# Appendix A: Courses Available for CSA Graduate Credit

Unless otherwise stated, the courses listed here involve 3 credits.

Unless otherwise stated, the courses listed here must be taken for an A-F grade.

Unless otherwise stated, a course from the list below cannot be repeated on a program of study.

**CS 4104: Data and Algorithm Analysis**

This course emphasizes the understanding of data structures and algorithms from an analytical perspective rather than from an implementation standpoint. The concepts developed allow discussion of the efficiency of an algorithm and the comparison of two or more algorithms with respect to space and run-time requirements. Analytical methods are used to describe theoretical bounds as well as practical ones. In general, this course addresses the constraints that affect problem solvability. A grade of C or better required in CS prerequisite 2604 or 2606. Pre: (2604 or 2606), (MATH 3134 or MATH 3034).

**CS 4114: Introduction to Formal Languages and Automata Theory**

The course presents a study of formal languages and the correspondence between language classes and the automata that recognize them. Formal definitions of grammars and acceptors, deterministic and nondeterministic systems, grammar ambiguity, finite state and push-down automata, and normal forms will be discussed. Pre: MATH 3134 or MATH 3034.

**CS 4124: Theory of Computation**

Theoretical analysis of the computational process; fundamental concepts such as abstract programs, classes of computational machines and their equivalence, recursive function theory, unsolvable problems, Church's thesis, Kleene's theorem, program equivalence, and generability, acceptability, decidability will be covered. Pre: MATH 3134 or MATH 3034.

**CS 4204: Computer Graphics**

Hardware and software techniques for the display of graphical information. 2D and 3D geometry and transformations, clipping and windowing, software systems. Interactive graphics, shading, hidden surface elimination, perspective depth. Modeling and realism. A grade of C or better required in CS prerequisite 2604 or 2606. Pre: 2604 or 2606.

**CS 4214: Simulation and Modeling**

Overview of discrete-event digital computer simulation and modeling. Fundamentals of model development, Monte Carlo simulation, the life cycle of a simulation study, input and output data analysis, world views and time control, random number and variate generation, credibility assessment of simulation results, simulation languages, applications of simulation using the General Purpose Simulation System (GPSS). A grade of C or better required in CS prerequisite 1706. Pre: 1706, STAT 4714.

**CS 4224: Performance Evaluation of Computer Systems**

Overview of techniques for measuring, improving, and tuning the performance of computer systems. Procurement, workload characterization, measurement principles, the representation of measurement data, software and hardware monitors, capacity planning, bottleneck detection, system and program tuning, simulation and analytic models and their applications, case studies. Pre: 3204, (STAT 4714 or STAT 4105 or STAT 4705).

**CS 4234: Parallel Computation**

Survey of parallel computer architectures, models of parallel computation, and interconnection networks. Parallel algorithm development and analysis. Programming paradigms and languages for parallel computation. Example applications. Performance measurement and evaluation. A grade of C or better required in CS prerequisite 3204. Pre: 3204.

**CS 4244: Internet Software Development**

Key technology underlying the World-Wide Web. Web architecture, including server design, caching, network protocols, and related standards (e.g. http, SHTTP, TCP/IP, MIME). Programming systems (e.g. Java, Active-X, component models). Security and cryptography. Document representations (e.g. XML, HTML, PDF, VRML). Legal and social issues of the Web. A grade of C or better required in CS prerequisite 3204. Pre: 3204.

**CS 4254: Computer Network Architecture and Programming**

Introduction to computer network architecture, and methods for programming network services and applications (e.g. DNS, Email and MIME, http, SNMP, multimedia). Wired, wireless, and satellite network architectures. OSI protocol model, with an emphasis on upper layers. Congestion control, quality of service, routing. Internet protocol suite (e.g. IP, TCP, ARP, RARP). Server design (e.g. connectionless, concurrent). Network programming abstractions (e.g. XDR, remote procedure calls, sockets, DCOM). Case studies (e.g. TELNET). A grade of C or better required in CS prerequisite 3204. Pre: 3204.

**CS 4304: Compiler Design and Implementation**

This course includes the theory, the design, and the implementation of a large language translator system. Lexical analysis, syntactic analysis, code generation, and optimization are emphasized. A grade of C or better required in CS prerequisite 3204. Pre: 3204.

**CS/MATH 4414: Issues in Scientific Computing**

Theory and techniques of modern computational mathematics, computing environments, computational linear algebra, optimization, approximation, parameter identification, finite difference and finite element methods and symbolic computation. Project-oriented course; modeling and analysis of physical systems using state-of-the-art software and packaged subroutines. Pre: MATH 2214, MATH 3214. (2H,3L,3C)

**CS/ECE 4504: Computer Organization**

Information representation and transfer; instructions and data access methods; the control unit and microprogramming; memories; input/output and interrupts; secondary storage; the von Neumann SISD organization; high level language machines; the RISC concept; special purpose processors including operating system, file, text, floating point, communication, etc. Multicomputers; multiprocessors; concurrent processing support; Pipeline machines, processor arrays, database machines; the data flow/data directed approach; computer networks. A grade of C or better required in CS prerequisite 3204. Pre: 3204.

**CS/ECE 4570: Wireless Networks and Mobile Systems**

Multidisciplinary, project-oriented design course that considers aspects of wireless and mobile systems including wireless networks and link protocols, mobile networking including support for the Internet Protocol suite, mobile middleware, and mobile applications. Students complete multiple experiments and design projects. Pre: 4254 or ECE 4564.

**CS 4604: Introduction to Database Management Systems**

Emphasis on introduction of the basic data base models, corresponding logical and physical data structures, comparisons of models, logical data design, and data base usage. Terminology, historical evolution, relationships, implementation, data base personnel, future trends, applications, performance considerations, data integrity. Senior standing required. A grade of C or better required in CS prerequisite 2604 or 2606. Pre: 2604 or 2606.

**CS 4624: Multimedia, Hypertext, and Information Access**

Introduces the architectures, concepts, data, hardware, methods, models, software, standards, structures, technologies, and issues involved with: networked multimedia (e.g., image, audio, video) information, access and systems; hypertext and hypermedia; electronic publishing; virtual reality. Coverage includes text processing, search, retrieval, browsing, time-based performance, synchronization, quality of service, video conferencing and authoring. Senior standing required. A grade of C or better required in CS prerequisite 2604. or 2606. Pre: 2604 or 2606.

**CS 4634: Design of Information**

Survey of the higher-order properties that allow data to become information, that is, to inform people. The course focuses on the design of user interface layouts, and on the design of texts and hypertexts, as well as on the information development process. Senior standing required. A grade of C or better required in CS prerequisite 2604 or 2606. Pre: 2604 or 2606.

**CS 4704: Software Engineering**

Introduction to the basic principles of software engineering. Issues in the software life cycle. Emphasis on methods for software design and testing. Project management and quality assurance. Significant software project required. A grade of C or better required in CS prerequisite 3704. Pre: 3704.

**CS 4804: Introduction to Artificial Intelligence**

Overview of the areas of problem solving, game playing, and computer vision. Search trees and/or graphs, game trees, block world vision, syntactic pattern recognition, object matching, natural language, and robotics. Senior standing required. A grade of C or better required in CS prerequisite 2604 or 2606. Pre: 2604 or 2606.

**CS 5014: Research Methods in Computer Science**

Preparation for research in computer science. Technical communication skills. Design and evaluation of experiments. The research process.

**CS 5104: Computability and Formal Languages**

Formal theory of computability, the halting problem, models of computation, and Church's thesis, and formal languages.

**CS 5114: Theory of Algorithms**

Methods for constructing and analyzing algorithms. Measures of computational complexity, determination of efficient algorithms for a variety of problems such as searching, sorting and pattern matching. Geometric algorithms, mathematical algorithms, and theory of NP-completeness.

**CS 5124: Algorithms in Bioinformatics**

Algorithms to solve problems found in biology, especially molecular biology. A variety of current problems in computational molecular biology will be introduced, investigated, analyzed for computational complexity, and solved with efficient algorithms, when feasible. A number of such problems will be shown to be intractable or other evidence of their difficulty will be presented. Prerequisites or graduate standing in CSA required.

**CS 5204: Operating Systems**

Issues in the design and functioning of operating systems. Emphasis on synchronization of concurrent activity in both centralized and distributed systems. Deadlock, scheduling, performance analysis, operating system design, and memory systems including distributed file systems.

**CS 5214: Modeling and Evaluation of Computer Systems**

An overview of modeling, simulation, and performance evaluation of computer systems, i.e., operating systems, database management systems, office automation systems, etc. Fundamentals of modeling, the life cycle of a simulation study, workload characterization, random number and variate generation, procurement, measurement principles, software and hardware monitors, capacity planning, system and program tuning, and analytic modeling. Duplication of subject matter of 4214 and 4224. Maximum of 6 hours credit may be obtained from 4214, 4224, 5214.

**CS 5224: Systems Simulation**

An in-depth treatment of systems simulation and simulation programming languages (SPLs). Input data modeling, simulation model formulation and representation, conceptual frameworks for modeling, a comparative study of some SPLs, principles of SPL design, statistical analysis of simulation output data, credibility assessment stages, model development environments.

**CS 5234: Advanced Parallel Computation**

Survey of leading high-end computing systems and their programming environments. Advanced models of parallel computation. Mapping of parallel algorithms to architectures. Performance programming and tools for performance optimization on parallel systems. Execution environments and system software for large-scale parallel computing. Case studies of parallel applications. Graduate standing required.

**CS 5244: Internet Software**

Languages and technologies needed to develop software for the Internet and world-wide web (WWW). Commonly used protocols and standards. Advanced technologies for distributed computation, component-based systems, interoperability with legacy systems, and database access. Principles and technologies for agent-based systems and electronic commerce. Credit will not be given for both 4244 and 5244.

**CS 5304: Translator Design and Construction**

Fundamental theory of parsing and translation and practical applications of this theory. Lexical analysis, parsing techniques based on top-down (LL, Recursive Descent) and bottom-up (LR, Precedence), code generation, code optimization techniques, and runtime systems.

**CS 5314: Programming Languages**

Indepth investigation of the principles of programming systems, not necessarily restricted to programming languages, both from the point of view of the user implementor. Algorithms of implementation, syntax and semantic specification systems, block structures and scope, data abstraction and aggregates, exception handling, concurrency, and applicative/functional/data-flow languages.

**CS/GBCB/BIOL 5424: Computational Cell Biology**

Use of mathematical models (nonlinear ordinary differential equations and stochastic processes) and simulation algorithms to explore the complex feedback circuits that control the behavior of living cells. Concepts and techniques from dynamical systems theory, bifurcation analysis, numerical methods, SBML (systems biology makeup language) and Matlab programming. Applications in gene regulatory networks, cell cycle control, circadian rhythms, cell signaling.

**CS/MATH 5465: Numerical Analysis**

A survey of the construction, analysis, and implementation of numerical algorithms in linear algebra, nonlinear equations and optimization, approximation by polynomials, quadrature, and ordinary differential equations.

**CS/MATH 5466: Numerical Analysis**

A survey of the construction, analysis, and implementation of numerical algorithms in linear algebra, nonlinear equations and optimization, approximation by polynomials, quadrature, and ordinary differential equations.

**CS/MATH 5474: Finite Difference Methods for Partial Differential Equations**

Finite difference methods for initial and boundary value problems for partial differential equations. Consistency, stability, convergence, dispersion, and dissipation. Methods for linear and nonlinear elliptic and parabolic equations, first- and second-order hyperbolic equations, and nonlinear conservation laws.

**CS/MATH 5484: Finite Element Methods for Partial Differential Equations**

Weak formulations of boundary-value problems for elliptic partial differential equations. Finite element spaces. Approximation theory for finite element spaces. Error estimates. Effects of numerical integration and curved boundaries. Nonconforming methods. Concrete examples of the application of the finite element method. Efficient implementation strategies. Time dependent problems.

**CS/MATH 5485: Numerical Analysis and Software**

Presentation and analysis of numerical methods for solving common mathematical and physical problems. Methods of solving large sparse linear systems of equations, algebraic eigenvalue problems, and linear least squares problems. Numerical algorithms for solving constrained and unconstrained optimization problems. Numerical solutions of nonlinear algebraic systems. Convergence, error analysis. Hardware and software influences. Efficiency, accuracy, and reliability of software. Robust computer codes.

**CS/MATH 5486: Numerical Analysis and Software**

Presentation and analysis of numerical methods for solving common mathematical and physical problems. Methods of solving large sparse linear systems of equations, algebraic eigenvalue problems, and linear least squares problems. Numerical algorithms for solving constrained and unconstrained optimization problems. Numerical solutions of nonlinear algebraic systems. Convergence, error analysis. Hardware and software influences. Efficiency, accuracy, and reliability of software. Robust computer codes.

**CS/ECE 5504: Computer Architecture**

Advanced computer architectures, focusing on multiprocessor systems and the principles of their design. Parallel computer models, programming and interconnection network properties, principles of scaleable designs. Case studies and example applications of pipeline processors, interconnection networks, SIMD and MIMD processors.

**CS/ECE 5510: Multiprocessor Programming**

Principle and practice of multiprocessor programming. Illustration of multiprocessor programming principles through the classical mutual exclusion problem, correctness properties of concurrency (e.g., linearizability), shared memory properties (e.g. register constructions), and synchronization primitives for implementing concurrent data structures (e.g., consensus protocols). Illustration of multiprocessor programming practice through programming patterns such as spin locks, monitor locks, the work-stealing paradigm and barriers. Discussion of concurrent data structures (e.g., concurrent linked lists, queues, stacks, hash maps, skiplists) through synchronization patterns ranging from coarse-grained locking to fine-grained locking to lock-free structures, atomic synchronization primitives, elimination, and transactional memory.

**CS/STAT 5525: Data Analytics I**

Basic techniques in data analytics including the preparation and manipulation of data for analysis and the creation of data files from multiple and dissimilar sources. The data mining and knowledge discovery process. Overview of data mining algorithms in classification, clustering, association analysis, probabilistic modeling, and matrix decompositions. Detailed study of classification methods including tree-based methods, Bayesian methods, logistic regression, ensemble, bagging and boosting methods, neural network methods, use of support vectors and Bayesian networks. Detailed study of clustering methods including k-means, hierarchical and self-organizing map methods.

**CS/STAT 5526: Data Analytics II**

Techniques in supervised, unsupervised, and visualized learning in high dimensional spaces. Theoretical, probabilistic, and applied aspects of data analytics. Methods include generalized linear models in high dimensional spaces, regularization, lasso and related methods, principal component regression (pca), tree methods, and random forests. Clustering methods including k-means, hierarchical clustering, biclustering, and model-based clustering will be thoroughly examined. Distance-based learning methods include multi dimensional scaling, the self organizing map, graphical/network models, and isomap. Supervised learning will consist of discriminant analyses, supervised pca, support vector machines, and kernel methods.

**CS/ECE 5560: Network and Computer Security**

Introduces both fundamental security principles as well as real-world applications of network and computer security. Covers a wide range of topics including authorization and access control, basic cryptography, authentication systems, e-commerce security, sensor network security, and legal and ethical issues. Graduate standing required.

**CS/ECE 5565: Network Architecture and Protocols**

Principles and concepts of networking and protocols, with emphasis on data link, network, and transport protocols. Contemporary and emerging networks and protocols to illustrate concepts and to provide insight into practical networks including the Internet. Quantitative and qualitative comparisons of network architectures and protocols.

**CS/ECE 5566: Network Architecture and Protocols**

Performance evaluation, design, and management of networks. Use of queuing and other analytical methods, simulation, and experimental methods to evaluate and design networks and protocols. Network management architectures and protocols. Graduate standing in EE, ECE, or IT is required.

**CS 5604: Information Storage and Retrieval**

Analyzing, indexing, representing, storing, searching, retrieving, processing and presenting information and documents using fully automatic systems. The information may be in the form of text, hypertext, multimedia, or hypermedia. The systems are based on various models, e.g., Boolean logic, fuzzy logic, probability theory, etc., and they are implemented using inverted files, relational thesauri, special hardware, and other approaches. Evaluation of the systems' efficiency and effectiveness. Graduate standing required.

**CS 5614: Database Management Systems**

Emphasizes concepts, data models, mechanisms, and language aspects concerned with the definition, organization, and manipulation of data at a logical level. Concentrates on relational model, along with introduction to design of relational systems using Entity-relationship modeling. Functional dependencies and normalization of relations. Query languages, relational algebra, Datalog, and SQL. Query processing, logic and databases, physical database tuning. Concurrency control, OLTP, active and rule-based elements. Data Warehousing, OLAP.

**CS 5634: Data Management in Bioinformatics**

Data models, query languages, and data management systems for bioinformatics applications. Logical data organization, functional dependencies, design of schemas, querying, manipulation, information integration, and data mining. Specialized data structures, interchange formats, and designs for applications such as sequencing and microarray analysis. Partially duplicates 5614. Prerequisite or graduate standing in CSA required.

**CS 5704: Software Engineering**

Study of the principles and tools applicable to the methodical construction and controlled evolution of complex software systems.All phases of the life cycle are presented; particular attention focuses on the design, testing, and maintenance phases. Introduction to software project management. Attention to measurement models of the software process and product which allow quantitative assessment of cost, reliability, and complexity of software systems.

**CS/ISE 5714: Usability Engineering**

Design and evaluation of effective user interfaces, beginning with principles for designing the product. Development process for user interaction separate from interactive software development. Development process includes iterative life cycle management, systems analysis, design, usability specifications, design representation techniques, prototyping, formative user-based evaluation. Integrative and cross-disciplinary approach with main emphasis on usability methods and the user interaction development process.

**CS 5724: Models and Theories of Human-computer Interaction**

Survey of models and theories of users and their use of computer equipment; conditions of application for various approaches. Task analysis, task modeling, representations and notations.

**CS 5734: Computer-supported Cooperative Work**

Review and critique of state-of-the-art computing systems supporting cooperative work. Introduction to toolkits, software architectures and implementation issues relevant to development of systems for cooperative work. Analysis of group interactions and concerns in collaborative activities such as writing, design, meetings, communication, and decision-making.

**CS 5744: Software Design and Quality**

This course focuses on critical aspects of the software lifecycle that have significant influence on the overall quality of the software system including techniques and approaches to software design, quantitative measurement and assessment of the system during implementation, testing, and maintenance, and the role of verification and validation in assuring software quality.

**CS 5754: Virtual Environments**

Introduction to the theory and practice of three-dimensional virtual environments (VEs). 3D input and output devices, applications of VEs, 3D user interfaces and human-computer interaction, 3D graphics techniques for VEs, 3D modeling and level of detail, evaluation of VEs, VE software systems and standards, collaborative and distributed VEs. Includes hands-on experience with VE hardware and software.

**CS 5764: Information Visualization**

Examine computer-based strategies for interactive visual presentation of information that enable people to explore, discover, and learn from vast quantities of data. Learn to analyze, design, develop, and evaluate new visualizations and tools. Discuss design principles, interaction strategies, information types, and experimental results. Research-oriented course surveys current literature, and group projects contribute to the state of the art.

**CS 5774: User Interface Software**

Survey of software architectures to build user interfaces, particularly focused on graphical user interfaces. Includes the design and implementation of user interfaces, the use of object-oriented application frameworks, software architecture for command undo, document management, layout managers, customized components, and separation of concerns in user interface software architectures. Discussion of research and advanced topics in User Interface Software.

**CS 5804: Introduction to Artificial Intelligence**

A graduate level overview of the areas of search, knowledge representation,logic and deduction, learning, planning, and artificial intelligence applications.

**CS 5814: Digital Picture Processing**

Representation and processing of greytone images. Construction and simulation of grey scales, digitization, thresholding, local neighborhood operations, template matching and filtering, enhancement and restoration, segmentation, connected components, matching, morphology.

**CS/GBCB 5854: Computational Systems Biology**

Phenomenological and data-driven models of molecular interaction networks. Applications of graph theory, discrete algorithms, data mining, and machine learning to the modeling and analysis of molecular interaction networks. Biological applications. Interaction between biological and computational disciplines in systems biology. Must have GBCB pre-requisite and CS pre-requisites or graduate standing in CSA or equivalent.

**CS 5894: Final Examination**

Pass/Fail only.

**CS 5904: Project and Report**

EQ grade only. Variable credit course. 1 to 19 credit hours.

**CS 5944: Graduate Seminar**

Pass/Fail grade only. 1 credit hour(s).

**CS 5974: Independent Study**

Pass/Fail grade only. Variable credit course. 1 to 19 credit hours.

**CS 5984: Special Study**

Variable credit course. 1 to 19 credit hours. May be repeated with different content for a maximum of 12 credit hours.

**CS 5994 - Research and Thesis**

EQ grade only. Variable credit course. 1 to 19 credit hours.

**CS 6104: Advanced Topics in Theory of Computation**

This course treats a specific, advanced topic of current research interest in the area of theory of computation. Papers from the current literature or research monographs are likely to be used instead of a textbook. Student participation in a seminar style format may be expected. May be repeated with different content for a maximum of 12 credit hours.

**CS 6204: Advanced Topics in Systems**

This course treats a specific advanced topic of current research interest in the area of systems. Papers from the current literature or research monographs are likely to be used instead of a textbook. Student participation in a seminar style format may be expected. May be repeated with different content for a maximum of 12 credit hours.

**CS 6304: Advanced Topics in Languages and Translation**

This course treats a specific advanced topic of current research interest in the area of languages and translation. Papers from the current literature or research monographs are likely to be used instead of a textbook. Student participation in a seminar style format may be expected. May be repeated with different content for a maximum of 12 credit hours.

**CS 6404: Advanced Topics in Mathematical Software**

This course treats a specific advanced topic of current research interest in the area of mathematical software. Papers from the current literature or research monographs are likely to be used instead of a textbook. Student participation in a seminar style format may be expected. May be repeated with different content for a maximum of 12 credit hours.

**CS 6504: Advanced Topics in Computer Architecture**

This course treats a specific advanced topic of current research interest in the area of architecture. Papers from the current literature or research monographs are likely to be used instead of a textbook. Student participation in a seminar style format may be expected. May be repeated with different content for a maximum of 12 credit hours.

**CS/ECE 6564: Multimedia Networking**

This course examines and explores recent advances in multimedia networking technologies. Major topics include multimedia compression and standards, quality of service (QoS) support mechanisms and protocols, performance analysis, network claculus, IP multicasting, Internet multimedia applications, and multimedia transport over wireless networks.

**CS/ECE 6570: Advanced Foundations of Networking**

This course covers theoretical foundations that are necessary for advanced study of networking. It focuses on algorithm design and optimization techniques that are most commonly used to solve complex networking problems. Major topics include complexity analysis with applications to networking problems, design and proof of approximation algorithms, design of meta-heuristic algorithms, formulation techniques for network optimization, linear and non-linear optimization techniques with applications to networking, design of distributed algorithms with proof of convergence for networks systems.

**CS 6604: Advanced Topics in Data and Information**

This course treats a specific advanced topic of current research interest in the area of data and information. Papers from the current literature or research monographs are likely to be used instead of a textbook. Student participation in a seminar style format may be expected. May be repeated with different content for a maximum of 12 credit hours.

**CS 6704: Advanced Topics in Software Engineering**

This course treats a specific advanced topic of current research interest in the area of software engineering. Papers from the current literature or research monographs are likely to be used instead of a textbook. Student participation in a seminar style format may be expected. May be repeated with different content for a maximum of 12 credit hours.

**CS 6724: Advanced Topics in Human-computer Interaction**

Addresses a specific advanced topic of current research interest in the area of human-computer interaction (HCI). Research monographs and papers from the current literature will be used as a source of material too new yet to be in a textbook. Student participation in a seminar-style format. Each offering of this course will address a different subtopic area of HCI. May be repeated with different content for a maximum of 12 credit hours.

**CS 6804: Advanced Topics in Intelligent Systems**

This course treats a specific advanced topic of current research interest in the area of intelligent systems. Papers from the current literature or research monographs are likely to be used instead of a textbook. Student participation in a seminar style format may be expected. May be repeated with different content for a maximum of 12 credit hours.

**CS 6824: Adv Topics Comp Biol & Bioinf**

Addresses a specific advanced topic of current research interest in the area of computational biology and bioformatics (CBB). Research monographs and papers from the current literature used as a source of material too new to be discussed in a textbook. Student participation in a seminar-style format. Each offering of this course will address a different subtopic area of CBB. May be repeated with different content for a maximum of 12 credit hours. Pre: Graduate standing; other pre-requisites may apply.

**CS 7994: Research and Dissertation**

EQ grade only. Variable credit course. 1 to 19 credit hours.

# Appendix B: Area wise Distribution of Courses

Historically, the second digit of the four-digit course numbers was intended to denote the cognizant area of the course, e.g., “1” denoted algorithms/theory, “3” denoted languages, and so on. As the number of areas grew, this convention became difficult and cumbersome to maintain. Hence this appendix gives the authoritative listing of all areas and the graduate courses that fall in each of them. Note that only 5000 and 6000 level CS courses are listed since these are the courses used to assess the breadth requirement for M.S. and Ph.D. degrees. Those 5000 level/above courses not discussed here (such as CS 5994, CS 7994) do not count toward the breadth requirement.

**Area 1: Algorithms and Theory**

CS 5104: Computability and Formal Languages

CS 5114: Theory of Algorithms

CS 6104: Advanced Topics in Theory of Computation

**Area 2: Computer Systems**

CS 5204: Operating Systems

CS 5214: Modeling and Evaluation of Computer Systems

CS 5224: Systems Simulation

CS 5234: Advanced Parallel Computation

CS 5244: Internet Software

CS 6204: Advanced Topics in Systems

**Area 3: Programming Languages**

CS 5304: Translator Design and Construction

CS 5314: Programming Languages

CS 6304: Advanced Topics in Languages and Translation

**Area 4: Numerical and Scientific Computing**

CS/MATH 5465: Numerical Analysis

CS/MATH 5466: Numerical Analysis

CS/MATH 5474: Finite Difference Methods for Partial Differential Equations

CS/MATH 5484: Finite Element Methods for Partial Differential Equations

CS/MATH 5485: Numerical Analysis and Software

CS/MATH 5486: Numerical Analysis and Software

CS 6404: Advanced Topics in Mathematical Software

**Area 5: Computer Architecture and Networking**

CS/ECE 5504: Computer Architecture

CS/ECE 5510: Multiprocessor Programming

CS/ECE 5560: Network and Computer Security

CS/ECE 5565: Network Architecture and Protocols

CS/ECE 5566: Network Architecture and Protocols

CS 6504: Advanced Topics in Computer Architecture

CS/ECE 6564: Multimedia Networking

CS/ECE 6570: Advanced Foundations of Networking

**Area 6: Data and Information**

CS/STAT 5525: Data Analytics I

CS/STAT 5526: Data Analytics II

CS 5604: Information Storage and Retrieval

CS 5614: Database Management Systems

CS 6604: Advanced Topics in Data and Information

**Area 7: Software Engineering**

CS 5704: Software Engineering

CS 5744: Software Design and Quality

CS 6704: Advanced Topics in Software Engineering

**Area 8: Human-Computer Interaction**

CS/ISE 5714: Usability Engineering

CS 5724: Models and Theories of Human-computer Interaction

CS 5734: Computer-supported Cooperative Work

CS 5754: Virtual Environments

CS 5764: Information Visualization

CS 5774: User Interface Software

CS 6724: Advanced Topics in Human-computer Interaction

**Area 9: Intelligent Systems**

CS 5804: Introduction to Artificial Intelligence

CS 5814: Digital Picture Processing

CS 6804: Advanced Topics in Intelligent Systems

**Area 10: Computational Biology and Bioinformatics**

CS 5124: Algorithms in Bioinformatics

CS 5424: Computational Cell Biology

CS 5634: Data Management in Bioinformatics

CS 5854: Computational Systems Biology

CS 6824: Adv Topics Comp Biol & Bioinf

Three courses are treated distinctly. CS 5014: Research Methods in Computer Science is considered to be a course in its own area for the purpose of determining the satisfaction of breadth requirements. The cognizant area of CS 5974: Independent Study depends on the description of the course and is decided by GPC in consultation with the faculty advisor for the course. Similarly, the cognizant area of CS 5984: Special Study depends on the specific offering of the course and is decided by GPC in consultation with the instructor of the course. For questions about area classifications, please contact the AGS.

# Appendix C: Approved Cognate Courses

# Appendix E: Forms

Over the course of your graduate study, you might need to complete several forms. The important ones are listed here for your reference. Those marked with a superscript (\*) are not required for all students. Unless otherwise stated, all completed forms must be turned in to the GC. Always make sure to use the latest/authoritative forms which can had either from the department site, from the graduate school’s website (<http://www.grads.vt.edu>) or the registrar’s website (http://www.registrar.vt.edu).

**Confidentiality Waiver Form\***

**Purpose:** Federal and Virginia Tech policies require that students be made aware of their rights with respect to privacy of student records. See <http://www.registrar.vt.edu/records/ferpa.php> for more details. CS department administrative personnel automatically have access to graduate student records. These include: department head, AGS, and the GC. In addition, graduate students may choose to grant access to their files to any CS faculty member, or they may choose to grant access only to specific faculty members. This is the form that authorizes who, besides the administrative personnel, can have access to a student’s records.

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** Can be submitted at any time prior to graduation. Not required.

**Web:** <http://www.cs.vt.edu/graduate/current_students/forms>/

**Independent Study Request Form\***

**Purpose:** To allow a student to register for CS 5974: Independent Study in a given semester. See Section **Error! Reference source not found.** for more information.

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** Prior to last day of the first week of classes.

**Web:** <http://www.cs.vt.edu/graduate/current_students/forms>/

**Late Drop Form\***

**Purpose:** To withdraw from a course after the drop deadline (see Appendix D: Typical schedule of semester).

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** Prior to the end of that semester after the drop deadline. Requires the instructor’s signature, the AGS’s signature, and the Dean of the Graduate School’s signature. Forms can be obtained from the main office of 2050 Torgersen Hall or 1102 Knowledgeworks II. Obtain the instructor’s signature, then leave the form with the GC, who will obtain the consent of the AGS and forward it to the graduate school.

**Student Resignation/Withdrawal Form\***

**Purpose:** To withdraw from all courses for a given semester.

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** Prior to the end of the given semester. The number of required signatures increases after the first week of classes.

**Web:** <http://www.registrar.vt.edu/forms/documents/resign_withdrawal.pdf>

**Credit Transfer Request Form\***

**Purpose:** To allow courses taken elsewhere to count toward a Virginia Tech plan of study (see Section 8).

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** Prior to the submission of the plan of study and typically in the first semester of graduate studies.

**Web:** <http://www.cs.vt.edu/graduate/current_students/forms>/

**Non-Virginia Tech Committee Member Registration Form\***

**Purpose:** To allow a professor/researcher outside of VT to serve on an advisory committee.

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** Prior to end of second semester in the program, prior to the submission of the plan of study (see below). Requires submission of a current curriculum vitae of faculty member.

**Web:** <http://www.grads.vt.edu/forms/index.html>

**Course Justification Form\***

**Purpose:** To allow coursework 5 years or older to be revalidated so that they can be used on a plan of study.

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** Prior to end of second semester in the program along with the plan of study (see above).

Web: <http://www.grads.vt.edu/forms/index.html>

**MS Plan of Study**

**Purpose:** Identifies the list of courses to be taken to fulfill MS degree requirements and the advisory committee.

**Applies to:** M.S. thesis, M.S. coursework students.

**When to submit:** Prior to end of second semester in the program

**Web:** <http://www.cs.vt.edu/graduate/current_students/forms>/

**Request for Plan of Study Changes\***

**Purpose:** To make changes to the courses listed on a plan of study.

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** After an approved plan of study is on file and prior to submitting an Application for Degree (see below).

**Web:** <http://www.grads.vt.edu/forms/index.html>.

**Change of Committee/Advisor Form\***

**Purpose:** To make changes to committee members, advisor(s), or composition of advisory committee.

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** Prior to submission of the Request to Admit Candidate to the Final Exam (see below).

**Web:** <http://www.grads.vt.edu/forms/index.html>.

**Request for Thesis Option Change\***

**Purpose:** To switch between thesis and non-thesis options.

**Applies to:** M.S. thesis, M.S. coursework students.

**When to submit:** Along with Change of Plan of Study Form (see above) if plan of study has already been submitted.

**Web:** <http://www.grads.vt.edu/forms/index.html>.

**Ph.D. Plan of Study**

**Purpose:** Identifies the list of courses to be taken to fulfill Ph.D. degree requirements and the advisory committee.

**Applies to:** Ph.D. students.

**When to submit:** Prior to end of second semester in the program.

**Web:** <http://www.cs.vt.edu/graduate/current_students/forms>/

**Request to Admit Candidate to the Preliminary Exam**

**Purpose:** Allows a Ph.D. preliminary exam to be scheduled.

**Applies to:** Ph.D. students.

**When to submit:** Two weeks prior to the desired exam date.

**Web:** <http://www.grads.vt.edu/forms/academics/Sched_Prelim.pdf>

**Request to Admit Candidate to the Final Exam**

**Purpose:** Allows a final exam to be scheduled. Can be submitted anytime during a semester but for Ph.D. students desiring to participate in commencement exercises (i.e., “to walk”), there is a decalred last day to have the final exam. This date is typically one month before the last day of classes. Final exams can indeed be scheduled after this date but this doesn’t guarantee ability to participate in commencement exercises. If a student desires “to walk” and requires an extension, the advisor could make a request to the graduate school on the student’s behalf and such requests are considered on an individual basis. If participation in commencement is not a consideration, the last day to have the final exam is typically the day before commencement.

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** At least two weeks prior to the desired exam date.

**Web:** <http://www.grads.vt.edu/forms/index.html>.

**Certificate of Defending Student Status Form\***

**Purpose:** To allow a student to register as a defense-only student in a given semester. This is typically used by students who expected to graduate the previous semester but were unable to do so. They might have hence completed all requirements save for the final exam or defense. Using this form, students register as a defense-only student in the current semester, which involves only one (1) credit and, hence, incur substantially lower tuition fees. But they must defend by a stipulated deadline into the current semester (typically within 2 weeks from the beginning of the semester), beyond which they are required to register for more credits like other students.

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** At least two weeks prior to the desired exam date. Submit along with the “Request to Admit Candidate to the Final Exam” form (see above).

**Web:** <http://www.grads.vt.edu/forms/index.html>.

**Thesis and Dissertation (ETD) Approval Form**

**Purpose:** Registers the thesis with the ETD repository, a pre-requisite for graduation. Similar to the “Request to Admit Candidate to the Final Exam” form, for (only) Ph.D. students, there is an absolute deadline to submit it if the student desires to participate in commencement, which is usually 2 weeks before the last day of classes.

**Applies to:** M.S. thesis and Ph.D. students.

**When to submit:** Within two weeks after successfully completing the final exam.

**Web:** <http://www.grads.vt.edu/forms/index.html>.

**Application for Degree**

**Purpose:** To declare intent to graduate in a given semester. Deadline is typically 1 month after the first day of classes (in order for the student’s name to appear in the commencement bulletins). While this form can be usually submitted online, it MUST be submitted in paper form for the case of a PhD student earning a MS coursework degree “on the way.”

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** As early as possible in the semester that the student intends to graduate. Degree cannot be awarded without this form on record even if the student has satisfied all requirements.

**Web:** <http://www.grads.vt.edu/forms/index.html>

**Application for Certificate Conferral Form\***

**Purpose:** Used when students augment their degree program with certificates such as the “HCI certificate” (see Section **Error! Reference source not found.**). There are more certificates offered by other departments. Similar deadlines as “Application for Degree” apply. Use the same form as the AFD (now called the AFDC).

**Applies to:** M.S. thesis, M.S. coursework, and Ph.D. students.

**When to submit:** As early as possible in the semester that the student intends to receive the certificate.

**Web:** <http://www.grads.vt.edu/forms/index.html>

Please also consult the graduate school’s website (<http://www.grads.vt.edu/forms/>) for an authoritative list of all forms.

# Appendix F: Guidelines for Constituting Ph.D. Qualifying Exam Committees

A PhD qualifying examination committee may be formed in any area recognized by the Department (see Appendix B). There may only be one committee for any area. A faculty member may serve on at most two committees during a given year. Each year, AGS will appoint a chair for each examination committee from among the volunteers for that semester.

The PhD Qualifying Examination is given during a period spanning the end of Fall semester and the start of Spring semester of each year. During early Fall semester, students interested in taking the exam should discuss potential research areas with faculty members so that examination areas of mutual interest can be discovered. Examination committees must post the reading list for their exam by November 1. The exam is normally administered during January and February, with scores reported to GPC by mid February.

Each examination committee will publish a reading list of 10-20 research papers by November 1. It is not a requirement that the papers broadly cover the area, or be "seminal papers" in the area. A list containing papers with results spanning a wide spectrum in regards to quality and relevance is desirable to gauge the student's ability to judge quality and importance of results. The body of work should serve as a good introduction to one or more aspects of the area, but is also selected in part to serve as a vehicle for the exam. For example, a committee giving an exam in algorithms might choose one year to assign a set of papers on NP-complete problems in bioinformatics. The following year, the papers assigned by that committee might be on a completely different topic.

All students taking the exam in a given year from a given examination committee should be given the same reading list, undergo the same examination process, and be graded using the same criteria.

The exam is meant to probe the student's understanding of the content of the papers, the student's ability to synthesize the content into a meaningful understanding of the issues involved, and from there, the student's ability to determine potential "next step" paths of research (based on the papers assigned). In general, the exam is testing the student's ability to critically analyze the material, make judgments regarding the quality and relevance of the results, as well as deriving ideas for future research directions for the specific subtopic addressed in the papers.

At the end of the examination process, the committee must arrive at a scoring in the range 0 to 3 (integer only), and report this score to the AGS by the deadline.