

# By the End of the 10th Grade

## Computing Systems

### Devices

- 1 Explain how abstractions hide the underlying implementation details of computing systems embedded in everyday objects. [9-10.CS.D.01](#)
- 

### Hardware & Software

- 1 Explain the abstraction and interactions between application software, system software and hardware. [9-10.CS.HS.01](#)
- 

### Troubleshooting

- 1 Develop, communicate and apply systematic troubleshooting strategies for correction of errors in computing systems. [9-10.CS.T.01](#)
- 

## Networks & the Internet

### Network Communication & Organization

- 1 Evaluate the scalability and reliability of networks by identifying and illustrating the basic components of computer networks (e.g., routers, switches, servers) and network protocols (e.g., IP, DNS). [9-10.NI.NCO.01](#)
  - 2 Describe the issues that impact network functionality (e.g., bandwidth, load, delay, topology). [9-10.NI.NCO.02](#)
- 

### Cybersecurity

- 1 Compare physical and cybersecurity measures by evaluating trade-offs between the usability and security of a computing system. [9-10.NI.C.01](#)
  - 2 Illustrate how sensitive data can be affected by attacks. [9-10.NI.C.02](#)
  - 3 Recommend security measures to address various scenarios based on information security principles. [9-10.NI.C.03](#)
  - 4 Explain trade-offs when selecting and implementing cybersecurity recommendations from multiple perspectives such as the user, enterprise and government. [9-10.NI.C.04](#)
- 

## Data & Analysis

### Storage

- 1 Translate and compare different bit representations of data types, such as characters, numbers and images. [9-10.DA.S.01](#)
- 2 Evaluate the trade-offs in how data is organized and stored digitally. [9-10.DA.S.02](#)

---

## Collection, Visualization & Transformation

- 1 Create data visualizations to help others better understand real-world phenomena. [9-10.DA.CVT.01](#)
- 2 Explain the insights and knowledge gained from digitally processed data by using appropriate visualizations, notions and precise language. [9-10.DA.CVT.02](#)
- 3 Evaluate and refine computational artifacts to make them more usable and accessible. [9-10.DA.CVT.03](#)

---

## Inference & Models

- 1 Show the relationships between collected data elements using computational models. [9-10.DA.IM.01](#)
- 2 Refine computational models to better represent the relationships among different elements of data collected from a phenomenon or process. [9-10.DA.IM.02](#)

---

## Algorithms & Programming

### Algorithms

- 1 Create a prototype that uses algorithms (e.g., searching, sorting, finding shortest distance) to provide a possible solution for a real-world problem. [9-10.AP.A.01](#)

---

### Variables

- 1 Create problem solutions that utilize primitive variables (e.g., strings, ints, Booleans, doubles). [9-10.AP.V.01](#)
- 2 Demonstrate the use of advanced variables (e.g., lists, arrays, objects) to simplify solutions, generalizing computational problems instead of repeatedly using primitive variables. [9-10.AP.V.02](#)

---

### Control

- 1 Apply the concepts of specific control structures (e.g., sequence, conditionals, repetition, procedures) considering program efficiencies such as readability, performance and memory usage. [9-10.AP.C.01](#)

---

### Modularity

- 1 Break down a solution into procedures using systematic analysis and design utilizing functional abstraction. [9-10.AP.M.01](#)
- 2 Create computational artifacts (file, graphic, video, audio) by systematically organizing, manipulating and/or processing data. [9-10.AP.M.02](#)

---

## Program Development

- 1 Using visual aids and documentation, illustrate the design elements and data flow (e.g., flowcharts, pseudocode) of the development of a program. [9-10.AP.PD.01](#)
- 2 Create a program by analyzing a problem and/or process, developing and documenting a solution, testing outcomes, debugging errors and adapting the program for a variety of users. [9-10.AP.PD.02](#)
- 3 While collaborating in a team, develop, test and refine programs that solve practical problems or allow self-expression. [9-10.AP.PD.03](#)
- 4 Evaluate and refine computational artifacts to make them more user-friendly, efficient and/or accessible. [9-10.AP.PD.04](#)

---

## Impacts of Computing

### Culture

- 1 Evaluate the ways computing impacts personal, ethical, social, economic and cultural practices. [9-10.IC.C.01](#)
- 2 Test and refine computational artifacts to reduce bias and equity deficits. [9-10.IC.C.02](#)
- 3 Demonstrate how a given algorithm applies to problems across disciplines. [9-10.IC.C.03](#)

---

### Social Interactions

- 1 Demonstrate through collaboration on a project how computing increases connectivity among people of various cultures. [9-10.IC.SI.01](#)
- 2 Explain how the degrees of communication afforded by computing have impacted the nature and content of career fields. [9-10.IC.SI.02](#)

---

### Safety, Law & Ethics

- 1 Explain the beneficial and harmful effects that intellectual property laws can have on innovation. [9-10.IC.SLE.01](#)
- 2 Explain the privacy concerns related to the collection and analysis of information about individuals that may not be evident to users. [9-10.IC.SLE.02](#)
- 3 Evaluate the social and economic consequences of how law and ethics interact with digital aspects of privacy, data, property, information and identity. [9-10.IC.SLE.03](#)
- 4 Define and classify a variety of software licensing schemes (e.g., open source, freeware, commercial) and discuss the advantages and disadvantages of each scheme in software development. [9-10.IC.SLE.04](#)
- 5 Identify and explain the potential impacts and implications of emerging technologies on larger social economic and political structures with evidence from credible sources. [9-10.IC.SLE.05](#)