Washington Computer Science

Grades 3, 4, 5

Adopted 2016

Algorithms and Programming

1B-A-2-1. Apply collaboration strategies to support problem solving within the design cycle of a program. 1B-A-2-1

1B-A-7-2. Use proper citations and document when ideas are borrowed and changed for their own use (e.g., using pictures created by others, using music created by others, remixing programming projects). 1B-A-7-2

1B-A-5-3. Create a plan as part of the iterative design process, both independently and with diverse collaborative teams (e.g., storyboard, flowchart, pseudo-code, story map). 1B-A-5-3

1B-A-5-4. Construct programs, in order to solve a problem or for creative expression, that include sequencing, events, loops, conditionals, parallelism, and variables, using a block-based visual programming language or text-based language, both independently and collaboratively (e.g., pair programming). 1B-A-5-4

1B-A-5-5. Use mathematical operations to change a value stored in a variable. 1B-A-5-5

1B-A-3-6. Decompose (break down) a larger problem into smaller sub-problems, independently or in a collaborative group. 1B-A-3-6

1B-A-3-7. Construct and execute an algorithm (set of step-by-step instructions) that includes sequencing, loops, and conditionals to accomplish a task, both independently and collaboratively, with or without a computing device. 1B-A-3-7

1B-A-6-8. Analyze and debug (fix) an algorithm that includes sequencing, events, loops, conditionals, parallelism, and variables. 1B-A-6-8

Computing Systems

1B-C-7-9. Model how a computer system works. [Clarification: Only includes basic elements of a computer system, such as input, output, processor, sensors, and storage.] 1B-C-7-9

1B-C-7-10. Use appropriate terminology in naming internal and external components of computing devices and describing their relationships, capabilities, and limitations. 1B-C-7-10

	1B-C-6-11.	**Identify, using accurate terminology, simple hardware and software problems that may occur during use, and apply strategies for solving problems (e.g., reboot device, check for power, check network availability, close and reopen app).** 1B-C-6-11
Data and Analysis	**1B-D-5-12.**	**Create a computational artifact to model the attributes and behaviors associated with a concept (e.g., solar system, life cycle of a plant).** 1B-D-5-12
	1B-D-5-13.	**Answer a question by using a computer to (e.g., sort, total and/or average, chart, graph) and analyze data that has been collected by the class or student.** 1B-D-5-13
	1B-D-4-14.	**Use numeric values to represent non-numeric ideas in the computer (binary, ASCII, pixel attributes such as RGB).** 1B-D-4-14
Impacts of Computing	**1B-I-7-15.**	**Evaluate and describe the positive and negative impacts of the pervasiveness of computers and computing in daily life (e.g., downloading videos and audio files, electronic appliances, wireless Internet, mobile computing devices, GPS systems, wearable computing).** 1B-I-7-15
	1B-I-7-16.	**Generate examples of how computing can affect society, and also how societal values can shape computing choices.** 1B-I-7-16
	1B-I-1-17.	**Seek out and compare diverse perspectives, synchronously or asynchronously, to improve a project.** 1B-I-1-17
Networks and the Internet	**1B-I-1-18.**	**Brainstorm ways in which computing devices could be made more accessible to all users.** 1B-I-1-18
	1B-I-1-19.	**Explain problems that relate to using computing devices and networks (e.g., logging out to deter others from using your account, cyberbullying, privacy of personal information, and ownership).** 1B-I-1-19
	1B-N-7-20.	**Create examples of strong passwords, explain why strong passwords should be used, and demonstrate proper use and protection of personal passwords.** 1B-N-7-20
	1B-N-4-21.	**Model how a device on a network sends a message from one device (sender) to another (receiver) while following specific rules.** 1B-N-4-21