CIS Standards

(Updated November 2016)

Computer Science A

for all Programs of Study

Please select the Computer Science A standards that match the program of study to make sure you have the correct standard reference number.

Computer Science A

Course Description:
Computer Science A focuses on further developing computational thinking skills through app development for mobile platforms. The course utilizes industry-standard tools. Students collaborate to create original solutions to problems of their own choosing by designing and implementing user interfaces and Web-based databases.

Course Code: 270701

Endorsements to Teach:
IT, Math

Programs of Study to which this Course applies:
Computer Science, Software Development

CIS. HS. 8. 1

Recognize and define computational problems.

CIS. HS. 8. 1. l	Provide examples of computationally solvable problems and difficult-to-solve problems.
CIS. HS. 8. 1. m	Decompose a large-scale computational problem by identifying generalizable patterns and applying them in a solution.
CIS. HS. 8. 1. n	Define Big-Oh notation and identify the worst-case complexity class for common algorithms.

CIS. HS. 8. 2

Develop and use abstractions in computational artifacts.

CIS. HS. 8. 2. m	Critically analyze and implement classic algorithms (e.g., sorting, searching) and use them in different contexts, adapting as appropriate.
CIS. HS. 8. 2. n	Evaluate procedural abstractions in terms of their efficiency, correctness, and clarity.
CIS. HS. 8. 2. o	Compare and contrast the list and array of data structures, and justify which is appropriate for a given problem.
CIS. HS. 8. 2. p	Create solutions using standard language-specific library classes identified in the AP Language subset.
CIS. HS. 8. 2. q	Select appropriate data types for variables based on the needs of the problem.
CIS. HS. 8. 2. r	Manage numeric data types in calculations to account for floating point error and loss of precision.
CIS. HS. 8. 2. s	Define basic object-oriented concepts of encapsulation and information hiding and provide rationale for their use.
CIS. HS. 8. 2. t	Employ object-oriented design in the implementation of programs containing multiple student-designed object types.
CIS. HS. 8. 2. u	Define the concepts of abstract classes, interfaces, inheritance, and polymorphism, and provide an example of how they are used to manage complexity.

CIS. HS. 8. 3

Create computational artifacts.

CIS. HS. 8. 3. k	Decompose a problem by creating new data types, functions, or classes.
CIS. HS. 8. 3. l	Demonstrate code reuse by creating programming solutions using libraries and APIs (e.g., graphics libraries, maps API).
CIS. HS. 8. 3. m	Develop programs using the AP language subset of statements, data types, procedures, etc.
CIS. HS. 8. 3. n	Write programs that organize data in lists, arrays, and multidimensional arrays in order to solve a real-world problem.
CIS. HS. 8. 3. o	Integrate grade-level appropriate mathematical techniques, concepts, and processes in the creation of computing artifacts.
CIS. HS. 8. 3. p	Store data in multiple variables and nested structures based on user input and program specifications.
CIS. HS. 8. 3. q	Design and use a file format to share persistent data between program instances.

CIS. HS. 8. 4

Use data to understand and model real-world situations.

CIS. HS. 8. 4. l	Extract relevant information from a string of text using parsing techniques within a program.
CIS. HS. 8. 4. m	Convert extracted data to the appropriate data type for computation or storage.
CIS. HS. 8. 4. n	Implement techniques of searching and sorting data gathered from users or data streams.
CIS. HS. 8. 4. o	Describe a basic computer simulation technique and its implementation.
CIS. HS. 8. 4. p	Devise an algorithm that models a real-world phenomenon and implement it in code.
CIS. HS. 8. 4. q	Evaluate the ability of a computational model or simulations to formulate, refine, and test hypotheses.
CIS. HS. 8. 4. r	Write a program that uses data analysis techniques to identify significant patterns in complex systems.

Computer Science A

Course Description:
Computer Science A focuses on further developing computational thinking skills through app development for mobile platforms. The course utilizes industry-standard tools. Students collaborate to create original solutions to problems of their own choosing by designing and implementing user interfaces and Web-based databases.

Course Code: 270701

Endorsements to Teach:
IT, Math

Programs of Study to which this Course applies:
Computer Science, Software Development

CIS. HS. 9. 1

Recognize and define computational problems.

CIS. HS. 9. 1. l	Provide examples of computationally solvable problems and difficult-to-solve problems.
CIS. HS. 9. 1. m	Decompose a large-scale computational problem by identifying generalizable patterns and applying them in a solution.
CIS. HS. 9. 1. n	Define Big-Oh notation and identify the worst-case complexity class for common algorithms.

CIS. HS. 9. 2

Develop and use abstractions in computational artifacts.

CIS. HS. 9. 2. m	Critically analyze and implement classic algorithms (e.g., sorting, searching) and use them in different contexts, adapting as appropriate.
CIS. HS. 9. 2. n	Evaluate procedural abstractions in terms of their efficiency, correctness, and clarity.
CIS. HS. 9. 2. o	Compare and contrast the list and array of data structures, and justify which is appropriate for a given problem.
CIS. HS. 9. 2. p	Create solutions using standard language-specific library classes identified in the AP Language subset.
CIS. HS. 9. 2. q	Select appropriate data types for variables based on the needs of the problem.
CIS. HS. 9. 2. r	Manage numeric data types in calculations to account for floating point error and loss of precision.
CIS. HS. 9. 2. s	Define basic object-oriented concepts of encapsulation and information hiding and provide rationale for their use.
CIS. HS. 9. 2. t	Employ object-oriented design in the implementation of programs containing multiple student-designed object types.
CIS. HS. 9. 2. u	Define the concepts of abstract classes, interfaces, inheritance, and polymorphism, and provide an example of how they are used to manage complexity.

CIS. HS. 9. 3

Create computational artifacts.

CIS. HS. 9. 3. k	Decompose a problem by creating new data types, functions, or classes.
CIS. HS. 9. 3. l	Demonstrate code reuse by creating programming solutions using libraries and APIs (e.g., graphics libraries, maps API).
CIS. HS. 9. 3. m	Develop programs using the AP language subset of statements, data types, procedures, etc.
CIS. HS. 9. 3. n	Write programs that organize data in lists, arrays, and multidimensional arrays in order to solve a real-world problem.
CIS. HS. 9. 3. o	Integrate grade-level appropriate mathematical techniques, concepts, and processes in the creation of computing artifacts.
CIS. HS. 9. 3. p	Store data in multiple variables and nested structures based on user input and program specifications.
CIS. HS. 9. 3. q	Design and use a file format to share persistent data between program instances.

CIS. HS. 9. 4

Use data to understand and model real-world situations.

CIS. HS. 9. 4. l	Extract relevant information from a string of text using parsing techniques within a program.
CIS. HS. 9. 4. m	Convert extracted data to the appropriate data type for computation or storage.
CIS. HS. 9. 4. n	Implement techniques of searching and sorting data gathered from users or data streams.
CIS. HS. 9. 4. o	Describe a basic computer simulation technique and its implementation.
CIS. HS. 9. 4. p	Devise an algorithm that models a real-world phenomenon and implement it in code.
CIS. HS. 9. 4. q	Evaluate the ability of a computational model or simulations to formulate, refine, and test hypotheses.
CIS. HS. 9. 4. r	Write a program that uses data analysis techniques to identify significant patterns in complex systems.