Course Description

▣ Undergraduate Program

CS101 Introduction to Programming **2:3:3(5)**
The course teaches the basic technique of computer programming and the basic knowledge in the computer structure, and use of the elective programming language to resolve given problems in structural programming. Based on the elective programming language, it teaches the data structure, input and output, flow control and incidental program, and by using the systematic division of problem solution and concept of module to solve the problems in numerical value field and non-numerical value field with the program experiment.

CS109 Programming Practice **2:3:3**
In this course students who have taken CS101 but who have otherwise little programming experience can develop their programming skills. The course introduces basic concepts of programming and computer science, such as dynamic and static typing, dynamic memory allocation, objects and methods, binary representation of numbers, using an editor and compiler from the command line, running programs with arguments from the commmand line, graphical user interfaces and event-based programming, using libraries, and the use of basic data structures such as arrays, lists, stacks, sets, and maps.

CS202 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.

CS204 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.

CS206 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.

CS211 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, VHDL, CAD tools and FPGAs.

CS220 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.

CS230 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.

CS270 Intelligent robot design and programming 2:3:3
This course aims to provide an opportunity for sophomores to experience creative system design using Lego mindstorm NXT kit and URBI robot software platform. In lectures, robotic CS is introduced and various examples are demonstrated to bring out students' interests. In lab hours, students build own intelligent robot system creatively. Students are educated to integrate hardware and software designs, and make presentations at the end of semester.

CS300 Introduction to 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)*

CS310 Embedded computer systems 3:3:4(10)
Embedded systems are found everywhere. The goal of this course is to develop a comprehensive understanding of the technologies behind the embedded computer systems, including hardware and software components. Students will gain hands-on experience in designing a embedded system using CAD tools and FPGAs. *(Prerequisite: CS211)*

CS311 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)*

CS320 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: CS206)*

CS322 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)*

CS330 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.

CS341 Introduction to Computer Networks 3:3:4
The goal of this course is to provide students with sound understanding of fundamental concepts and problems in networking and to train them in network programming. We begin with an introduction to key applications in today's Internet and then cover the reliable transfer protocol, TCP, and its congestion control; and the IP layer that covers the diversity in physical layer technologies and provides an end-to-end abstraction. Finally, we include key concepts in multimedia networking and security in communication networks. *(Prerequisite: CS230)*

CS348 Introduction to Information Security **3:0:3**

This course covers the overall contents of information security. Students will be exposed to fundamental concepts in information security including cryptography, system security, software security, web security and network security. This course introduces how security attacks occur in the modern computing environments. Students will also have opportunities to understand techniques to discover and disable such security attacks.

CS350 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.

CS360 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 calculus, and QBE; and file and indexing systems for data storage. Advanced topics, such as data inheritance, database design issues using functional and multivalued dependencies, database security, and access rights, are also covered. *(Prerequisite: CS206)*

CS361 Introduction to Data Science **3:0:3**

Data science is an inter-disciplinary field focused on extracting knowledge from typically large data sets. This course aims at teaching basic skills in data science for undergraduate students. It covers basic probability and statistics theories required for data science; exploratory data analysis (EDA) required for understanding a given data set; and predictive analysis based on statistical or machine learning techniques. Additionally, it discusses recent big data processing techniques and various data science applications. The students will learn how to implement the methodologies using the Python language.

CS370 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.

CS371 Introduction to Deep Learning **3:0:3**

This is an undergraduate-level introductory course for deep learning. There have been enormous advances in the field of artificial intelligence over the past few decades, especially based on deep learning. However, it is not easy to see what frontiers the current deep learning is facing and what underlying methods are used to enable these advances. This course aims to provide an overview of traditional/emerging topics and applications in deep learning, and basic skill sets to understand/implement some of the latest algorithms.

CS372 Natural Language Processing with Python **3:0:3**

The course offers students a practical introduction to natural language processing with the Python programming language, helping the students to learn by example, write real programs, and grasp the value of being able to test an idea through implementation, with an extensive collection of linguistic algorithms and data structures in robust language processing software.

CS374 Introduction toHuman-Computer Interaction **3:0:3**

This course introduces fundamental concepts, theories, and methods for designing, prototyping, implementing, and evaluating user interfaces. Students apply these lessons to a practical problem in

a team project, which follows a user-centered design process.

CS376 Machine Learning **3:0:3**

Machine learning, a sub-field of computer science, has been popular with the era of intelligent softwares and attracted huge attention from computer vision, natural language processing, healthcare and finance communities to name a few. In this introductory course, we will cover various basic topics in the area including some recent supervised and unsupervised learning algorithms.

CS380 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.

CS402 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.

CS406 Mathematics for Computer Science **3:0:3**

The main interest for computer scientist is how to compute a solution of the given problem with limited computation resources. This constraint leads to a unique set of mathematical tools for computer scientists. Hence I would like to introduce a class which conveys mathematical tools and their underlying concepts suitable for computer scientists in general.

CS408 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.

CS409 Software Projects for Industrial Collaboration **1:6:3**

This course aims to help students internalize project-based competencies that are essentially needed in the software industries. First of all, they get to figure out the fundamentals and philosophies of software engineering through panel discussions with the reading list. Also, they are asked to be organized into teams with mentors from the industry companies, and to conduct their own software project based on the infrastructures and tools that are really used in the field, minimizing the gap between academia and practitioners.

CS411 System for Artificial Intelligence **3:0:3**

Tremendous success of Artificial Intelligence (AI) can be attributed to two primary reasons: (1) significant advances in ML algorithms with great emphasis on Deep Learning, and (2) high-performance computing mainly fueled by hardware accelerators such as GPU and specialized software systems. This course focuses on the second reason and look at AI in the system perspective. This course will look into the entire computing stack built solely for AI, particularly Machine Learning and Deep Learning, This stack constitutes domain-specific programming interface and platforms (e.g., Tensorflow), DNN compilers (e.g., TVM), and hardware accelerators (e.g., GPU and TPU).

CS420 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.

CS422 Computation Theory **3:0:3(8)**
This course deals with models of computation, computable and incomputable functions, temporal and spatial complexities, tractable and intractable functions.

CS440 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.

CS442 Mobile Computing and Applications **3:0:3**
We cover fundamental concepts and problems in mobile and wireless networking and teach system design and implementation in mobile computing. Topics we cover are: introduction to data communications, CDMA, WiFi, and WiBro/WiMAX. Issues related to mobile computing platforms as well as systems comprising sensor networks are also covered. The term project involves application design and development for mobile computing.

CS443 Distributed Algorithms and Systems **3:0:3**
The goal of this course is to provide students with theoretical basis of distributed system design and hands-on experience with distributed systems. The course will start with introduction to functional programming, and then proceed to the MapReduce-like cloud computing framework. Then we expose students to distributed algorithms. Students learn how to program massively parallel jobs in a cloud computing environment and build theoretical underpinnings to expand MapReduce experience to a greater diversity of cloud computing applications. *(Prerequisite: CS330, CS341)*

CS447 Web Security Attack Laboratory **2:3:3**
The course introduces web attacks that trigger various vulnerabilities in web services. It covers SQL injection, cross-site scripting, and cross-site request forgery, which constitute core web attacks, as well as same-origin policy. The course also provides a lab session for each week, which helps students practice actual attacks in a simulated web environment. The goal of the course is to let students learn and understand various web threats via conducting the covered attacks by themselves.

CS453 Automated Software Testing **3:0:3(6)**
This class covers basics of automated software testing techniques with regard to practical applications. These automated testing techniques can provide high reliability for complex embedded software compared to traditional testing methods in a more productive way. This class utilizes various automated software testing tools and learn about their underlying mechanisms for maximal benefit.

CS454 Artificial Intelligence Based Software Engineering **3:0:3**
This course aims to introduce the operations and applications of metaheuristic and bio-inspired algorithms, including genetic algorithm, swarm optimization, and artificial immune system. By considering diverse problems ranging from combinatorial ones to performance improvement of complex software system, students are expected to learn how to apply computational intelligence to unseen problems.

CS457 Requirements Engineering for Smart Environments **3:0:3**
This course is designed to learn requirements engineering concepts and techniques for developing software systems in modern smart computing environments such as the World Wide Web, Internet of Things (IoT), and mobile computing environments. In this course, students learn the core

concepts and techniques of software requirements engineering, the key characteristics of the Web, IoT and mobile computing environments, and practical methods to elicit, model, analyze and manage requirements for developing software systems in the modern computing environments.

CS458 Dynamic Analysis of Software Source Code 3:0:3

This class teaches automated SW testing techniques that analyze target source code to automatically generate various test inputs which explore diverse behaviors of a target program. This class guides students to use various open-source software testing tools and learn the underlying mechanisms of the tools to maximize the performance of automated testing.

CS459 Introduction to Services Computing 3:0:3(6)

This course is designed to learn technologies and strategies for modeling and building service oriented architecture and service applications in various computing environments such as Internet of Things, mobile computing and cloud computing environments to integrate various computing resources and capabilities in users' point of views.

CS470 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.

CS471 Graph Machine Learning and Mining 3:0:3

Graphs are fundamental tools for modeling relationships between objects, enabling us to model diverse real-world problems and data. Graph machine learning and graph mining techniques are utilized in many modern AI and big data analytics domains. This course introduces various graph-based machine learning and mining techniques, including graph neural networks (applying deep learning ideas to graphs), knowledge graphs (representing human knowledge as graphs), graph representation learning (converting graphs into feature vectors), random walks and centrality measures on graphs, graph clustering, and graph anomaly detection. Also, this course introduces how these techniques are applied in information retrieval, natural language understanding, computer vision & graphics, robotics, and bioinformatics.

CS473 Introduction to Social Computing 3:0:3

Computing today no longer only concerns a single user in front of their computer. An increasing number of modern systems are inherently social, involving a large group of users to collaborate, discuss, ideate, solve problems, and make decisions together via social interaction. This course aims to introduce major concepts, real-world examples, design issues, and computational techniques in social computing. Students apply the lessons to a practical problem via a semester-long team project.

CS474 Text Mining 3:0:3

This course will introduce the essential techniques of text mining, understand as the process of deriving high-quality information from unstructured text. The techniques include: the process of analyzing and structuring the input text with natural language processing, deriving patterns with machine learning, and evaluating and interpreting the output. The course will cover some typical text mining tasks such as text categorization, text clustering, document summarization, and relation discovery between entities.

CS475 Machine Learning for Natural Language Processing 3:0:3

This course will cover important problems and concepts in natural language processing and the machine learning models used in those problems. Students will learn the theory and practice of ML

methods for NLP, read and conduct research based on latest research publications.

CS477 Introduction to Intelligent Robotics **3:0:3**

This course will introduce students to the basics of embodied intelligence called intelligent robotics. The course aims to study the fundamental concepts in intelligent robotic system that can sense, plan, and act in the world. To do that, we will discuss 1) the basic concepts, such as control, kinematics, in traditional robotics and 2) state-of-the-art technologies, such as task-and-motion planning and machine learning theories, toward intelligent robotic system. The course will include a brief review of basic tools, such as Robot Operating System (ROS), and also overview contemporary techniques. It will also include individual exercise and final (individual/team) projects.

CS479 Machine Learning for 3D Data **3:0:3**

3D Data are widely used in many applications in computer vision, computer graphics, and robotic. In this course, we will cover the recent advances in machine learning techniques for processing and analyzing 3D data and discuss the remaining challenges. Most of the course material will be less-than 5-year-old research papers in several sub-fields including Computer Vision, Computer Graphics, and Machine Learning. The course will be project-oriented (no exam, no paper-and-pencil homework, but easy programming assignments) and consist of seminar-style reading group presentations.

CS481 Data Visualization **3:0:3**

Data visualization techniques help data scientists to interact with data to extract insightful information, examine hypotheses, and perform data storytelling for decision making. This course covers the fundamental concepts of data visualization, such as design principles, representation, perception, color, and data storytelling. Besides, it will provide in-depth tutorials and practices on the entire visualization process (i.e., ideation, prototyping, and usability testing) by building a web-based interactive service with Python and JavaScript. The course will be delivered in an active learning format such that concept learning is followed by in-class activities and programming practices. Furthermore, there will be programming sessions (e.g., Web programming, Python data processing, and visualization) and design studio sessions (e.g., design process and peer feedback). A final project on building real-world visual analytics solutions will help students to use the techniques learned in the class (e.g., exploring a mobile and wearable sensor dataset on the web).

CS482 Interactive Computer Graphics **3:2:3**

With advances in computing environment, we can get high quality rendering of 3D virtual world in realtime. This course is designed for understanding practical algorithms for realizing 3D computer graphics and visualization essential for not only computer animation but also in various interactive applications including computer games, simulation, and virtual reality. This is a projects-oriented class that will introduce the concepts of interactive computer graphics. Students are expected to work on a team to develop their own project.

CS483 Geometric Modeling and Processing **3:0:3**

3D content creation is a crucial part of many industries such as graphics, AR/VR, CAD/CAM, and digital fabrication, which tasks typically include designing and creating virtual objects/scenes or reconstructing a real environment. Processing scanned 3D data is also an important problem in many applications as 3D scanning technology is being widely applied, for example, in autonomous driving, robot navigation, and 3D object replication. In this course, we discuss fundamental mathematical methods for geometric 3D modeling and geometric data processing, which can be used (not only in graphics-related fields but) in many other areas in science and engineering.

CS484 Introduction to Computer Vision **3:0:3**

In this course, students will learn the basic principles and techniques of image processing. Expanding the foundations of image processing, they will learn 3-dimensional image processing from camera images and also techniques for deep learning-based image understanding, combined with artificial intelligence. To this end, the curriculum of this course consists of three parts: (1) the basic principles and understanding of image processing, (2) the basic principles and understanding of 3D image processing, and (3) the basic principles and understanding of image processing using artificial intelligence. Students learn and experience basic principles for computer vision and various image processing applications based on the deep understanding of computer vision.

CS485 Machine Learning for Computer Vision **3:0:3**

The course studies concepts, theories and state-of-the-art methods for visual learning and recognition. This module is unique focusing on a broader set of machine learning, for computer vision, in an optimisation perspective.

CS486 Wearable User Interface **3:0:3**

As computer forms and utilization environments become diverse, various user interfaces are evolving beyond the traditional GUI. Especially with the advancement of AR/VR platforms, the importance of wearable user interfaces is increasing. This course aims to understand various genres of wearable user interfaces, major prototyping techniques for researching them, and multi-modal channels for proposing new wearable interfaces.

CS489 Computer Ethics & 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.

CS490 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.

CS492 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.

CS493 Special Topics in Computer Science Ⅰ **1:0:1**

CS494 Special Topics in Computer Science Ⅱ **2:0:2**

CS495 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.

CS496 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.

▣ Graduation Program

CS500 Design and Analysis of Algorithm **3:0:3(6)**
This course introduces basic techniques for the design and analysis of computer algorithms, such as divide-and-conquer, the greedy method, and dynamic programming. Students learn to reason algorithmically about problems arising in computer applications, and experience the practical aspects of implementing an abstract algorithm.

CS504 Computational Geometry **3:0:3(8)**
Computational geometry studies algorithms and data structures for processing and storing geometric objects. This courses discusses algorithm design techniques such as plane sweep and geometric divide & conquer; data structures such as point location structures, interval trees, segment trees, and BSP trees; and geometric structures such as arrangements, triangulations, Voronoi diagrams, and Delaunay triangulations.

CS510 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.

CS520 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.

CS522 Theory of Formal Language 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.

CS524 Program Analysis **3:0:3**
This course introduces a technique called program analysis that estimates the behavior of programs before running them. Instead of running programs with infinite inputs, program analysis statically estimates runtime behaviors of programs within a finite time. The course will cover fundamental theories, designs and implementations of program analysis including semantic formalism and the theory of abstract interpretation.

CS530 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.

CS540 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.

CS541 Smart Business Application and Development **3:0:3(5)**
The course is intended for graduate students to understand and develop smart business application running on smart phones. It provides a comprehensive guide covering programming technology on Mobile Internet, Mobile Security and Payment, Location based and Context Aware Services, Social

Network Services, and Business Model Development Method through Case Study, Value Chain Analysis and Economic Feasibility Study. An application is proposed and developed by students as team consisting of business and engineering areas for the purpose of creating new application services and businesses.

CS542 Internet System Technology **3:0:3(9)**

This course reviews the state-of-the-art of today's Internet system as well as service architectures, describes the challenges facing them, and discusses emerging approaches. In particular, the course covers issues around Internet traffic characterization; protocols; server architectures and performance; mobile and pervasive services and systems, virtualization; content distribution; peer-to-peer architecture, quality of services (QoS); and architectural alternatives for applications and services. The goal of the course is to gain understanding of the current research issues and a vision of the next generation Internet system and service architecture.

CS543 Distributed Systems **3:0:3(3)**

This course provides theoretical knowledge and hands-on experience with distributed systems' design and implementation. The course will focus on the principles underlying modern distributed systems such as networking, naming, security, distributed sychronization, concurrency, fault tolerance, etc. along with case studies. Emphasis will be on evaluating and critiquing approaches and ideas. (*Prerequisite: CS510, CS530)*

CS546 Wireless Mobile Internet **3:0:3(5)**

This course is intended for graduate students who want to understand Wireless Mobile Internet. It provides a comprehensive technical guide covering introductory concepts, fundamental techniques, recent advances and open issues in ad hoc networks and wireless mesh networks. The course consists of lectures, exams and term project.

CS548 Advanced Information Security **3:0:3(6)**

The main objective of this course is to provide students with comprehensive knowledge of information security. The course helps students to build profound understanding of information security by teaching the fundamentals of information security, which include, but are not limited to: cipher, access control, protocol, and software engineering. The primary fous of the course is on the general concept of information security.

CS548 Advanced Information Security **3:0:3(6)**

The main objective of this course is to provide students with comprehensive knowledge of information security. The course helps students to build profound understanding of information security by teaching the fundamentals of information security, which include, but are not limited to: cipher, access control, protocol, and software engineering. The primary fous of the course is on the general concept of information security.

CS550 Software Engineering **3:0:3(4)**

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

CS552 Models of Software Systems **3:0:3(10)**

For long time, computer scientists have investigated the problem of automating software development from a specification to its program. So far the efforts were not fully successful but much of the results can be fruitfully applied to development of small programs and critical small portions of large programs. In this course, we study the important results of such efforts and, for that, we learn how to model software systems with formal description techniques, how to model software systems such that the various properties expected of the software systems are verifiable and how to verify various properties of software systems though the models.

CS554 Designs for Software and Systems **2:3:3(4)**

Development of software and systems requires to understand engineering design paradigms and methods for bridging the gap between a problem to be solved and a working system. This course teaches how to understand problems and to design, architect, and evaluate software solutions.

CS560 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.

CS562 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.

CS564 Data Science Methodology **3:0:3(6)**

The ability to handle big data and statistically analyse them is crucial for data scientists. This course covers social data basics and tools to handle, analyze, and visualize such data via utilizing key analysis packages in R.

CS565 IoT Data Science **3:0:3**

The goal of this course is to learn the basics of how to use sensor data for designing intelligent IoT services. The course covers the entire process of IoT data science for ubiquitous computing: i.e., data collection, pre-processing, feature extraction, and machine learning modeling. Mobile, wearable, and smart sensors will be used, and the types of sensor data covered include motion (e.g., vibration/acceleration, GPS), physiological signals (e.g., heart rate, skin temperature), and interaction data (e.g., app usage). Students will learn the basic digital signal processing and feature extraction techniques. Basic machine learning techniques (e.g., clustering, supervised learning, time-series learning, and deep learning) will be reviewed, and students will master these techniques with in-class practices with Google Co-lab and IoT devices. A final mini-project will help students to apply the techniques learned in the class to solve real-world IoT data science problems.

CS570 Artificial Intelligence and Machine Learning **3:0:3(6)**

Classical artificial intelligence algorithms and introduction to machine learning based on probability and statistics.

CS572 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.

CS574 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.

CS575 AI Ethics **3:0:3**

Recent progress in AI technologies and research have raised concerns about data privacy and

protection, misuse of AI to harm people and society, bias in data and trained models, and AI divide that benefits the rich people and nations more than the poor. It is thus very important to learn about the ethical issues of AI including bias, fairness, privacy, trust, interpretability, and societal impact.

CS576 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.

CS577 Robot Learning and Interaction **3:0:3**
This course will introduce graduate students to the emerging area of robot learning and interaction toward human-centered robotics. The course overviews each robotic learning and interaction areas including learning from demonstration (LfD), (inverse) reinforcement learning (RL), natural language interaction, interactive perception, etc. We will then review the state-of-the-art technologies and exercise a part of technologies using simulated robotic manipulators via Robot Operating System (ROS). Finally, we will exercise the learned techniques via final individual/team projects.

CS578 Bionic Human-Robot Interaction **3:0:3**
We aim to study neural signal modellings through the integration of AI, control theory, neuroscience, biomechanics and robot design, and go over technologies of the human-robot interaction by using neural signals in the aspect of both software and hardware engineering. Discussion on the current and future trends and search about interdisciplinary approaches are planned. Various application examples will be demonstrated to promote students' understanding.

CS579 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.

CS580 Computer Graphics **3:1:3(5)**
We will study fundamentals of computer graphics and their applications to games, movies, and other related areas. In particular, we will study different branches, fundamentals, rendering, animation, and modeling, of computer graphics. Also, CS580 can be taken by students who have not taken any computer graphics related courses in their undergraduate courses.

CS584 Human-Computer Interaction **3:0:3**
The goals of the course are 1) introducing the scientific foundations and methods for empirical HCI research, 2) introducing the wide range of advanced user interfaces, and 3) providing an experience of realizing and evaluating novel user interface ideas.

CS590 Semantic Web **3:0:3(6)**
"Semantic Web" allows machines to process and integrate Web resources intelligently. Beyond enabling quick and accurate web search, this technology may also allow the development of intelligent internet agents and facilitate communication between a multitude of heterogeneous web-accessible devices.

CS591 Software Ecosystem **3:0:3(5)**
As the importance of software in the overall industrial economy grows, and as the software industry undergoes important transformations, this course reviews software technology and the issues that surround its dissemination and use from a number of relevant perspectives. This includes the

perpectives from the user, the creator, manager, software supply industry, software creation industry, government.

CS592 Special Topics in Computing **3:0:3**
This free topic course for graduate students covers ever evolving computer science topics to expose the latest trends in computer science to students. In addition, the course provides a validating platform to find a new topic intended for future regular courses.

CS600 Graph Theory **3:0:3(6)**
This course is intended as a first course in graph theory. It covers the basic theory and applications of trees, networks, Euler graphs, Hamiltonian graphs, matchings, colorings, planar graphs, and network flow.

CS610 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.

CS612 Social network-aware ubiquitous computing **3:0:3**
This course is intended for graduate students. This course introduces the fundamentals of social network aware ubiquitous computing. The first half of the course focuses on the main components of ubiquitous computing and social networking. The core concepts of social network aware ubiquitous computing will be explained by analysis of and discussion on existing approaches. Students will be asked to participate in prototyping of a social network aware ubiquitous computing application and/or system.

CS620 Advanced 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.

CS632 Embedded Operating Systems **3:0:3(6)**
The goal of this course is to provide in-depth design concepts and implementation skills required for designing and developing embedded operating systems. Topics covered include boot loader, process management, memory management, I/O device management, and file systems in embedded operating systems.

CS634 Real-Time Systems **3:0:3**
This course aims to provide 1) broad understanding on real-time systems, 2) in-depth knowledge on real-time scheduling theories, and 3) hands-on experience on real-time operating systems. In particular, it will deal with real-time issues on smartphone operating systems.

CS636 UX-oriented Platform Design Studio I **0:9:3**
This course provides a studio-oriented eduction for designing and prototyping UX-oriented SW platforms. Based on user study and creative concept development method, students will learn to extract system requirements, design a platform, and implement the proposed system. This course will emphasize design and implementation aspects for user-oriented SW systems, in addition to basic theoretical aspects for creative concept.

CS644 Ubiquitous Networking **3:0:3(6)**
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.

CS646 Digital Contents Security **3:0:3**
In this course, the technology related with the contents security is studied. Various security issues of the multimedia including image, video and audio are covered.

CS650 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.

CS652 Software & Systems Product Line Engineering **3:0:3(6)**
In contrast that traditional software engineering has been focussed on single systems, software & systems product line (SSPL) is applicable to family of software systems and embedded systems. Students will understand the SSPL paradigms and will learn how to realize & evaluate SSPL. The key knowledge areas in this course include reference model, scoping, commonality, variability, domain and application engineering.

CS654 Software Process **3:0:3(6)**
Software process is an important leverage point from which to address software quality and productivity issues. Students will learn theoretical foundations on software process, the methods of defining process, and how to apply the process concepts to improve software quality and productivity.

CS655 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 course we address basic modelling methods in system analysis and study static and dynamic analysis of systems using Petri Nets.

CS656 Software Engineering Economics **3:0:3**
The primary objectives of this course are to enable the students to understand the fundamental principles underlying software management and economics; to analyze management situations via case studies; to analyze software cost/schedule tradeoff issues via software cost estimation tools and microeconomic techniques; and to apply the principles and techniques to practical situations.

CS660 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.

CS662 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.

CS664 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, multivalved dependency, generalized dependency and crash recovery.

CS665 Advanced Data Mining **3:0:3**
Mining big data helps us find useful patterns and anomalies which lead to high impact applications

including fraud detection, recommendation system, cyber security, etc. This course covers advanced algorithms for mining big data.

CS670 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.

CS671 Advanced Machine Learning **3:0:3(6)**
This course will cover advanced and state-of-the-art machine learning such as graphical models, Bayesian inference, and nonparametric models.

CS672 Reinforcement Learning **3:0:3(2)**
This course covers reinforcement learning, which is one of the core research areas in machine learning and artificial intelligence. Reinforcement learning has various applications, such as robot navigation/control, intelligent user interfaces, and network routing. Students will be able to understand the fundamental concepts, and capture the recent research trends.

CS674 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.

CS676 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.

CS680 Advanced Computer Graphics **3:0:3(6)**
In this class we will discuss various advanced computer graphics, virtual reality, and interaction techniques. More specifically, we will look into rendering, visibility culling, multi-resolution, cache-coherent methods, and data compression techniques for rasterization, global illumination and collision detection.

CS681 Computational Imaging **3:0:3**
This course provides an introduction to color in computer graphics, with an in-depth look at two fundamental topics: digital color imaging techniques and numerical visual perception models. Students will work on an individual project on color of their choice.

CS682 Digital Storytelling **3:0:3(3)**
The need for a computational approach to storytelling is growing due to the digitalization of all media types - text, image, and sound. Regardless of media types, the story forms the underlying deep structure. This course is concerned with computational aspects of storytelling: building a 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.

CS686 Motion Planning and Applications **3:0:3**
In this class we will discuss various techniques of motion and path planning for various robots. We go over various classic techniques such as visibility graphs and cell decomposition. In particular, we will study probabilistic techniques that have been used for a wide variety of robots and extensively investigated in recent years.

CS688 Large-Scale Image & Video Retrieval **3:0:3**

In this class we will discuss various techniques related to image/video retrieval. Especially, we will go over image/video features (e.g., SIFTs and GISTs), their indexing data structures, and runtime query algorithms. We will also study scalable techniques that can handle large-scale image/video databases, in addition to looking into novel applications of them.

CS700 Special Topics in Computation Theory 3:0:3(8)
Students study recent papers or books in the area of Theory of Computation.

CS710 Special Topics in Computational Architecture 3:0:3(6)
This course covers recently developed, new computer architectures. Students study and analyze new computational models, high-level languages, computer architectures etc.

CS712 Topics in Parallel Processing 3:0:3(6)
In this course, students study parallel processing architectures, algorithms, and languages, especially their use in 5th generation computers. The course is based on recent papers, and can be seen as a continuation of Parallel Processing (CS610).

CS720 Special Topics in Programming Languages 3:0:3(2)
This course covers recent research topics related to programming languages, such as theory, new paradigms, programming language design & implementation etc.

CS730 Special Topics in Operating Systems 3:0:3(6)
The goal of this course is to develop abilities related to role and performance of operating systems. Students study and debate topics such as designing and implementing a new operating systems for a new environment, utilizing an existing operating systems effectively, OS architecture, ways of evaluating OS performance, file systems, threads, parallel operating systems, etc.

CS744 Special Topics in System Architecture 3:0:3(9)
In this course, students learn about the structure of computer systems through individual projects and experiments related to user interfaces and object-oriented architectures.

CS748 Special Topics in Information Security 3:0:3(3)
The goal of this course is to discuss with the research trends and hot issues on information security and suggest the best security practices on new emerging IT services or systems as the security expertise.

CS750 Special Topics in Software Engineering 2:3:3(6)
Students study advanced topics in software engineering, such as formal specification, reuse, software development environments, theory of testing, proving program correctness, etc.

CS760 Special Topics in Database System 3:0:3(6)
In this course, students study and discuss recent developments and topics in database systems.

CS770 Special Topics in Computer Vision 3:0:3(8)
This course consists of lectures about major topics related to computer vision, seminars, and projects. Recent major topics are motion detection and analysis, parallel computer vision systems, CAD-based 3-D vision, knowledge-based vision, neural network-based vision, etc.

CS772 Special Topics in Natural Language Processing 3:0:3(6)
This course covers the theory of natural language processing and recent developments in practice. Students study the theory of language, parsing, situational semantics, belief models etc. They practice by designing and developing utilities and systems.

CS774 Special Topics in Artificial Intelligence 3:0:3(6)

The goal of this course is to provide students with recent theory of AI and its application. It covers information representation. heuristic search, logic and logic language, robot planning, AI languages, expert system, distributed AI system, uncertainty problem and so on.

CS776 Special Topics in Cognitive Science 3:0:3(6)
This course defines humans' cognitive ability, and then studies a variety of methodologies by which cognitive psychology, artificial intelligence, computer science, linguistics, and philosophy apply this ability to machines. This course focuses on 'neural networks' as a computational model of the brain and as a method for approaching fields that computers cannot solve efficiently, such as pattern recognition, voice recognition and natural language processing.

CS780 Special Topics in Interactive Computer Graphics 2:3:3(10)
This course covers advanced topics of computer graphics such as modeling geometric objects, rendering and processing three-dimensional objects, and manipulating motion. The course surveys and analyzes recent results, and discusses the research focus for the future.

CS788 Special Topics in Human-Computer Interaction 3:0:3(6)
This course focuses on technical problems in the interaction between humans and computers. Human-Computer interaction (HCI) is related to somatology, sociology, psychology as well as software and hardware. Through this course, students survey and analyze recent research tendencies, and discuss the future developments.

CS790 Technical Writing for Computer Science 2:3:3(6)
The ability to communicate about technical matters is critical for IT professionals. The purpose of this course is to develop the student's technical communication skills, primarily in writing, but also in oral communication. Students practice the skills necessary for writing technical papers. Through active discussions and reviews, students work on their ability to convey technical ideas in a concise and well-organized manner.

CS891 Special Topics in Computer Science Ⅰ 1:0:1

CS892 Special Topics in Computer Science Ⅱ 2:0:2

CS893 Special Topics in Computer Science Ⅲ 3:0:3

CS960 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.

CS965 Individual Study in M.S.
Students are given an individual project in which they can bring the material learnt in other courses into practice. They experience the experimental side of problems, and improve their experimental and analytic abilities. Before registering for this course, students must discuss their project with the responsible professor.

CS966 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.

CS980 Ph.D. Dissertation 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.

CS986 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.