learn LISP and PROLOG, the two most commonly used programming languages in artificial intelligence. The basic programming concepts, grammar, and symbol manipulation are covered in the course. Using intelligent problem solving methods, students build natural language processing systems, database programs, pattern matching programs, learning programs, expert systems, etc.

CS 402 Introduction to Logic for Computer Science 3:0:3(6)
This course is about basics of logic used in computer programming. Topics covered in this course are: propositional calculus, predicate calculus, axiomatic theories, skolemization, unification, and resolution.

CS 408 Computer Science Project 1:6:3
Students learn project management and large-system programming skills that are not usually covered in any single course. Students form teams, and execute one of project ideas suggested by a professor. The scope of the project must cover multiple areas in computer science and be of a magnitude sufficient for a team project.

CS 410 Introduction to VLSI Design 3:0:3(4)
Students learn about very large-scale VLSI chip design using nMOS technology. For basic design methodology, students study stix diagram and layout design; advanced topics like switch and gate logic, PLA's 2-phase clocking, design rules, floor planning, and design technique are covered.

CS 420 Compiler Design 3:0:3(6)
Through this course, students study basic rules and implementation considerations in implementing a programming language. More details on grammar checks for program syntax, implementation optimization, relations between programming languages and compilers, the role of interpreters, run-time systems, and semantically accurate expressions are also covered.

CS 422 Computation Theory 3:0:3(8)
This course deals with computational models, computable and incomputable functions, temporal and spatial complexities, tractable and intractable functions.

CS 440 Data Communication 3:0:3(6)
This course covers basic principles in data communications, such as LAN, WAN, multimedia (e.g., voice and video) transmission. It introduces students to key elements and concepts in network construction. Compared to more CS441, emphasis is placed on lower layer protocols and network topologies.

CS441 Introduction to Computer Network 3:0:3(9)
The goal of this course is to provide undergraduate students with sound understanding of fundamental concepts and problems in networking and to train them in network programming. We will cover topics in the data link, networking, transport, and application layers.

CS 455 Software Project 2:3:3(4)
In this course, students develop programs of practical value, using basic software engineering techniques and software tools. Students are evaluated based on the team effort put into project documentation and development process management. Final deliverables are evaluated based on productivity and reliability.

CS 470 Introduction to Artificial Intelligence 3:0:3(8)
This course introduces basic concepts and design techniques of artificial intelligence, and later deals with knowledge representation and inference techniques. Students are to design, implement, and train knowledge-based systems.

CS 480 Introduction to Computer Graphics 3:3:4(6)
The goal of this course is for students to acquire theory and hands-on experience in computer graphics. Topics covered are: basic functions and principles of input and output devices used in computer graphics,

architectures and features of graphics systems, basic geometric models and their generation algorithms, theories and practice behind 2D and 3D conversion. Basic ideas of hidden line and surface removal and color models are introduced.

CS 489 Computing Ethics and Social Issues 3:0:3(2)
Computers have had a significant impact on our life, more so than any other machine before. In this course, we discuss social problems that computers have caused and ethical issues that challenge technical experts.

CS 490 Research in Computer Science 0:6:3
Students work either alone or as a team to conduct a research project and present its results as a thesis. Students can join an advisor's research projects, or learn practical problem solving techniques, as well as research idea development, project management skills and technical writing.

CS 492 Special Topics in Computer Science 3:0:3(6)
The goal of this course is to expose undergraduate students to recent research problems and results in the selected area of research.

CS 495 Individual Study 0:6:1
This course is to allow a student interested in a specific topic to work with faculty and conduct research in one's area of interest. At the beginning of a semester, a student must discuss a research topic with faculty, and submit a study plan. Any student, no matter what grade one is in, can take this course, and get up to 4 credits.

CS 496 Seminar 0:2:1
Domestic and international researchers are invited to give talks on various topics and future directions in computer science and to get involved in discussion with students.

❒ Graduate Program

CS 500 Design and Analysis of Algorithms 3:0:3(6)
This course presents principles and techniques for design and analysis of computer algorithms, in order for students to gain theoretical and systematic insight into real problems that arise in computer applications. (Students are required to code some existing algorithms aiming at experiencing practical aspects of implementing algorithms on an abstract level.) Students learn to reason about algorithms at an abstract cevel, and experience the practical aspects of implementing an abstract algorithm.

CS 510 Computer Architecture 3:0:3(6)
This goal of this course is to provide the student with an understanding of (*i*) the architectural aspect of the performance issues, and (*ii*) investigation of the full spectrum of design alternatives and their trade-offs.

CS 520 Theory of Programming Languages 3:0:3(6)
This course reviews design principles and implementation techniques of various programming languages. This course also introduces a wide spectrum of programming paradigms such as functional programming, logic programming, and object-oriented programming.

CS 522 Theory of Formal Languages and Automata 3:0:3(6)

This course is intended to understand the current theories of deterministic parsing of context-free grammars. Two basic parsing schemes, LR(k) and LL(k) parsing, are considered and the practical SLR(1) and LALR(1) techniques are discussed. The syntactic error recovery in LR-based parsing is also discussed.

CS 530 Operating System 3:0:3(6)

The main focus of this course is to understand the concurrency features of modern operating systems. Concurrent programming is dealt with in detail to simulate various parts of an OS. Other topics that are required to understand the process-oriented OS structure are also discussed.

CS 540 Network Architecture 3:0:3(9)

The goal of this course is to provide students with an understanding on the following topics. (1) the concept of layered architectures, (2) the design and implementation of communication protocols, (3) the multimedia communication protocol, and (4) the design of high-speed protocols. The course also covers many aspects of protocol engineering: design, implementation and test of communication protocols.

CS 542 Internet Systems Technology 3:0:3(9)

This course reviews the state-of-the-art of today's Internet and Web architectures, describes the challenges facing them, and discusses emerging approaches. In particular, the course covers issues around Internet traffic characterization; Protocols; Web server performance; Mobile Services and Systems, Server clustering; Caching architectures; Peer-to-peer service, Quality of Service (QoS) on the Web; and System support for E-commerce. The goal of the course is to gain understanding of the current research issues and a vision of the next generation Web architecture.

CS 550 Software Engineering 3:0:3(4)

This course covers fundamental concepts required in developing reliable softwares in a cost-effective manner.

CS 560 Database System 3:0:3(6)

This course addresses current technologies of various aspects of database systems. The main objective of this course is to study the design and implementation issues of high performance and high functionality database systems. Through this course, the students will have concrete concepts on database systems and will have in-depth knowledge on most issues of advanced database researches.

CS 562 Database Design 3:0:3(6)

The goal of this course is to establish a consistent framework for database design. Practical database design methodology, major principles, tools and analysis techniques for various phases of database design process are studied.

CS 570 Artificial Intelligence 3:0:3(6)

The goal of this course is to give both a comprehensive introduction to core concepts of AI and hands-on experience in symbolic language programming. This course not only provides a thorough discussion of AI's foundational technologies including predicate calculus, search, and AI languages, but introduces processing, pattern recognition, computer vision, and neural networks.

CS 574 Natural Language Processing I 3:0:3(6)

As a typical application of symbolic AI machine translation(M.T) addresses the major issues involving computational linguistics, rules base, and more fundamentally knowledge representation and inference. In this regard, the goal of the course is to provide students with first-hand experience with a real AI problem. The topics include application of M.T., basic problems in M.T., and classical approaches to the problems.

CS 576 Computer Vision 3:0:3(8)

The goal of this course is to provide students with theory and application of computer vision. Major topics include digital image fundamentals, binary vision, gray-level vision, 3-D vision, motion detection and analysis, computer vision system hardware and architecture, CAD-based vision, knowledge-based vision, neural-network-based vision.

CS 579 Computational Linguistics 3:0:3(6)

This course focuses on universal models for languages, especially English and Korean. For computational study, issues on knowledge representation, generalized explanation on linguistic phenomena are discussed. When these models are applied to natural language processing, properties needed for computational models and their implementation methodologies are studied.

CS 580 Interactive Computer Graphics 2:3:3(10)

This course presents the principles of three-dimensional graphics, including geometric, modeling and realistic image synthesis. It also covers techniques for representing, manipulating, and rendering three-dimensional objects.

CS 600 Graph Theory 3:0:3(6)

This course is intended as a first course in graph theory. Topics include existence theorem, characterization theorem, graph algorithms, and applications of graph theory.

CS 604 Computational Geometry 3:0:3(8)

The goal of this course is to introduce advanced topics on design and analysis of geometric algorithms.

CS 610 Parallel Processing 3:0:3(8)

This course discusses both parallel software and parallel architectures. It starts with an overview of the basic foundations such as hardware technology, applications and, computational models. An overview of parallel software and their limitations is provided. Some existing parallel machines and proposed parallel architectures are also covered.

CS 620 Theory of Compiler Construction 3:0:3(2)

This course's goal is to expose students to some research issues in modern programming language implementation. Topics include conventional data-flow analysis techniques, semantics-based flow analysis, type inference, type-based program analysis, and garbage collection.

CS 642 Distributed Processing Systems 3:0:3(3)

This course covers various high-level issues in distributed processing. We also analyze ISO / OSI upper layers (presentation, application layer protocols) and document switching systems.

CS 644 Advanced Network Architecture 3:0:3(3)

This course serves to provide a more complete understanding of network architecture. In particular, these topics are discussed: internet architecture, architecture components, and architectural implication of new technologies and non-technical issues. The course is composed of lectures, invited presentations and term projects.

CS 650 Advanced Software Engineering 3:0:3(6)

In this course, the fundamental concepts of object-orientation are covered from requirement analysis to implementation with various object-oriented methods including OMT, Booch method, and UML. In addition, several advanced topics in the field of object-orientation are also covered. These advanced topics include parallel and distributed object system, real-time issues, and so on.

CS 655 System Modeling and Analysis 3:0:3(6)
Today's information systems are getting more complex, and need for automation systems is ever increasing. In this couse we address basic modelling methods in system analysis and study static and dynamic analysis of systems using Petri Net.

CS 660 Information Storage and Retrieval 3:0:3(6)
This course covers content analysis and indexing, file organization and record classification for information storage, query formulation, retrieval models, search or selection process, and application systems on question-answering systems, on-line information services, library automation, and other information systems.

CS 662 Distributed Database 3:0:3(6)
The goal of this course is to study the theory, algorithms and methods that underlie distributed database management systems.

CS 664 Advanced Database System 3:0:3(6)
The goal of this course is to study the formal foundation of database systems. The course covers advanced topics such as deductive databases, relational database theory, fixed point theory, stratified negation, closed-world assumption, safety, multivalued dependency, generalized dependency, crash recovery and object-oriented databases.

CS 670 Fuzzy and Intelligent System 3:0:3(6)
The aim of this course is to introduce basic concepts and knowledge of the fuzzy theory and its applications. This course also covers some important intelligent systems including the neural network model and genetic algorithm, and the fusion of the different techniques will be discussed.

CS 674 Natural Language Processing II 3:0:3(6)
The goal of this course is to provide students with current topics in natural language processing (NLP). Students are expected to get acquainted with various leading-edge ideas and techniques in NLP.

CS 676 Pattern Recognition 3:0:3(3)
Through this course, students are expected to acquire general ideas of pattern recognition and its application. Three fields (character, speech and image processing) will be studied in which pattern recognition techniques can be successfully applied.

CS 678 Intelligent Robotics 3:0:3(6)
The goal of this course is to provide students with state-of-the-art technologies in intelligent robotics. Major topics include sensing, path planning, and navigation, as well as artificial intelligence and neural networks for robotics.

CS 682 Digital Storytelling 3:0:3(3)
The need for computational approach to storytelling is growing due to the digitalization of all media types - text, image, and sound. Regardless of media types, story forms the underlying deep structure. This course is concerned with the computational studies on storytelling: building the computational model for storytelling, narrative design, and applications of the computational model to the Web, games, e-books, and animation. Students are expected to build a coherent perspective on designing, implementing, and analyzing digital media.

CS 684 Human-Computer Interaction 3:0:3(3)
This course discusses the basic and core concepts of HCI, namely, the way of interacting with the computerized environment. The course begins with issues related to humans, then broadens the discussion domain to society and culture. Based on basic studies on human, society and culture, we shall discuss the issues in designing, analyzing and implementing various interaction architectures.

CS 700 Topics in Computation Theory 3:0:3(8)
This course is designed to cover recent developments and research results in computation theory.

CS 710 Topics in Computational Architecture 3:0:3(6)
This course is designed to cover recent developments and research results in computational architecture.

CS 712 Topics in Parallel Processing 3:0:3(6)
This course is designed to cover recent developments in the area of parallel processing.

CS 720 Topics in Programming Languages 3:0:3(2)
The goal of this course is to cover recent research trends and results in programming languages.

CS 730 Topics in Operating Systems 3:0:3(6)
This course is designed to cover recent developments in operating systems research.

CS 744 Topics in System Architecture 3:0:3(9)
This course is designed to cover recent developments in system architecture.

CS 750 Topics in Software Engineering 2:3:3(6)
The lectures in this offering cover topics of interest in software engineering.

CS 760 Topics in Database System 3:0:3(6)
This course covers recent issues in database systems.

CS 770 Topics in Computer Vision 3:0:3(8)
This course is designed to cover topics of interest in computer vision.

CS 772 Topics in Natural Language Processing 3:0:3(6)
This course covers topics of interest in natural language processing.

CS 774 Topics in Artificial Intelligence 3:0:3(6)
This course covers advanced topics of interest in artificial intelligence.

CS 776 Topics in Cognitive Science 3:0:3(6)
This course is designed to cover topics of interest in cognitive science.

CS 780 Topics in Interactive Computer Graphics 2:3:3(10)
This course is designed to cover recent developments in interactive computer graphics.

CS 788 Topics on Human-Computer Interaction 3:0:3(6)
This course is designed to cover topics of interest in human-computer interaction.

CS 790 Technical Communication in Computer Science 3:0:3(3)
A technical communication capability is critical for IT professionals. This course includes not only oral

communication but also writing. Primary purpose of this course is to develop technical communication skill with special emphasis on writing. In particular, students are strongly encouraged to practice and develop the skills necessary for writing technical papers. Through active discussions and reviews, students are expected to be able to convey technical ideas in a concise and well-organized manner.

CS 960 M.S. Thesis Research

A student selects an M.S. thesis topic with an advisor, and carries on independent research. The student is required to submit an M.S. thesis as an end product.

CS 966 Seminar (M.S.) 1:0:1

Domestic and international researchers are invited to give talks on various topics and future directions in computer science and to get involved in discussion with students.

CS 980 Ph.D. Disseration Research

A student selects a Ph.D. thesis topic approved by an advisor, and carries on independent research. The student is required to submit a Ph.D. thesis as an end product.

CS 986 Seminar (Ph.D.) 1:0:1

Domestic and international researchers are invited to give talks on various topics and future directions in computer science and to get involved in discussion with students.