Washington Computer Science

# Level 2: 6-8

| | | |
|---|---|---|
| Computing Systems | 1 | Recommend improvements to the design of computing devices, based on an analysis of how users interact with the devices. 2-CS-01 |
| | 2 | Design projects that combine hardware and software components to collect and exchange data. 2-CS-02 |
| | 3 | Systematically identify and fix problems with computing devices and their components. 2-CS-03 |
| Networks and the Internet | 4 | Model the role of protocols in transmitting data across networks and the Internet. 2-NI-04 |
| | 5 | Explain how physical and digital security measures protect electronic information. 2-NI-05 |
| | 6 | Apply multiple methods of encryption to model the secure transmission of information. 2-NI-06 |
| Data and Analysis | 7 | Represent data using multiple encoding schemes. 2-DA-07 |
| | 8 | Collect data using computational tools and transform the data to make it more useful and reliable. 2-DA-08 |
| | 9 | Refine computational models based on the data they have generated. 2-DA-09 |
| Algorithms and and Programming | 10 | Use flowcharts and/or pseudocode to address complex problems as algorithms. 2-AP-10 |
| | 11 | Create clearly named variables that represent different data types and perform operations on their values. 2-AP-11 |
| | 12 | Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals 2-AP-12 |
| | 13 | Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs. 2-AP-13 |
| | 14 | Create procedures with parameters to organize code and make it easier to reuse. 2-AP-14 |

| | |
|---|---|
| | **15** Seek and incorporate feedback from team members and users to refine a solution that meets user needs. 2-AP-15 |
| | **16** Incorporate existing code, media, and libraries into original programs, and give attribution. 2-AP-16 |
| | **17** Systematically test and refine programs using a range of test cases. 2-AP-17 |
| | **18** Distribute tasks and maintain a project timeline when collaboratively developing computational artifacts. 2-AP-18 |
| | **19** Document programs in order to make them easier to follow, test, and debug. 2-AP-19 |
| **Impacts of Computing** | **20** Compare tradeoffs associated with computing technologies that affect people's everyday activities and career options. 2-IC-20 |
| | **21** Discuss issues of bias and accessibility in the design of existing technologies. 2-IC-21 |
| | **22** Collaborate with many contributors through strategies such as crowdsourcing or surveys when creating a computational artifact. 2-IC-22 |
| | **23** Describe tradeoffs between allowing information to be public and keeping information private and secure. 2-IC-23 |