Overview of Informatics Curriculum for Beginners at Universities (Plan)

Competency	Areas	Knowledge Area of Computer Science Curricula 2013 (CS2013)	Required Class Hours	Optional Class Hours
Technical understanding of computer science / Familiarity with common themes and principles / Appreciation of the interplay between theory and practice / System-level perspective / Problem solving skills / Commitment to life-long learning / Commitment to professional responsibility / Awareness of the broad applicability of computing / Appreciation of domain-specific knowledge	Architecture and Organization	AR, CN, GV	2	2
	Discrete Structures	DS	1	7
	Systems Fundamentals	SF, NC, OS, PL, IM	3	67
	Algorithms and Programming	AL, SDF	8	46
	Platform-based Development	PBD	-	9
	Graphics and Visualization	GV, CN	-	4
	Data Science and Artificial Intelligence	IM, IS	5	78
	Human-Computer Interaction	HCI	-	6
	Social Issues and Professional Practice	SP, IAS	7	23
Project experience	Software Engineering	SE	2	28
Communication and organizational skills	Social Issues and Professional Practice	SP	2	-
			30	270

*Required class hours: The minimum class hours for a required topic are calculated and considered as 28 class hours for 1st semester (2 hours for 1st semester × 14 weeks).
*Optional class hours: Minimum class hours for selectively handling advanced knowledge after teaching Required Topics

*CS2013 Knowledge Area: Algorithms and Complexity(AL), Architecture and Organization(AR), Computational Science(CN), Discrete Structures(DS), Graphics and Visualization(GV), Human-Computer Interaction(HCI), Information Assurance and Security(IAS), Information Management(IM), Intelligent Systems(IS), Networking and Communications(NC), Operating Systems(OS), Platform-based Development(PBD), Parallel and Distributed Computing(PD), Programming Languages(PL), Software Development Fundamentals(SDF), Software Engineering(SE), Systems Fundamentals(SF), Social Issues and Professional Practice(SP)

Informatics Curriculum for Beginners at Universities (Plan)

*Hours: Minimum teaching-learning hours

Competency	Areas	Required Topics	Optional Topics
Technical understanding of computer science / Familiarity with common themes and principles / Appreciation of the interplay between theory and practice /	Architecture and Organization	○ Numerical expressions on computers (0.5 hours) -Bits -Bytes -Binary number -Decimal number -Binary notation -Hexadecimal notation ○ Data representation (0.5 hours) -Data and metadata -File formats and encoding -Text/binary files	○ Information theory (2 hours) -Information quantitation -Entropy -Entropy encoding

System-level perspective / Problem solving skills / Commitment to life-long learning / Commitment to professional responsibility / Awareness of the broad applicability of computing / Appreciation of domain-specific knowledge		-Text: Unicode -(Digital) Image: RGB, Pixel -(Digital) Sound and music: MIDI -Video ○ Digitalization of analog information (1 hour) -Sampling -Compression -Quantizing -Encoding	
	Discrete Structures	○ Introduction to discrete structures (1 hour) -Truth tables -Logical function -Boolean algebra -Understand basic probability	○ Understand basic set theory and set-theoretic notation (1 hour) ○ Advanced discrete structures -Truth tables (1 hour) -Logical function (1 hour) -Boolean algebra (1 hour) -Understand basic probability (2 hours) ○ Graphs (1 hour)
	Systems Fundamentals	○ Introduction to internal structures and operations of computer (1 hour) -Components of computer systems: hardware and software (introduction) -Hardware 1. CPU(Architecture) = Processor 2. Memory(=Ram): hard disk, SD card, USB flash memory USB 3. Storage	○ Advanced internal structures and operations of computer -Von Neumann architecture (1 hour) -Logic circuit (2 hours) -Principles of error correction (2 hours) -Computer commands (2 hours) -Input/output device (1 hour)
		-	○ Introduction to network (1 hour) [Examples] -Introduction to the internet -Internet connectivity -The web -Protocol 1. Client/server 2. HTTP 3. Hostname 4. DNS 5. TCP/IP ○ Social networking (1 hour) -SNS ○ Hardware and naming schemes (1 hour)
		-	○ Operating systems (2 hours) -What is OS? -Why is OS necessary? -Introduction of features of operating systems: process management, memory management, interrupt management, and other tasks of OS -What is the file system? ○ Real-time OS (1 hour)
		-	○ Parallel processing (2 hours) -Why is parallel processing necessary?

			-Instruction level parallelism -SIMD parallelism -Multicore parallelism ○ Distributed processing (2 hours) -Web service -Internet -Domain name -Scalability
		○ Introduction to web programming language (2 hours) -Why do programming languages exist? -Why do we have many programming languages? -Roles of programming languages	○ Advanced programming language (3 hours) -Various programming models -CPUs and machine language -Language processors -Program efficiency -Compiler ○ Program development (3 hours) -Object-oriented design -Top-down design ○ Object-oriented programming (OOP) (2 hours) -Objects -Class -Subclasses -Inheritance -Overriding ○ Data types -Primitive data types (e.g., number, bool) (6 hours) -Composite data types (e.g., record, union, array, list, function, reference) (6 hours) ○ Functional programming -Lambda calculus (4 hours) -Higher-order functions (4 hours) ○ Advanced programming structures (2 hours) -String manipulation -Assertions ○ Logic programming (2 hours)
		-	○ Various examples of information systems (2 hours) -Planning information systems -Types and uses of information systems -Understanding future IT ○ Databases (3 hours) -Information system hardware and data processing -Development history and roles of databases -Characteristics of operational databases -Social functions of operational systems -Introduction of multidimensional analytical database

			-Understanding multidimensional OLAP analysis -Characteristics of data warehouse -Why are databases necessary? -Understanding data quality ○ Data modeling (2 hours) -Network models -Hierarchical model -Introduction to relational databases ○ Database system (2 hours) -SQL and relationship algebra -NoSQL -Normal form and consistency -Trouble recovery ○ SQL (8 hours) -Understanding database language SQL -Relational databases: SQL, Tables -Creating a database -Querying using SQL -Advanced Querying using SQL -SQL with Python
	Algorithms and Programming	-	○ Algorithm complexity -Criteria for good algorithms (e.g., What is Arthur Benjamin's multiplication algorithm? How does it compare to algorithms you know for long multiplication?) (1 hour) -Asymptotic notation (Big O) (1 hour) -Algorithmic complexity (1 hour) (Efficiency of algorithms, computational complexity (time complexity, space complexity)) -Limits of computation (1 hour): Incomputability ○ Sorting algorithm (3 hours) -Selection sort -Bubble sort -Merge sort ○ Searching algorithm (2 hours) -Linear search -Binary search ○ Kinds of algorithm (6 hours) -Greedy algorithm -Divide-and-conquer -Dynamic programming -Backtracking
		○ Computational Thinking (Algorithmic Thinking) (1 hour) 1. Decomposition 2. Abstractions 3. Testing 4. Debugging ○ Introduction to algorithm and programming (1 hour)	**[For courses focused on programming] (31 hours)** ○ Advanced programming concept I (4 hours) -Basic syntax -Variables -Data types (e.g., number, character string) -I/O

		-What is computation? 1. What is an algorithm? 2. What is computer program? 3. What is programming? ○ Data structures (2 hours) [Examples] -Arrays: Arrays of objects, Built-in array functions -Characters -Strings -Numbers -Lists; Linked lists -Queues -Stacks -Dictionary -Tuple -Sets -Trees -Hash table -Sequences (lists, strings, tuple) -Container type (dictionary, tuple, set) -Abstract data structure (FIFO, LIFO, dictionary) -Aliasing ○ Introduction to programming concept I (2 hours) -Basic syntax -Variables -Data types (e.g., number, character string) -I/O ○ Introduction to programming concept II (2 hours) -Control structures (Control flow statements) 1. Sequential processing 2. Conditional branch 3. Iteration	○ Advanced programming concept II (4 hours) -Control structures (control flow statements) 1. Sequential processing 2. Conditional branch 3. Iteration -Conditional statements (2 hours) (conditionals/branching/ selection statements decision) 1. If-statements 2. Switch -Iteration (2 hours) (loops/loop statements) 1. While loop 2. For loop -Operator (2 hours) 1. Arithmetic operator 2. Comparison operator 3. Logical operator -Functions (4 hours) 1. Function definition 2. Function call 3. Argument 4. parameter 5. Local variable 6. Global variable -Recursion (2 hours) -Module (4 hours) -Exceptions (4 hours) ○ Programming development method -Programming style (1 hour) -Use of libraries (practice) (2 hours)
	Platform-based Development	-	○ Overview of platform-based development (1 hour) [Examples] -An overview of architectural styles across multiple platforms including web, mobile, apps, industrial, game, robotics, etc. -Integration of transaction back-end system and front-end web technology in web architecture ○ Front-end web technology (HTML/CSS/JavaScript) -Introduction to HTML (1 hour) [Examples] 1. Version of HTML 2. HTML document structure 3. Body elements 4. About “tags” 5. Building your own web page HTML -Introduction to CSS (1 hour) [Examples] 1. Roles of CSS

			2. Forms of CSS 3. Properties of CSS 4. Other methods of creating websites -JavaScript (2 hours) 1. JavaScript: basics 2. JavaScript: decisions 3. JavaScript: iteration and arrays 4. JavaScript: functions and events ○ Web design -Webpage layout (2 hours) 1. Webpage layout 2. Layout example: New York Times 3. Layout: Washington Post -Responsive web design (2 hours) 1. What is responsive web? 2. Using HTML5 3. Using CSS3 4. Forms
	Graphics and Visualization	-	○ Computer graphics (1 hour) [Examples] -How to compute 'Shading' -What existing 'Umwelt' extensions do humans already enjoy? -Does 'Visualization' qualify as an Umwelt extension? -Why is good picture-making an 'Optimization' process?
		-	○ Examples of modeling (3 hours) [Examples] 1. Environmental and social phenomena -Modeling environmental and social phenomena -Predicting market development 2. Chemical and physical phenomena -Modeling chemical and physical phenomena -Solving differential equations using Excel -Controlling heat transfer by balancing heating and cooling -Predicting intimate heat transfer 3. World of atoms -Modeling the world of atoms -Molecular dynamics predicting 1 picosecond in the future by solving the world of atoms and molecules -Becoming professionals who visualize solid, liquid, and gas states and 'visualize and fascinate' analysis results
	Data Science and Artificial Intelligence	○ Overview of data science (1 hour) -Data science in a nutshell: what and why data science; fundamental principles guiding data science; impact of data science across fields	**[For courses focused on data analysis] (26 hours)** ○ Data mining -What is data analytics? (2 hours) -Data analysis using programming languages (Excel, Python, Map-Reduce, R) 1. Data preprocessing (2 hours) 1) Descriptive statistics of data

		-Introduction and review of recent developments and application of data sciences	2) Data correlation 3) Data processing: cleaning, counting 2. Data similarity (2 hours) 3. Data clustering (2 hours) 4. Association rule discovery (2 hours) 5. Network analysis (2 hours) 6. Text mining (2 hours) -Data analytics applications (Google Analytics, Scientific Data Sets, Open Data, Web Data Services, Data Integration) (2 hours) -Understanding big data processing and analysis technologies (2 hours) ○ Web crawling (4 hours) ○ Data visualization (4 hours) [Examples] -Basic charts and plots, multivariate data visualization, text rendering -Applications and case studies of and data visualizations -Reporting -Data visualization software: Plotting data in Python: the use of library matplotlib -Principles of visual design, perception and color theory -Examples of effective visualization for diverse types of datasets, e.g. matrices, graphs, trees, scalar fields, vector fields, high-dimensional data, etc. -Annotation in data visualization -One of professional software for data visualization: Tableau, QlikView, Power BI, d3.js, Gephi, Weka, etc.
		○ Overview of artificial intelligence (1 hour) [Examples] -Concept of AI -History of AI -Strong AI, weak AI -Turing test -Chinese room -Examples of successful recent AI applications (e.g., IBM Watson, Google self-driving cars) -What makes something intelligent? -Limits of AI -Algorithms used in AI (e.g., pattern recognition) ○ Introduction to machine learning (1 hour) [Examples] -What is learning? -Why have machines learn?	**[For courses focused on data analysis] (52 hours)** ○ AI principal areas (1 hour) -Reasoning -Knowledge representation ○ Advanced machine learning (5 hours) [Examples] -Kinds of machine learning: regression, supervised learning, unsupervised learning, reinforcement learning (1 hour) -Writing machine learning programs using Python (2 hours) - MIL (Multiple instance learning) technique (2 hours) ○ Neural networks (30 hours) [Examples] -Overview of neural networks (2 hours)

		-Kinds of machine learning: regression, supervised learning, unsupervised learning, reinforcement learning ○ Practice of AI services available (2 hours) [Examples] -AI creative services (e.g., Google Doodle, Living Archive, Deep Dream Generator) -AI recognizing pictures (e.g., AutoDraw, Quick, draw!) -AI translation services (e.g., Google Translate, NAVER Papago) -AI search service (e.g., Google Lens, Naver Smart Lens)	-Neutral network problem (2 hours) -Perceptron (2 hours) -MLP (Multi-layer perceptron) (2 hours) -Backpropagation (2 hours) -Convolutional neural network (CNN) (3 hours) -Classification (3 hours) -Data augmentation (3 hours) -Various CNN localizations (CAM) (3 hours) -Segmentation (FCN) (3 hours) -GAN (Generative adversarial network) (3 hours) -Intelligent search (2 hours) ○ Deep learning (2 hours) -Introduction to deep learning -Technologies related to deep learning -Deep learning problem -MNIST classification ○ Computer vision (2 hours) -How can we program a computers to 'see' in 3D? -How good is 3D capture? -Is 2D any simpler? Why? -Traditional CV vs. deep neural nets/learning -Introduction of computer vision application areas -Introduction of various computer vision applications ○ Other AI theories (5 hours) [Examples] -Sampling and randomness (1 hour) -Evolutionary algorithms (1 hour) -Natural language understanding: Chatbot (1 hour) -Expert system (1 hour) -Game theory and auctions (1 hour) ○ Robotics (7 hours) -Overview of robotics (1 hour) -Sensor, actuator, module, input/output, communication (2 hours) -Automatic driving: latest trend, basic information processing, and social tasks of introducing automatic driving of vehicles (2 hours) -Robots: latest trend of robot research applied to industrial plants and everyday lives, information technologies used in robots, and social tasks of introducing robots (2 hours)
	Human-Computer Interaction	-	○ User interface (1 hour) -Definition and functions of interfaces

			-Graphical user interface (GUI) ○ Interaction (2 hours) [Examples] -Gulf of execution -Gulf of evaluation -Affordances, signifiers - gulf of execution -Perception, cognition and the gulf of evaluation -Perceived causality and the gulf of evaluation ○ HCI methods (2 hours) [Examples] -User-centered design -Design process: maximizing good ideas -Rapid prototyping -Visual design elements (layout, color, font, labeling) ○ Future of HCI (1 hour) [Examples] -Bret Victor's brief rant: what's wrong with this picture? -Michael Nielson's vision: primitives for thinking -Fei-Fei Li on human-centered AI
	Social Issues and Professional Practice	○ Information society and IT (change/evolution) (2 hours) [Examples] -Computer development history (various computers) -Systems and networks 1. The expanding network 2. And much of the growth is mobile 3. What's wrong with mobile? 4. For better access 5. Internet connectivity -Social problems and how information technologies can solve them: Case study -Computing and education -Contact points between SNS, GPS, big data and society -Merits and demerits of the internet (will it accelerate democracy, is the internet a public space or a community?) ○ Introduction to modern information technologies (1 hour) [Examples] -Sharing economy -Sharing model: B2C(Business-to-customer), P2P(Peer-to-Peer) -E-commerce -Digital marketing -Location-based services -Evolving computers (e.g., supercomputers, wearable	○ AI algorithmic bias (4 hours) -The basic methodology of machine learning 1. Features and labels 2. Training procedures and evaluation 3. The problem of interpretability -Sources of bias in algorithmic decision-making 1. Bias in features and labels 2. Bias in training procedures 3. Implications for discrimination law ○ Introduction to information security (3 hours) -Computer security: examples and mechanisms of security, defense measures, threats, attacks, and system vulnerabilities (2 hours) -Using IT services in a secure manner (1 hour) [Examples] 1. Passwords 2. Using virtual private network (VPN) 3. Secure browsing 4. Securing Wi-Fi connection 5. Using security suite software 6. Configuring personal firewall 7. Junk mail filtering and sorting 8. Email, private mode

		computers, VR, AR, 3D printers, deep learning, quantum computers, IoT (Internet of Things)) -Digital computing (hardware, software, peopleware) -Networking and communication systems -Data storage technologies -Internet and the web (technologies, services, applications) -Smart city technologies (e.g., Internet-of-things (IoT), big data/data mining, open data, cloud, mobile apps, e-government, artificial intelligence, etc.) -Smart city applications (e.g., smart energy, environment, waste, government, community, transportation, smart building/smart homes, public health, safety, etc.) -Various AI technologies and methods (e.g., AI applications in medicine/health, fintech, smart city, lawtech, insurtech, etc.) ○ Knowledge required to properly handle information in society (1 hour) -Basic information about appropriate expression methods and credibility judgment to deliver proper information to others and not misinterpret information from others -Basic knowledge about handling information for adequate decision-making in society -Basic methods of analyzing human behavior in society on the information level ○ Copyright (1 hour) [Examples] -Ethical and cultural problems regarding the use of and handling digital information - Digital right management (DRM) -Digital watermarking -Illegal reproductions -Boundary between rights and ownership -Open source project ○ Privacy (1 hour) [Examples] -Values served by privacy	○ Information system security issues and measures by case studies (2 hours) ○ Information security policies and practices (2 hours) ○ Encryption technologies and applications (3 hours) -How to use encryption while transferring data and using mobile data storage devices -How to use password-protection while transferring data and using mobile data storage devices -Symmetric-key cryptography -Public-key cryptography (SSL, https) -Electronic signatures/digital signature -Encryption of wireless LAN -Blockchain -PKI (Public key infrastructure) -Zero-knowledge -Application in secret calculations and anonymous authentication ○ History of cryptography (2 hours) ○ Network and security (2 hours) -Security basics -Classification of security threats -Internet security -Safety of mutual authentication ○ Cybersecurity (2 hours) [Examples] -Cyberspace -Cybercrime (computing crime) -Cyber law -Cyber ethics -Cyber policy -Cyber defense -Cyberattack

		-The Panopticon -Contextual integrity -Psychological dimensions of privacy -Evaluating common fallacies about privacy -Boundary between privacy and security -Differential privacy -Technologies and legal systems protecting privacy 1. Legal grounds for protecting privacy 2. Technical solutions to protect privacy ○ Ethics in IT (1 hour) [Examples] -Freedom of expression -Relevant ethics in IT legislations -Ethical issues related to big data and technologies -Precautions about emails 1. What should not be done using emails 2. Email etiquette 3. Precautions about spam emails 4. Precautions about viruses 5. Precautions about phishing frauds -Side effects and etiquette on SNS -Computing crimes	
Project experience	Software Engineering	○ Team learning (2 hours) -Debates, role plays, group assignments, etc.	**[For courses focused on project activities]** (28 hours) ○ Project activities -Preparation of project overview -Preparation of project plan -Project concept design -Detailed project design -Project execution (checking progress of each step (major changes, expected effects, improvements) -Prototyping -Detailed system design and major functions -Project evaluation (activity report, mutual evaluation by team members, progress report, etc.)
Communication and organizational skills	Social Issues and Professional Practice	○ Presentations (2 hours) -Investigation, analysis, discussion, and presentation of social issues	-