Texas Educator Certification Examination Program

TExES® Computer Science 8–12 Curriculum Crosswalk

Test Content Categories	Required Course Numbers										
Domain I — Technology Applications Core											
Competency 001: *The computer science teacher knows technology terminology and concepts; the appropriate use of hardware, software and digital files; and how to acquire, analyze and evaluate digital information*.											
A. Knows and uses technology terminology and concepts appropriate to the task.											
B. Knows the appropriate use of software and hardware components.											
C. Demonstrates knowledge of various types of networks (e.g., LAN, WAN, intranets and the Internet).											
D. Knows how to select, connect and use a variety of local and remote peripheral devices.											
E. Knows how to manage compatibility issues for a variety of media, file formats (e.g., text, graphics, image, video, audio), file naming conventions, file management structures and digital organization strategies.											
F. Knows how to evaluate software for quality, appropriateness, effectiveness, efficiency, support and licensing to make decisions regarding its proper acquisition and use.											

	Required Course Numbers										
Test Content Categories											
G. Knows how to access, manage and manipulate information from secondary storage devices.											
H. Knows strategies for searching, acquiring and accessing information from electronic resources.											
I. Knows how to assess the accuracy and validity of acquired information and how to resolve information conflicts through research and comparison of data from multiple sources.											
J. Demonstrates knowledge of intellectual property rights (e.g., copyright, Creative Commons, free and open source licensing) when accessing, using, manipulating and editing electronic data.											
K. Demonstrates knowledge of issues of unacceptable use of computer resources including, but not limited to, cyberbullying and harassment, computer hacking, computer piracy, plagiarism, vandalism, intentional virus setting and invasion of privacy.											
L. Demonstrates ethical and lawful acquisition of digital information, including the use of established methods to cite sources.											
M. Understands digital safety, privacy rules, digital etiquette, acceptable use of technology and the ethical and legal responsibilities of using social media.											
N. Knows how to use online help and other support documentation to troubleshoot minor technical problems with hardware and software.											

	Required Course Numbers										
Test Content Categories											
O. Knows how to develop documentation for a variety of products.											
P. Demonstrates knowledge of technology's historical and future impact on society.											
Competency 002: *The computer science teacher knows how to use technology tools to solve problems, evaluate results and communicate information in a variety of formats for various audiences.*											
A. Knows how to plan, create, edit, analyze and represent data in documents using general productivity software.											
B. Knows how to explore complex concepts using simulations, models, interactive virtual environments and new technologies to develop hypotheses, modify input and analyze results.											
C. Demonstrates knowledge of how to design and implement procedures to track trends, set timelines and evaluate the progress of products using project management tools for continual improvement in process and product development.											
D. Knows how to evaluate projects for design, purpose, audience and content delivery using various criteria (e.g., project specifications, rubrics).											
E. Knows how to select representative products to be collected and stored in an electronic evaluation tool and to evaluate products for relevance to the assignment or task.											

	Required Course Numbers										
Test Content Categories											
F. Knows how to plan and design products that are accessible to learners with various needs and abilities.											
Competency 003: *The computer science teacher knows how to plan, organize, deliver and evaluate instruction that effectively utilizes current technology for teaching the Technology Applications Texas Essential Knowledge and Skills (TEKS) to all students*.											
A. Knows how to implement grade-level appropriate instructional practices, activities and materials to improve student learning.											
B. Knows how to implement lessons using various instructional strategies.											
C. Demonstrates knowledge of issues related to the balanced use of technology for diverse populations.											
D. Knows how to implement instruction that allows students to solve problems by posing questions, collecting data and interpreting results.											
E. Knows how to develop and facilitate collaborative tasks among group members, incorporating various perspectives while exploring alternative solutions.											
F. Knows strategies to help students learn how to locate, retrieve, analyze, evaluate, communicate and retain content-related information from a variety of texts and digital sources.											

	Required Course Numbers										
Test Content Categories											
G. Knows how to evaluate student projects and portfolios using various assessment methods (e.g., formal, informal).											
H. Knows how to promote effective self-evaluation and use of feedback from peers.											
I. Knows the relationship between instruction and assessment.											
J. Knows how to adjust instruction based on assessment results.											
K. Demonstrates knowledge of emerging technology and its role in education.											
L. Knows the importance of self-assessment and planning for professional growth.											
Domain II — Program Design and Development											
Competency 004: *The computer science teacher knows problem-solving strategies and different procedures for program design*.											
A. Exhibits knowledge of all phases of the software system life cycle and understands its cyclical nature.											
B. Knows the characteristics of programming design strategies (e.g., design specification, top-down design, step-wise refinement, black box, object-oriented design).											
C. Knows how to apply problem-solving strategies to implement design.											

	Required Course Numbers										
Test Content Categories											
D. Demonstrates the use of visual organizers (e.g., flow diagrams, Unified Modeling Language [UML]) to document program designs and implementations.											
E. Knows how to create robust programs with emphasis on design to facilitate maintenance, program expansion, reliability, validity and efficiency.											
Competency 005: *The computer science teacher knows procedures for software development and implementation*.											
A. Knows the characteristics of models used in the development of software systems.											
B. Demonstrates the ability to use an integrated development environment (IDE).											
C. Demonstrates knowledge of collaborative strategies for the development of complex software systems (e.g., design/ implementation teams, software validation/testing, risk assessment).											
D. Demonstrates the ability to work independently or collaboratively to implement a solution to a problem according to design specifications, identifying data types, objects needed, subtasks to be performed and reusable components from existing code.											
E. Demonstrates the use of programming style conventions (e.g., spacing, indentation, descriptive identifiers, comments, documentation, standardized programming style) to enhance the readability and functionality of code.											

	Required Course Numbers										
Test Content Categories											
F. Knows how to create robust programs with emphasis on programming style to facilitate maintenance, program expansion, reliability, validity and efficiency.											
G. Knows how to create and use libraries of generic modular code for efficient programming.											
H. Demonstrates the ability to read, evaluate, correct and improve existing code.											
I. Knows how to create robust programs by avoiding runtime errors and handling anticipated errors (e.g., correct handling of input and output, division by zero, type mismatch).											
J. Demonstrates the ability to test programs by entering valid and invalid data; investigating boundary conditions; testing classes, methods and libraries in isolation; and performing stepwise refinement.											
K. Demonstrates the ability to debug program errors (e.g., syntax, runtime, logic) using error messages, reference materials, language documentation and other effective strategies.											
Competency 006: *The computer science teacher knows computer science terminology and concepts and the characteristics of different programming languages and paradigms*.											
A. Knows fundamental computer science vocabulary, including terms related to hardware, software and computational thinking.											

	Required Course Numbers										
Test Content Categories											
B. Knows specific programming terminology, including terms related to data type, data structures, algorithms and programming constructs.											
C. Knows the differences between low-level and high-level languages.											
D. Knows the differences between compiled and interpreted languages.											
E. Knows the characteristics of and differences in current programming languages and paradigms (e.g., procedural, object-oriented).											
Domain III — Programming Language Topics											
Competency 007: *The computer science teacher correctly and efficiently uses data types, data structures and functions in the development of code.*											
A. Understands various computer-related number-base systems and uses them to count, convert and perform mathematical operations.											
B. Understands the characteristics of standard data types in current programming languages (e.g., integer, floating point, character, string, Boolean).											
C. Demonstrates the ability to develop code using constants, variables, data structures and appropriate scope (e.g., local, global).											
D. Understands the difference between primitive and referenced data types (e.g., objects, lists).											

	Required Course Numbers										
Test Content Categories											
E. Demonstrates the ability to cast between data types and provide object functionality to primitive data types.											
F. Demonstrates effective use of standard libraries (e.g., math, string) in the development of code.											
G. Demonstrates the ability to create user-defined functions and procedures.											
H. Demonstrates understanding of the difference between parameters that are passed by value or by reference.											
I. Knows how to identify object-oriented data types and how to delineate the advantages and disadvantages of object data.											
J. Demonstrates the ability to process data in one-dimensional and multi-dimensional arrays.											
K. Understands how to implement input/ output processes (e.g., file, keyboard).											
L. Demonstrates the ability to manipulate text using string processing functions (e.g., concatenation, substring, search).											
M. Understands concepts related to the traversal and processing of abstract data types (e.g., stacks, queues, linked lists, trees, graphs).											

	Required Course Numbers										
Test Content Categories											
Competency 008: *The computer science teacher correctly and efficiently uses statements and control structures in the development of code*.											
A. Creates mathematical expressions using arithmetic operators (addition, subtraction, multiplication, division, integer division and modulus division).											
B. Develops programs using standard operators (e.g., arithmetic, relational, logical, assignment), operator precedence and short-circuit evaluation.											
C. Demonstrates an understanding of conditional and iterative control structures.											
D. Demonstrates coding proficiency in current programming languages, including an object-oriented language.											
E. Demonstrates understanding of object-oriented design and the relationships (including composition and inheritance) among defined classes, abstract classes and interfaces.											
F. Designs classes that encapsulate data and related methods.											
G. Demonstrates understanding of polymorphism in overloading and overriding features of classes.											
H. Demonstrates the ability to use abstract classes and interfaces to design and implement multi-class programs.											

	Required Course Numbers										
Test Content Categories											
Competency 009: *The computer science teacher knows how to construct, compare and analyze various algorithms*.											
A. Constructs searching algorithms (e.g., linear and binary searches).											
B. Constructs sorting algorithms including, but not limited to, selection, insertion, merge and quick sorts.											
C. Analyzes the best-, average- and worst-case run-time efficiencies of various algorithms using informal comparisons and Big-O notation.											
D. Traces, compares and uses iterative and recursive algorithms.											
E. Demonstrates an understanding of and the ability to develop common algorithms to solve practical problems.											
Domain IV — Specialized Topics											
Competency 010: *The computer science teacher knows discrete mathematics topics relevant to computer science*.											
A. Demonstrates knowledge of terminology and the appropriate application of sets, functions and relations.											
B. Constructs truth tables (for negation, conjunction, disjunction, implication, biconditional, bit operators) and uses them to demonstrate propositional relations.											
C. Converts spoken language statements to appropriate statements in propositional logic.											

	Required Course Numbers										
Test Content Categories											
D. Demonstrates proficiency in the use of Boolean algebra, including De Morgan's laws, to identify propositional equivalences.											
E. Uses formal logic proofs and logical reasoning to solve problems and evaluate algorithmic complexity.											
F. Computes permutations and combinations of a set and interprets the meaning in context.											
G. Knows how to exhibit, describe and justify mathematical ideas and arguments through the use of precise mathematical language in written or oral communication.											
Competency 011: *The computer science teacher knows digital forensics topics*.											
A. Understands that digital forensics involves the recovery and investigation of material found in digital devices, often in relation to computer crime.											
B. Demonstrates knowledge of legal, illegal, ethical and unethical information-gathering methods; possible gray areas; and ways in which developing laws and guidelines affect digital forensics practices.											
C. Understands that digital forensics involves the application of tools in a variety of investigations related to malicious attacks (e.g., worm infections, malware, phishing incidents, viruses, Trojans, rootkits, email threats).											

	Required Course Numbers										
Test Content Categories											
D. Identifies and describes businesses and government agencies that use digital forensics.											
E. Demonstrates knowledge of how digital forensics fits in the workplace and how to establish guidelines, procedures and recommendations for the use of digital forensics tools.											
F. Knows how to describe the function and use of digital forensics toolkits in the analysis of network traffic data and data files from various storage media.											
Competency 012: *The computer science teacher knows robotics topics*.											
A. Demonstrates knowledge of technology concepts, systems and operations as they apply to robotics.											
B. Knows how to utilize the design process to prototype, construct, evaluate, refine and document the development of a robot.											
C. Demonstrates the use of computers to manipulate a robot.											
D. Develops algorithms to move and provide interaction with a robot, including applying instructions, collecting sensor data and performing simple tasks (e.g., following lines, moving objects, avoiding obstacles).											
E. Knows how to explore the effects robots have on society.											

	Required Course Numbers										
Test Content Categories											
F. Knows how to use software applications to simulate the behavior of robots of varying complexity, present design concepts and test solution strategies.											
Competency 013: *The computer science teacher knows game and mobile application development topics*.											
A. Demonstrates an understanding of the software-development process specifically applied to mobile and desktop game applications.											
B. Demonstrates knowledge of the basic game design process and elements.											
C. Knows the fundamentals of game art, including the look and feel, graphics coordinate system, basics of color theory and image rendering.											
D. Demonstrates an understanding of the user experience and knows how to create effective user interfaces, game rules and instructions.											
E. Knows how to use board games to research and collect game-play data.											
F. Demonstrates an understanding of game programming essentials, including event-driven programming and collision detection.											
G. Knows how to use a simulation tool to imitate a mobile device's functionality.											