CIS Standards

(Updated November 2016)

Foundations of Computing

for all Programs of Study

Please select the Foundations of Computing standards that match the program of study to make sure you have the correct standard reference number.

Foundations of Computing

Course Description:

Designed to be the first computer science course for students who have never programmed before, Foundations of Computing is a starting point for Computer Science. Students will explore the impact of computing in society and build skills in digital citizenship and cybersecurity. Beyond learning the fundamentals of programming, students build computational-thinking skills by applying computer science to collaboration tools, modeling and simulation, and data analysis.

Course Code: 270704

Endorsements to Teach:

IT, Math

Programs of Study to which this Course applies:

Information Technology, Computer Science

CIS. HS. 7. 21

Recognize and define computational problems.

CIS. HS. 7. 21. a Given a simple problem, identify its component parts or subproblems.
CIS. HS. 7. 21. b Independently solve subproblems that make up a larger computational problem.
CIS. HS. 7. 21. c Explain how solutions to multiple subproblems work together to solve a larger problem.
CIS. HS. 7. 21. d Provide a precise definition for the term algorithm and explain its relationship to computational solutions.

CIS. HS. 7. 22

Develop and use abstractions in computational artifacts.

CIS. HS. 7. 22. a Define abstraction in terms of computer science and provide an example of how abstraction is used to manage complexity in daily life.
CIS. HS. 7. 22. b Represent equivalent data using different encoding schemes (e.g., binary, unicode, Morse code, student-created codes).
CIS. HS. 7. 22. c Given an abstraction and its documentation, use it effectively in a novel student-created artifact to manage complexity or avoid duplication of effort.
CIS. HS. 7. 22. d Extend or modify an existing abstraction to create new functionality.
CIS. HS. 7. 22. e Identify repetitive elements of program code and develop functionally equivalent versions that reduce redundant code or hide complexity of a task.

CIS. HS. 7. 23

Create computational artifacts.

CIS. HS. 7. 23. a Develop programs that use sequences of statements, loops, and conditional statements.
CIS. HS. 7. 23. b Create variables to store data in a program.
CIS. HS. 7. 23. c Use and update data stored in variables.
CIS. HS. 7. 23. d Design and develop computational artifacts that address personally or socially-relevant concerns.

CIS. HS. 7. 24

Use data to understand and model real-world situations.

CIS. HS. 7. 24. a Use computational tools (e.g., apps, spreadsheets) to investigate a phenomenon within a moderately-sized dataset (50-1000 data points).
CIS. HS. 7. 24. b Filter or transform data using a computational tool.
CIS. HS. 7. 24. c Explain results of a data-driven investigation and a reproducible process for computing the results.
CIS. HS. 7. 24. d Use an existing interactive computer simulation to understand a real-world system (e.g., ecosystem, epidemic, power grid).
CIS. HS. 7. 24. e Adjust inputs to an existing simulation to gain additional insights about a real-world system.

CIS. HS. 7. 25

Test and iteratively refine computational solutions.

CIS. HS. 7. 25. a Describe an iterative design process used in creating computational artifacts (e.g., Polya's problem solving process, Agile development, user-centered design).
CIS. HS. 7. 25. b Apply an iterative design process to solve problems, both independently and collaboratively.
CIS. HS. 7. 25. c Locate and diagnose errors in program code.
CIS. HS. 7. 25. d Correct errors in program code.

Foundations of Computing

Course Description:
Designed to be the first computer science course for students who have never programmed before, Foundations of Computing is a starting point for Computer Science. Students will explore the impact of computing in society and build skills in digital citizenship and cybersecurity. Beyond learning the fundamentals of programming, students build computational-thinking skills by applying computer science to collaboration tools, modeling and simulation, and data analysis.

Course Code: 270704

Endorsements to Teach:
IT, Math

Programs of Study to which this Course applies:
Information Technology, Computer Science

CIS. HS. 8. 1

Recognize and define computational problems.

CIS. HS. 8. 1. a Given a simple problem, identify its component parts or subproblems.
CIS. HS. 8. 1. b Independently solve subproblems that make up a larger computational problem.
CIS. HS. 8. 1. c Explain how solutions to multiple subproblems work together to solve a larger problem.
CIS. HS. 8. 1. d Provide a precise definition for the term algorithm and explain its relationship to computational solutions.

CIS. HS. 8. 2

Develop and use abstractions in computational artifacts.

CIS. HS. 8. 2. a Define abstraction in terms of computer science and provide an example of how abstraction is used to manage complexity in daily life.
CIS. HS. 8. 2. b Represent equivalent data using different encoding schemes (e.g., binary, unicode, Morse code, student-created codes).
CIS. HS. 8. 2. c Given an abstraction and its documentation, use it effectively in a novel student-created artifact to manage complexity or avoid duplication of effort.
CIS. HS. 8. 2. d Extend or modify an existing abstraction to create new functionality.
CIS. HS. 8. 2. e Identify repetitive elements of program code and develop functionally equivalent versions that reduce redundant code or hide complexity of a task.

CIS. HS. 8. 3

Create computational artifacts.

CIS. HS. 8. 3. a Develop programs that use sequences of statements, loops, and conditional statements.
CIS. HS. 8. 3. b Create variables to store data in a program.
CIS. HS. 8. 3. c Use and update data stored in variables.
CIS. HS. 8. 3. d Design and develop computational artifacts that address personally or socially-relevant concerns.

CIS. HS. 8. 4

Use data to understand and model real-world situations.

CIS. HS. 8. 4. a Use computational tools (e.g., apps, spreadsheets) to investigate a phenomenon within a moderately-sized dataset (50-1000 data points).
CIS. HS. 8. 4. b Filter or transform data using a computational tool.
CIS. HS. 8. 4. c Explain results of a data-driven investigation and a reproducible process for computing the results.
CIS. HS. 8. 4. d Use an existing interactive computer simulation to understand a real-world system (e.g., ecosystem, epidemic, power grid).
CIS. HS. 8. 4. e Adjust inputs to an existing simulation to gain additional insights about a real-world system.

CIS. HS. 8. 5

Test and iteratively refine computational solutions.

CIS. HS. 8. 5. a Describe an iterative design process used in creating computational artifacts (e.g., Polya's problem solving process, Agile development, user-centered design).
CIS. HS. 8. 5. b Apply an iterative design process to solve problems, both independently and collaboratively.
CIS. HS. 8. 5. c Locate and diagnose errors in program code.
CIS. HS. 8. 5. d Correct errors in program code.