Virginia Computer Science

Programming (9-12)

Adopted 2017

Cybersecurity

1. **The student will describe and use best practices of program development that make some common flaws less likely and explain how this improves computer security.** PRG.1

Data and Analysis

2. **The student will create programs that model the relationships among different elements in collections of real-world data.** PRG.2
3. **The student will translate numbers between machine representations and human-accessible representations.** PRG.3

Algorithms and Programming

4. **The student will design and implement a program working individually and in teams using a text-based language.** PRG.4
5. **The student will explain the software life cycle and how it applies to iterative development processes.** PRG.5
6. **The student will design and implement an algorithm** PRG.6
   a. with compound conditional execution, and analyze and evaluate complex Boolean conditions; and PRG.6.A
   b. using complex iteration, including nested loops. PRG.6.B
7. **The student will implement programs that accept input from a variety of sources and produce output based on that input.** PRG.7
8. **The student will trace the execution of iterative and recursive algorithms, illustrating output and changes in values of named variables.** PRG.8
9. **The student will perform complex computations** PRG.9
   a. on numbers, including modular division and random number generation; and PRG.9.A
   b. on strings, including substring manipulation and processing individual characters. PRG.9.B
10. **The student will demonstrate an understanding of different data types by using appropriate constructs to convert between them when appropriate.** PRG.10

	11. **The student will analyze a large-scale computational problem, identify generalizable patterns, and implement a solution.** PRG.11
	12. **The student will implement an algorithm that uses existing functions and accesses existing libraries or APIs to satisfy its requirements.** PRG.12
	13. **The student will write functions, both with and without parameters, and both with and without return values, that represent abstractions useful to the solution of a larger problem.** PRG.13
	14. **The student will create programs demonstrating an understanding of the interactions between classes in object-oriented design, and by implementing classes with instance data and methods to satisfy a design specification.** PRG.14
	15. **The student will use code written by others by reading the documentation and incorporating it into their programs using proper citation of the reused code.** PRG.15
	16. **The student will read and store data in 1D and 2D collections, and design and implement algorithms to process and manipulate those collections.** PRG.16
	17. **The student will adapt classic algorithms for use in a particular context and analyze them for effectiveness and efficiency.** PRG.17
	18. **The student will develop and use a series of test cases to verify that a program performs according to its design specifications, including edge cases and all branches.** PRG.18
	19. **The student will, through the process of code review, evaluate a program's correctness, readability, usability, and other factors.** PRG.19
	20. **The student will use a systematic approach** PRG.20
Impacts of Computing	21. **The student will identify some of the practical, business, and ethical impacts of open source and free software and the widespread access they provide.** PRG.21