A Model Curriculum for K–12 Computer Science

Level I Objectives and Outlines

Daniel Frost
University of California, Irvine
5058 Donald Bren Hall
Irvine, CA 92697-3425
949-824-1588
frost@uci.edu

Anita Verno
Bergen Community College
400 Paramus Road
Paramus, NJ 07652
201-447-7909
averno@bergen.edu

David Burkhart
West Muskingum Middle School
100 Kimes Road
Zanesville, OH 43701
davidlburkhart@gmail.com

Michelle Hutton
The Girls' Middle School
180 N. Rengstorff Ave.
Mountain View, CA 94043
mfh@pobox.com

Karen North
Houston ISD, Piney Point ES
70 Patti Lynn Lane
Houston, TX 77024
knorth@houstonisd.org
knorth@wt.net

Copyright © 2009 by the authors and the Computer Science Teachers Association (CSTA),
2 Penn Plaza, Suite 701, New York, NY 10121-0701.
Permission to make digital or hard copies of portions of this work for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Abstracting with credit is permitted.

Table of Contents

Introduction

Overview

In 2003, ACM (the Association for Computing Machinery) published *A Model Curriculum for K–12 Computer Science*; a second edition was released in 2006. The *Model Curriculum* describes a vision of K–12 computer science education comprising four levels:

I, *Foundations of Computer Science,* recommended for grades K–8;
II, *Computer Science in the Modern World,* recommended for grades 9 or 10;
III, *Computer Science as Analysis and Design,* recommended for grades 10 or 11;
IV, *Topics in Computer Science,* recommended for grades 11 or 12.

The Computer Science Teachers Association (CSTA) has taken the lead in writing implementation documents for the *Model's* four levels. CSTA is a membership organization that supports and promotes the teaching of computer science and other computing disciplines in kindergarten through twelfth grade. CSTA committees produced the Level II implementation document in 2004 and the Level III implementation document in 2007. This current report is an elaboration of the *Model*'s Level I framework.

This report provides the outline and objectives for computer science education in grades K–8, in a manner that will be accessible to all students, regardless of economic status. Unlike the Level II and Level III reports, Level I does not describe a single course. Because Level I covers nine years of education, we have emphasized a broad and integrated set of student learning outcomes, held together by a coherent concept of computer science for younger students. This document does not address specifics such as the order of topic presentation, exercises, or assessment measures. In particular, we have avoided references to specific software systems, programming languages, or hardware platforms. This *Level I Objectives and Outcome*s document can be the starting point for a teacher, school, district, or state to make computer science a vital part of K–8 education.

Purpose and Role of the Level I Objectives and Outlines

The societal impact of computing is undeniable, affecting children even younger than kindergarten age. People interact with a wide variety of computational devices daily, from ATMs, cell phones, and computerized voting booths to desktop computers. The rapid pace of technology change creates a need for students to be taught the underlying principles and concepts upon which digital technology is built. In kindergarten through eighth grade, students can learn about computer science in way that is age-appropriate, engaging, and sets them on a path towards a lifetime of healthy interaction with technology in the digital 21st Century.

Computer science in kindergarten through eighth grades encompasses a wide variety of topics, principles, and skills. The *Model Curriculum* broadly defines computer science as follows:

> *Computer science* (*CS*) is the study of computers and algorithmic processes, including their principles, their hardware and software designs, their applications, and their impact on society.

Following this definition, we see the core of K–8 computer science as having three components:

1. knowledge of the computing environment—computer hardware, software applications, and networked systems—that is an integral part of modern society;
2. a way of thinking that uses computers, information, algorithms, and programming languages as a creative medium for solving problems of all kinds; and
3. an appreciation for the complex and changing interactions between computing, individuals, organizations, and culture.

A student whose education includes the topics and student learning objectives described in this report will understand the principles underlying computing devices and systems, and will be well positioned for further computer science education in high school and college.

About this Document

For the purposes of curriculum exposition, we divide the subject matter of K–8 computer science into twelve topics, which are grouped into three categories:

- Computers and software applications
    - Topic 1: Parts of a computer
    - Topic 2: Standard software
    - Topic 3: Operating systems
    - Topic 4: Computer networks
    - Topic 5: The World Wide Web and communicating over networks
- Problem solving with computer science
    - Topic 6: Representing information digitally
    - Topic 7: Problem solving and algorithms
    - Topic 8: Computer programming
- Social context of computing
    - Topic 9: Privacy and security
    - Topic 10: Evaluating and using information from networked sources
    - Topic 11: Human-computer interaction
    - Topic 12: Computers in society

Each topic presentation has several subsections. The "Topic Description" provides a high-level overview of the topic's major themes. The "Background Information" subsection is a brief survey of the topic, concentrating on aspects relevant to grades K–8. "Materials and Supplies" alerts the teacher to any special requirements for the topic. The Student Learning Objectives and the linked Focus areas and Sample Activities are the heart of this implementation document. They are divided into grade ranges K–2, 3–5, and 6–8, with the material in each range reflecting the growing maturity and intellectual capacities of the students. The grade range assigned to each Objective and Focus area should be considered to be no more than a guideline. Each Topic's Focus areas indicate several elements of instruction that support the grade ranges' Student Learning Objectives. The Focus areas should be thought of as inspirations and suggestions, and not as a fixed agenda. A Sample Activity has been suggested for each of the Focus areas to help make them clear and concrete. These activities should be modified to meet the specific needs of the classroom.

How to Use this Document

This is a thinking person's curriculum. It should not be followed blindly, but used as an outline and guide to create a program of study for each grade level. Each teacher should craft a curriculum to meet the needs of his or her students, covering the material at an appropriate depth and speed. Although each topic is presented independently, many are interdependent. Teachers are encouraged to link topics as appropriate and to present topics in the order that meets the needs of their students. **The ordering of topics in this document is not meant to imply an order of presentation in the classroom.** Similarly, all topics and focus areas do not need to be given equal emphasis. Teachers and school districts should use this Level I curriculum as a starting point to develop lesson plans, exercises, and assessments that will best suit their students. Lesson plans and activities to support each stated objective can be contributed to and accessed from the CSTA Source Web Repository at: http://csta.acm.org/WebRepository/WebRepository.html.

Computer science is inherently an interdisciplinary subject, since almost all fields deal with information, communication, algorithms, and processes. The topics in the Level I curriculum can be taught as part of computer science courses or modules, or they can be integrated into other subjects in the school day.

Thanks and Acknowledgments

Many people assisted the authors and contributed ideas and words to this document. Attendees at the "Making CS Happen in K-8" session of the Computer Science & Information Technology Symposium on June 27, 2009 made several excellent suggestions, for which the authors are grateful. In addition, over fifty people reviewed a draft of this document and submitted comments by email, which led to many improvements. The authors would also like to thank the following contributors:

- Debbie Carter, Computer Science and Math Teacher, Roxbury High School, Roxbury, NJ
- Steve Cooper, Professor of Computer Science, St. Joseph's University, Philadelphia, PA
- Myra Deister, Computer Science Teacher, Sunny Hills High School, Fullerton, CA
- Kathryn Frost, Director of Information Technology, Share Our Selves, Costa Mesa, CA
- Bill Glass, President, Terrapin Software, Cambridge, MA
- John Harrison, Computer Science Teacher, Prince Anne High School, Virginia Beach, VA
- Linda Hoffman, Technology Teacher, Metzger Middle School, San Antonio, TX
- Carolyn Oates, Computing Teacher, St. Pölten, Austria
- Joyce Paisley, Instructional Technology Specialist, Alamance County, NC
- Chris Stephenson, Executive Director, Computer Science Teachers Association
- Hadar Ziv, Lecturer, Dept. of Informatics, University of California, Irvine, CA

Topic 1: Parts of a Computer

Topic Description:
Understanding computer hardware helps students build confidence in their computing skills and enhances their understanding of computer software. Computer hardware systems and computer software systems share many similarities in their structures.

Background Information:
To work effectively, one must understand the physical tool as well as what it can do. Understanding the hardware will help the student understand the role of software. An analogy can be made to building. Most young students have been exposed to (plastic) carpentry tools and have had the opportunity to select and use the tools. Being able to label the tools—hammer, screw driver—and know what they can contribute, helps students use those tools successfully.

Personal computers, desktop computers, laptops, and similar computer systems have multiple hardware components, sometimes packaged in one unit and sometimes consisting of physically separate components. Each part of a PC can be categorized by the role it plays.

- Input – enables the human operator to send information and commands to the computer.
- Output – conveys information to the human operator.
- Memory and storage – retains information in digital form. Main memory is volatile and is reset when power is turned off. Non-volatile storage, such as a hard disk, retains information without power.
- Processor / Central Processing Unit (CPU) – performs arithmetic and control operations.
- Network connection – connects a computer with other computers, either with cables or wirelessly.
- Other components, such as the power supply and case.

Some devices, such as touch screens, fall into more than one category. Computational devices other than personal computers often have similar components, but they may be packaged into a single unit.

Common terminology used to describe computer capacity and performance includes the units hertz and bytes and the quantity prefixes kilo-, mega-, giga- and tera-.

Materials and Supplies:
Personal computers with a word processor or text editor, and a calculator.

Student Learning Objectives	
A K–2 student will be able to:	
1.	Identify the input and output components of a personal computer.
2.	Safely and correctly perform basic operations involving a personal computer.
3.	Use input devices with a personal computer.

A 3–5 student will be able to:	
1.	Identify input, output, storage, and processing components of a personal computer.
2.	Draw a diagram to explain the flow of data through parts of a personal computer.
3.	Safely and correctly connect peripherals to a computer.
4.	Use standard input devices at an age-appropriate level.
A 6–8 student will be able to:	
1.	Evaluate computer components in terms of features and price.
2.	Compare PCs with other electronic devices.
3.	Identify a variety of cables and ports used on PCs

Focus	**Sample Activity**
Grades K–2:	
Vocabulary	The teacher points to a part of a computer and the class says the part's name. Emphasis is on input and output components.
Roles of PC components	The teacher names a part of the computer, and the class says whether it is for input or output (or both).
Starting up and shutting down a PC	Students practice starting, rebooting, and shutting down a computer. If multiple computers are available, the students time the start-up, reboot, and shut-down process on each computer. The teacher leads the students in a discussion of possible reasons for the variation.
Keyboard	Students classify types of keys on the keyboard. They practice using the keyboard by entering lower-case and upper-case text into a text editor or word processor. Students complete a math worksheet. Then they launch a PC calculator. They check their math work by entering numbers using both the standard keyboard and the numeric keypad.

Focus	**Sample Activity**
Mouse	Students use the mouse button to start programs, open folders, and open files. Students start a software application by double-clicking on its icon. They try variations of the time lapse between two clicks to determine those that are recognized as a double-click.
Grades 3–5	
Vocabulary	The teacher points to a part of a computer (or a schematic drawing of a computer) and the class says the part's name. Emphasis is on storage and processing components.
Roles of PC components	The teacher names a part of the computer, and the class says whether it is for storage or processing (or both). Students pass around computer parts such as hard drives and motherboards and draw sketches of what they see.
Evaluating features	For a selected component, students use data from advertisements and web sites to create a table of features for several models or brands.
Volatile and non-volatile memory	Students enter some information into a word processor or text editor, save the information to a non-volatile memory device, enter some more information, and then shut off the computer's power. The computer is turned back on and the students determine what information was lost.
Keyboard and mouse	Students keyboard using correct hand, arm, and body position. Students are given a document in a word processor and practice navigating around the document and selecting text using both mouse and keyboard skills.

Focus	Sample Activity
Grades 6–8	
Roles of components in electronic devices	For a particular electronic device with computational capabilities (e.g. a cell phone), students list components and their roles, identifying those that commonly are found on PCs and those that are not.
Keyboard	Students practice keyboarding until they can use the keyboard with reasonable speed and accuracy, using correct hand, arm, and body positions.
Units of measurement	Students convert amounts from one measurement unit to another. The result may use rounding (for example, convert 6,308 bytes to 6.3 kilobytes; convert 0.6 gigahertz to 600 megahertz).
Cables and ports	Working in small groups, students write a report listing all the different types of cables, ports, and connectors that they find in the school computer lab and at home. The report should include drawings or photographs, the official or technical names, data capacity, and whether or not the connection is powered.
History of computers	Students create a timeline for the development and evolution of computer hardware, possibly including pre-electronic computational devices. The timeline can be augmented, where possible, with examples of earlier hardware, photographs or advertisements, and notes as to cost and capabilities. Challenge students to use their imagination and extend the timeline forward 10 years.

Topic 2: Standard Software

Topic Description:
A wide variety of standard software applications are used by K–8 students in school and at home. Familiarity with applications is important for success in high school, college, and a career and introduces students to many of the foundational ideas of computing.

Background Information:
K–8 students may use a range of software applications, many not designed specifically for K–8 students. Examples include word processors and text editors, presentation creators, spreadsheets, web browsers, email clients, multimedia players, multimedia content creators, computer and video games, search engines, web authoring software, and programming languages and environments

Other types of standard software are designed specifically for educational environments, such as learn to type programs; test taking software; and classroom, homework, and assessment management systems.

Materials and Supplies:
Computers with several types of software applications.

Student Learning Objectives	
A K–2 student will be able to:	
1.	Launch and use specified software to complete an activity.
A 3–5 student will be able to:	
1.	List at least three software titles and give an example of an appropriate use for each.
2.	List similarities and differences in the functions of at least three application programs.
A 6–8 student will be able to:	
1.	List one or more software titles that can be used, given a stated task.
2.	List standard features of software.

Focus	Sample Activity
Grades K–2	
Locating software programs	Students locate and open a specified software application using two different techniques. When appropriate, students identify the correct icon for the application.

Focus	**Sample Activity**
Use of software	Students use a specified software application to complete an assigned task.
Grades 3–5	
Exposure to software applications	Students work in a variety of local and online software applications to complete assigned tasks.
Choosing and using software	Students are instructed to use the computer to complete a given task. The students choose an appropriate software application and use the tool effectively to accomplish the task.
Knowledge of software suites	Students open several applications within a software suite and create a table showing the similarities and differences in the common user interface.
Using built-in help	Students select an unfamiliar feature of a software application and read the associated help text. Students then write a paragraph discussing how helpful they found the help system.
Grades 6–8	
Multiple software applications can be used to complete the same task	Students identify several software applications that can be selected to complete a specified task and then complete the task at least twice, using different applications. Students then write a paragraph comparing the applications and stating their preferences.
Critical thinking about software applications	Students select a familiar software application and write a one page proposal, make a presentation to the class, or create a video advertisement for the next version of the application, listing specific new or changed features that they believe are desirable.

Topic 3: Operating Systems

Topic Description:
An operating system (OS) is a bundle of software that sits between application programs and the computer hardware. This topic focuses on operating systems commonly found on desktop computers.

Background Information:
Important OS tasks include:

- handling files and directories;
- managing external devices attached to the computer;
- providing standard user interface elements, such as buttons, menus, and mouse pointers;
- allowing multiple programs to run on a single computer at the same time;
- providing a collection of standard utility programs.

Materials and Supplies:
Access to computers with any operating system. Ability to access system utilities is helpful.

Student Learning Objectives	
A K–2 student will be able to:	
1.	Create well-named files using standard system software.
2.	Utilize basic OS user interface elements.
A 3–5 student will be able to:	
1.	Perform basic file and directory operations using the OS.
2.	Utilize intermediate OS user interface elements.
3.	Recognize and run a variety of OS utility programs.
A 6–8 student will be able to:	
1.	Describe the role of the OS as an intermediary between application programs and hardware.
2.	Utilize advanced OS user interface elements and features.

Focus	Sample Activity
Grades K–2	
Logging on to the computer	Students learn how to log on and off their school computers, using a network if necessary.

Focus	Sample Activity
Desktop icons and software applications	Students create files of various types and observe the icons that indicate the file type. Students draw (on paper) the icons associated with different software.
Good file names	Student create an appropriate and descriptive name for a file based on the description of the information found within the file.
Standard OS dialogs	Students launch a familiar software application and use the OS browse/save/open dialog box to open a specified file. Students modify, print, and save the file with a new name.
Menus and buttons	Students launch a familiar software application and make selections using menus and buttons within the application.
Grades 3–5	
File name extensions	Students are given a list of common file name extensions. They try to open various file types with different application software and record the result (for example, they open a text file with a graphics application). Students investigate the application software available on the computers and record at least one software application which will correctly open files of each extension type. In classes where the operating system does not require file extensions, modify the assignment to have students match the icon with the application.
Distinguishing proprietary and non-proprietary file types	Students define “proprietary.” Students create a list of proprietary extensions and the applications that can be used to open each file type. They attempt to open a proprietary file with the wrong application. Students create a list of extensions that are not proprietary and practice opening non-proprietary files using various applications.

Focus	**Sample Activity**
Understanding files and directories (folders)	The teacher presents the analogy of computer file, directory or folder, and disk to paper documents, manila folder, and filing cabinet. Students draw a storage scenario (files, directories, disk) for their schoolwork.
Selecting an appropriate place to save a file	Students choose between alternative locations on the hard disk (e.g. the desktop, a personal directory, a temporary directory) when saving a file, and explain their reasons for the choice. Students choose between alternative media (e.g. the computer's hard disk, a removable storage device, a school server, web storage space) when saving a file, and explain their reasons for the choice.
Manipulating windows	Students resize, minimize, maximize and move an application or utility program window. Two or more techniques for performing each operation are attempted.
OS utilities	Students use various OS utilities to complete simple daily tasks, such as using the OS calculator to check their math homework, checking the day of the week for the next holiday or special event, or using a simple text editor to take notes.
Navigating between running applications	Students start several applications running, each with one or more windows (possibly overlapping or obscuring other windows), and use control keys or the mouse to switch between or find applications.
Grades 6–8	
Comparing operating systems	Students research (using the web, magazines, retail stores) more than one OS (for personal computers and electronic devices), and write a list of pros and cons for each one.
Understanding directories	Given a diagram showing a hierarchy of directories and files, students write out the fully qualified names (paths) of files in nested directories.

Focus	**Sample Activity**
Exploring multi-tasking	Students experiment with the effects of running multiple applications, especially CPU-intensive ones, at the same time, and report on the speed and memory impact. They use an operating system utility to observe CPU and memory usage.
Role of the OS	Students describe, with words and diagrams, the role of the OS.
Using the clipboard	Students open a text editor and a word processor, type a two-paragraph message into the text editor, and copy the text into the word processor.

Topic 4: Networks

Topic Description:
Computer networks consist of multiple computers connected to each other so that they can exchange data and share resources.

Background Information:
A network is a group of computers connected together to communicate and to share resources. A network that covers a small area such as a home, office, or school computer lab is a LAN (Local Area Network). A network that includes computers in multiple locations is a WAN (Wide Area Network). The Internet is the world's largest WAN.

Every network of computers has a *topology*, which is the physical layout of communication links between computers on the network. A small number of computers can each be directly connected to all the others, but typically each computer in a network is only connected to a few of the other computers in the network. In this case, information may flow from one computer to another by passing through one or more intermediate computers. The topology of the Internet is complex and ever-changing. Communication between two computers that are physically distant from each other usually passes through a dozen or more intermediate computers.

The Internet is a world-wide system for interconnecting smaller networks. It provides the basis for the World Wide Web, email, FTP (file transfer protocol) and many other communication activities. When a computer is connected to the Internet, it has an IP (Internet Protocol) address, which consists of four numbers between 0 and 255, written like this: 74.125.19.99. The Internet has a system for converting between IP addresses and more readable hostnames. For example, the hostname "www.google.com" corresponds to the IP address above. Data transmitted over the Internet is broken up into packets. Each packet includes the IP addresses of the source and destination computers.

The Internet was developed in the United States in the 1960s and 1970s and was originally called the ARPANET, after the Advanced Projects Research Agency of the Defense Department.

Materials and Supplies:
No special resources required.

Student Learning Objectives	
A K–2 student will be able to:	
1.	Define the term network and identify several networks.
2.	Name a task performed on a computer that requires a network.
A 3–5 student will be able to:	
1.	Give examples of standards and protocols used in a variety of communication and information sharing tasks.

2.	Explain the roles of the client and the server in a client-server architecture.
A 6–8 student will be able to:	
1.	Describe the way data moves over the network.
2	Diagram a stated logical network topology.

Focus	**Sample Activity**
Grades K–2	
Understanding networks	Working in groups, students identify several networks other than networks of computers. Possibilities include social networks, water and gas systems, and television networks. For each kind of network, students describe the network's nodes and links.
Working with local and remote resources	Two tasks are assigned to students, one requiring use of local resources (e.g. the computer's CD/DVD player and disk drive) the other requiring use of remote resources (e.g. a web site). Students determine which activity is using local resources and which is using remote resources. Students then point to the local resources and draw a picture of the remote resources. Student explain the picture and define the words "remote" vs. "local" in the context of computing.
Grades 3–5	
Protocols	Students identify and explain protocols that occur in non-computer contexts. Examples include the direction of letters in text, the location of the table of contents and the index in a book, the organization of the address and return address on an envelope, and various classroom tasks (such as turning in homework). Students suggest alternate protocols, and imagine what would happen if no protocols existed.

<u>**Focus**</u>	<u>**Sample Activity**</u>
Hostnames and network topologies	Students make up hostnames (e.g. boys.com and girls.com) and unique email addresses. Mail messages are written on 3x5 cards. Some students act as mail servers and route all mail to and from their host to other servers.
Network topologies	Students arrange the classroom in different configurations (for example, chairs in groups of four students, all chairs in one line). The class writes a short story, each student or group contributing one paragraph. Students transmit information (the story) through the network by passing their paragraphs with the teacher as the destination. What protocol should be used for one group to get the attention of another group? A discussion ensues as to how to reassemble the paragraphs to create the story.
Client-server architecture	Students list non-computing examples of servers which can supply information resources to clients. Examples include a librarian, a friend with a watch, and a teacher. Students define an informal protocol for making requests of a server and returning information.
Grades 6–8	
Error correction	In small groups, students take on the roles of Internet routers and attached computers, with IP addresses. Each message, written on a 3x5 card, consists of source and destination IP addresses, a short sentence or question, and a number indicating the count of letters and punctuation symbols in the message. Each router flips a coin twice and if it comes up heads both times, changes one letter in the message. Students discuss techniques for detecting these random changes.
Data communication over networks	Arrange students in groups that represent the configuration of several LANs. Ask how data can be communicated within the LAN and from one LAN to another. Have students suggest ways that a LAN and WAN can identify the target of a message.

Focus	**Sample Activity**
Network topologies	Students research and draw graphs of a variety of logical network topologies and networked structures, such as cell phone networks and airline flight route structures.

Topic 5: The World Wide Web and Communicating over Networks

Topic Description:
One of the most important uses of computers and computer networks is to help people communicate and exchange information. The World Wide Web is a system of linked documents that are accessed using the Internet. Documents may be static or dynamic and may provide access to databases of information. Email is a system for sending and receiving messages over the Internet.

Background Information:
The web was invented by Tim Berners-Lee in 1990 and 1991. The World Wide Web has three major components. The user runs a web browser, which formats and displays web pages. Web servers store collections of web pages (called web sites) and data. Users and web servers are connected by the Internet, which passes data between them. A user's browser connects to a web server and requests a web page, which is transmitted back to the user's computer over the Internet. Web pages have links to other web pages; following these links is the primary technique for traversing the web.

Web pages are identified with URLs (Uniform Resource Locaters). A URL consists of a scheme (e.g. http://), a host (e.g. www.google.com), a port (e.g. 80) which is usually omitted, and a path or other page identifying information (e.g. /newuser/login.html).

Search engines are a central part of the web. They consist of "inverted lists," or indices, which associate words with the URLs of web pages on which those words appear. These lists are created and updated by computer programs called "spiders" which read web pages and update the lists.

Communication can occur in asynchronous or synchronous form. A popular example of asynchronous communication is email (electronic mail). An email address consists of a user name (1 to 64 characters long), the @ symbol, and a domain name, such as example.com. Examples of synchronous communication are chat rooms, instant messaging, and telephone calls, where several individuals can communicate in real time. Some guidelines for responsible communication: do not forward an email or message without prior and specific approval from a responsible adult; do not respond to somebody you don't know; never write anything online that you would not want your parents and teachers to read.

Materials and Supplies:
A computer with Internet access, a browser, and a text editor for creating web pages.

Student Learning Objectives
A K–2 student will be able to:
1. Access and use the World Wide Web in an age-appropriate manner.

A 3–5 student will be able to:	
1.	Use the World Wide Web to locate information.
2.	Name and identify the main parts of a URL.
3.	Identify at least ten top level domains (TLDs) and the associated category or country.
4.	Edit a web page template and view it locally in a browser.
5.	Use at least one form of electronic communication in a safe manner.
A 6–8 student will be able to:	
1.	Create a basic web page using HTML and CSS that conforms to standards.
2.	Explain the relationship between a web server, a web page, and a browser.
3.	Use advanced search engine options and refine searches to locate information.
4.	State at least two benefits and one drawback of using networks for communication.

Focus	**Sample Activity**
Grades K–2	
Accessing the WWW	Using a computer connected to the Internet, students start a browser, enter a given URL, and bring up a web page.
Using a web site	Students follow links and click links appropriately within a specified web site.
Grades 3–5	
Finding and saving web data for later use	Students are given a topic and use a web search engine to locate relevant data. Students copy the data and the URL to a text editor or word processor for later use in a research project.
Editing a web page template that includes HTML tags	The teacher provides a template that includes the basic HTML tags (html, head, title, body) as well as a list of a few common tags (ex: h1-h6, p, img, hr). The class storyboards a simple web page and each student creates his/her own version of the page. The students share their pages with one another.

Focus	**Sample Activity**
Search techniques	Using a textbook or other reference book they are not familiar with, students try to locate information using the table of contents, the index, and sequential skimming of all pages. Students report on which technique is most effective, and for what reasons. Students search for the same information using the WWW and discuss how the manual search techniques relate to the electronic techniques.
Top level domains	Students fill in an unlabeled world map with country code TLDs (instead of country names).
Grades 6–8	
Creating a web site that conforms to standards	A class selects a topic. Students are divided into teams, each team responsible for a subtopic. Each student creates one or more web pages for his/her assigned subtopic. The team leader creates a home page for the subtopic. The class creates a home page for the site. As each web page is developed, the student validates it using a validator such as the one available from the World Wide Web Consortium. Example: The topic is animals; subtopics are land, sea, and air animals; each web page represents one animal within the subtype.
Top level domains	In History or Social Studies class, students research how TLDs have changed for countries which have ceased to exist (e.g. Czechoslovakia, USSR).
Social networks	Students work in small groups to compile a list of ways in which social networking is facilitated by computer networks. For each entry on the list, they identify one or more benefits, limitations, and risks.

Topic 6: Representing Information Digitally

Topic Description:
The digital representation of information is a central component of computer science and modern information technology. Digital representation means a computer oriented model of the information, a translation of that information into bits, and a physical representation of those bits.

Background Information:
Computers can input, store, process, and output many kinds of information. Some common types of information are numbers, text, images, music, and video. The information must be converted into a format that can be managed digitally. This is done using a coding scheme where the data is represented as a series of bits. The word "bit" is a contraction of **b**inary dig**it**. A bit has two possible values or states, which are often called 0 and 1. Since a single bit doesn't hold much information, it is common to use eight-bit bytes, each of which can have one of 256 different values. Bits are stored or transmitted in a physical medium, such as a hard disk, DVD, computer memory, or copper wire.

Different types of information are encoded into bits and bytes in different ways. A small integer value can be stored in one byte, which holds a number between 0 and 255. To permit larger integers, more bytes are used. In the ASCII system, letters, digits, punctuation, and some symbols are each assigned to one byte value.

On a computer monitor, each pixel has a color created by combining light from red, green, and blue light emitting units. Each of these units has an intensity that goes from 0 (off) to 128 (half on) to 255 (fully on); this range was selected because a single byte can hold a number between 0 and 255. With three component colors per pixel, each pixel requires three bytes to store its color value—one for the intensity of red, another for the intensity of green, and a third for the intensity of blue.

Audio data is a continuous stream of varying pitch and volume that must be converted to digital format to be processed by a computer. To output the music or sound, the computer's hardware and software re-create the appropriate pitch and volume and play the analog equivalent of the digitized sound.

Materials and Supplies:
No special resources required.

Student Learning Objectives	
A K–2 student will be able to:	
1.	Classify and code objects and text using a specified coding scheme.
A 3–5 student will be able to:	
1.	List two kinds of information that can be stored in a byte.

A 6–8 student will be able to:	
1.	Provide two reasons why compressing a file is useful.
2.	Evaluate a list of items to determine if they are digital.
3.	Use prefixes which indicate quantity.

Focus	**Sample Activity**
Grades K–2	
Digitizing information	Students select a shape (square, circle, a letter) and fills in grid squares on a sheet of graph paper to form the shape. Students discuss why this works better for some shapes than others. What is the impact of drawing a large shape or a small shape?
Coding information	Do we need to use the usual letters to write a word? Students assign a two-digit number to each letter of the alphabet (and some symbols). Students write words using this code instead of letters. Students swap words and decode them back to text.
Understanding binary values	Working in small groups, students identify and list types of information that can take on one of exactly two values (e.g. the printer has paper, whether Caps Lock key is on), a small number of discrete values (e.g. traffic signal color, name of font), or a wide range of values (e.g. person's height, person's name). Examples can come from computer applications and the real world.
Grades 3–5	
ASCII	Students are given an ASCII chart. Students then convert a sentence to ASCII. Students swap sentences, "read" the ASCII code, and convert it back to the original sentence. The teacher presents a list that contains text with upper case and lower case letters, numbers, and punctuation. Students sort the list based on the ASCII coding sequence.

Focus	**Sample Activity**
Range of values in a byte	Students are challenged to determine how many possible values eight bits can take on. Half the class is assigned values starting with 1; and the other half is assigned values starting with 0. Each half of the class should further subdivide, based on the second bit. Continue subdividing as far as possible, then each student lists and counts the possibilities for the remaining bits. Counts should be summed within each subgroup and passed back to the larger group. The total should equal 256!
Red–green–blue system for specifying color values	The teacher prepares overheads with colored squares. As each color is displayed, students try to determine the red, green, and blue values, in the range 0–255, of that color. For instance, the color pink might be Red=255, Green=128, Blue=128. This exercise can optionally be done using a color-mixing or color-choosing utility on the computer.
Grades 6–8	
Challenges of modeling information digitally	Students are given a list of types of information. Some naturally have discrete values (e.g. “number of students in classroom”); some have values that can easily be converted to discrete values (e.g. “height” – to nearest millimeter), and some are difficult to define digitally (e.g. smell). Students put each type into one of these categories and explain why they chose the categories they did.
ASCII and Unicode	Students translate a sentence containing non-ASCII characters into Unicode. The encoded version is given to another student who decodes it back to the original.

Focus	**Sample Activity**
Digitizing information	Students digitize data that is not naturally digital, such as best-dressed student or best movie. Student will disagree on criteria. Suggest setting up a point system for various qualities, and then adding up the points. Discuss the limitations of this approach.
Hexadecimal	Students learn the hexadecimal system for representing values, and learn to add in hexadecimal. If students are making web pages with HTML, they specify colors in HTML using hexadecimal.
Data compression	Students invent a new scheme for compressing song lyrics supplied by the teacher. They carefully write detailed instructions for compression and decompression.
Digital data representation	Students research and write reports on a real-world conversion from analog to digital data representation. They may explain how data is stored on compact disks and DVDs and transmitted over cable for TV.

Topic 7: Problem Solving and Algorithms

Topic Description:
People use computers to solve a wide variety of problems, involving everything from keeping track of money, to simulating aspects of the world, to generating beautiful images and sounds. Algorithms are rules or procedures for solving problems. In the context of computers, two important aspects of algorithms are that the problem has to be completely defined, and the steps of the procedure have to be specified with absolutely no ambiguity.

Background Information:
Algorithms are used in all aspects of daily and academic life. They appear in math classes as rules for performing operations on multi-digit numbers. In English class, students learn an algorithm for looking up words in the dictionary. The celebrated scientific method, involving hypotheses, experimentation, and observations, can be viewed as an algorithm. In their daily lives, students learn algorithms for preparing simple meals, straightening up their rooms, and finding the theater where a particular movie is playing.

In non-computer contexts, algorithms can be expressed informally ("open the dictionary to around the middle") or by example ("watch as I scramble the eggs so that you'll see how to do it"). However, because computers have no intelligence or common sense, an algorithm for a computer must be completely expressed in detail and without ambiguity. This may not be clear to the young student, who sees computers correct misspellings, adjust the color in photographs, and suggest helpful web pages. Students should learn that all these functions of computers are the results of algorithms created by humans.

Algorithms are composed of smaller algorithms or procedures. On a computer, the "bottom-most" level consists of simple operations that are "hard-wired" into the CPU. Complex algorithms can be created by combining simpler components.

Materials and Supplies:
No special resources required.

Student Learning Objectives	
A K–2 student will be able to:	
1.	Define the word "algorithm."
2.	Write the steps for a simple everyday task.
A 3–5 student will be able to:	
1.	Create an algorithm for an everyday task that results in successful task completion.
2.	Follow an algorithm to complete a task.
3.	Determine the output of an instruction that uses logical AND and OR.

A 6–8 student will be able to:
1. State that the utility of a computer is due to statements executing in a logical order.
2. Write an algorithm to solve an assigned problem using a specified set of commands.
3. Select the most efficient algorithm from a given set to solve a stated problem.

Focus	**Sample Activity**
Grades K–2	
Understanding the word “algorithm”	Teachers in all classes use the word “algorithm” to describe well-defined procedures.
Describing an algorithm	The class discusses a well-known game and agrees on step-by-step instructions for playing the game. The teacher writes the instructions on the board.
Grades 3–5	
Writing a precise algorithm	Students write the steps for how to complete a task such as making a sandwich. Each student gives the algorithm to another student who tries to complete the task by following the steps. If this is unsuccessful, the first student revises the instructions.
Understanding instructions with AND or OR	Students are given a step from an algorithm, with AND or OR, and they determine the outcome. For example: *If a number is greater than 1 and less than 7 then display Hello*. The teacher asks: What will the outcome be if the number is 9? What will the outcome be if the number is 3?
Combinatorics	Students determine the number of combinations possible for a hamburger when a restaurant offers a specified number of choices for bun, patty, and garnishes. The garnishes can be either a series of binary options (e.g. lettuce or no lettuce), or a choice of any three from a list.

Focus	**Sample Activity**
Grades 6–8	
Building algorithms	Students learn four primitive commands to control the movement of a robot: forward 1 step, backward 1 step, rotate left 30 degrees, rotate right 30 degrees. In small teams, students move one team member from a starting point to a destination. The teacher points out that computers have a limited vocabulary of basic commands that can be used in a logical manner to accomplish a task.
Algorithm efficiency	Students are presented a word problem and several methods of solving it. After they experiment with the various problem-solving methods, they identify the most time efficient method for solving the problem and what made it more efficient than other methods.
Relationship between Boolean Algebra and circuits	Students are given a diagram of an electronic circuit that includes an AND or OR gate. Students express the diagram using Boolean expressions and create a table of inputs and outputs.
Simulations	Students write in English an algorithmic description that simulates the changing populations of predators and prey (e.g. foxes and rabbits) in an isolated population, due to initial numbers, birth rates, and death rates due to hunting or lack of food. They convert the description to diagram, spreadsheet, or programming language form. The simulation is run several times (by hand or on the computer) and the results are recorded and reported to the class.
Parallel processing	Students write a procedure for one person to sort a shuffled card deck into a standard sequence (e.g. card rank within suit). Students then devise a procedure for two people to sort a deck, with the goal of keeping each person as busy as possible. The exercise is repeated for six people. Students conduct experiments to measure the time each algorithm takes (both clock time and person time). As a group, the class discusses the optimum number of people to use for sorting a card deck.

Topic 8: Computer Programming

Topic Description:
All operations performed by a computer are controlled by computer programs written in computer programming languages.

Background Information:
Computer programming involves the use of a programming language to write a series of instructions, called a computer program, that the computer can interpret and carry out. Computer programs express algorithms. Different kinds of programming languages have been developed for various kinds of problems. For example, COBOL was designed for business data processing, Logo was created to teach computer programming to children, and C was designed for writing system software. In addition to writing programs, computer programmers talk to people to understand their requirements, plan out a program before starting to write it, test programs to identify and remove errors, write documentation, and train users.

For over 50 years educators have had success introducing students to computer programming at every age from Kindergarten to college. A wide variety of programming languages, tasks, and pedagogical approaches have worked well in different situations.

Materials and Supplies:
A wide variety of programming languages and environments have been used successfully at all grade levels in the K-8 range. Select a programming language and tools based on the needs of the students, the teacher's background, and the requirements of the IT support team and school administration.

Student Learning Objectives	
A K–2 student will be able to:	
1.	State that computers are controlled by computer programs.
A 3–5 student will be able to:	
1.	State the purpose of programming languages.
2.	Code and test a sequential program to perform a simple task.
A 6–8 student will be able to:	
1.	Write a computer program that implements an algorithm.
2.	Code and test a program to solve a stated problem, using variables and at least one decision or loop.

Focus	**Sample Activity**
Grades K–2	
Terminology	Students define the terms “computer program,” “computer programmer,” and “computer programming language.” The teacher leads the students in a discussion relating these terms to their personal frame of reference. For instance, directions to a friend’s house = program, person who writes the directions = programmer, the text itself = language. Teachers should emphasize that just as you cannot get to your friend’s house without clear and accurate directions, the computer cannot perform an action without a program that clearly specifies the necessary steps.
Becoming aware of computer programs	The students compile a list of their favorite software applications. The teacher explains that these applications are computer programs that are providing instructions to the computer.
Grades 3–5	
Knowing about a variety of programming languages	Students are divided into teams. Each team is given a list of two programming languages. The team researches the languages and creates a table that compares their use. Comparison items might include devices, types of companies, institutions (engineering, financial, etc.) or individuals that might use the programming language. Teams present their findings to the class.
Familiarity with a computer programming language	Students read programs supplied by the teacher, perform the actions specified in the program, and determine the output or result.
Computer programming	Students are given the specifications for a simple program. The teacher leads a discussion that results in the appropriate statements being written. If possible, students enter the code in the computer and run the program. Alternatively, the students perform the actions specified in the program.

Focus	**Sample Activity**
Grades 6–8	
Converting an algorithm to a computer program	Students are given a simple mathematical algorithm, such as converting from Fahrenheit to Celsius, or determining the average of a list of numbers. On paper, they write a program to implement the algorithm, and then write and run the program on a computer.
Planning, writing, and testing computer programs	Students work in teams of three or more, taking the roles of Customer, Programmer, and Tester. The Programmer and Tester interview the Customer and write down specifications for a simple computer program. The Programmer writes a program that meets the specifications. The Customer and Tester devise a set of test cases that can be used to validate the program. The Tester runs the test cases and reports the results to the Programmer, who makes any needed corrections to the program. The Customer writes a final report.

Topic 9: Privacy and Security

Topic Description:
The power, ubiquity, and wide reach of computer systems give rise to concerns about privacy, safety, and security. That much of this power is easily accessible to young children makes these concerns particularly urgent.

Background Information:
Several aspects of computer systems, which are not always evident to children (or adults), are factors in privacy and security:

- Nothing stored on a computer or transmitted over a computer network is completely private. You can never be sure who has access to your disk drives or who reads your emails as they are copied from one part of the network to another.
- Data on a computer is extremely hard to erase. Even if you delete a file or email, one or more copies of it probably exist.
- If you send information out over the network, it is out of your control.
- There is no way to be completely sure that that the people you are communicating with over a network are really who they say they are.
- Multiple programs can run on a computer at the same time, and you don't usually know all the programs that are running.
- A file that is downloaded from a web site or copied from another computer may contain a virus or another type of harmful executable code.
- Many computer systems use passwords to provide security; however, many people choose passwords which are easy to guess.
- Computers sometimes fail or lose power, which can result in loss of data.

Materials and Supplies:
No special resources required.

Student Learning Objectives	
A K–2 student will be able to:	
1.	State safety rules for using a computer.
A 3–5 student will be able to:	
1.	State at least two safety concerns when using the Internet.
2.	List two challenges to keeping a computer secure.
3.	List information that should not be provided electronically without parental permission.
4.	Create a back-up policy that ensures data security.
A 6–8 student will be able to:	
1.	State at least two privacy concerns relating to computer use.
2.	List two concerns relating to data acquired over the Internet.
3.	Recognize inappropriate use of a network and the importance of notifying an adult.

4.	List several rules for safe web surfing and email use.

Focus	**Sample Activity**
Grades K–2	
Safe computer use	Students discuss and role play rules and safe practices for using a computer at school. Students take a list of rules home and discuss them with their parents. When the students return to school, they discuss whether school rules and home rules are the same and why they might differ.
Choosing good passwords	Students choose a password, between six and twelve characters long, write it on a 3x5 card along with their name, and put the card in a box which the teacher keeps. A few days later, students are asked to recall their passwords, and the teacher checks how many remembered correctly. The class discusses what makes a password easy to remember and whether easy to remember also means easy for others to guess?
Grades 3–5	
Internet safety	Students research Internet safety rules and report back to the class. The whole class then creates a shared list of Internet safety rules.
Computer security	Students discuss reasons why we lock doors and put personal items away. What might happen if we did not lock the door to our home? What are the possible consequences of leaving personal items (e.g. underwear) out when friends come to play? The teacher helps students relate this to personal exposure on the Internet.
Digital theft	Students discuss what “theft” means in the context of digital information. When is making a copy of something the same as stealing it?

Focus	Sample Activity
Keeping it personal	Students create an Internet Use Promise contract where they state the information that they promise not to share with anyone on the Internet. Students sign their individual contracts and present them to their parents.
Backing up files	The teacher demonstrates methods for backing up information specific to the classroom situation. Students discuss the frequency of backup.
Safe web surfing	Students organize or participate in an Internet Safety Day, where students are allowed to go to web sites of their choosing that enhance learning. When students go to games sites that have fighting or guns (not allowed at school) or that just provide play and not learning, the class discusses a better choice.
Determining the appropriateness of email	The teacher sends several emails to students. Some have blank subject lines, some have appropriate subject lines and educational content, and others contain content that could be considered phishing. Each student decides which can be safely opened and represents appropriate communication partners, and responds to only safe emails. Students delete unsafe/inappropriate email and state the reason they believe it is unsafe.
Grades 6–8	
Dangers of revealing personal information	Students discuss, read about, or role play situations in which personal information in digital format is accessed by unintended people. Working together, the class creates guidelines for avoiding these situations.
Making friends on the Internet	Students discuss safe ways to find friends on the Internet, for example pen pals through a teacher, or moderated social networks for children.

Focus	**Sample Activity**
How viruses are spread	The teacher shows, with animation or role play, how viruses may be transmitted through downloads from web sites, opening e-mail attachments, and clicking on links in e-mail messages. Students discuss safe practices (keeping anti-virus software up-to-date, avoid clicking on links unless you know the person who sent the message). Students compare the spread of a computer virus to the spread of a cold virus.
Types of malicious software	Students research and write definitions of several types of malicious software, such as worms, viruses, and Trojan horses.
Staying safe in our electronic world	Students write a “Class Statement Against Cyberbullying,” which lists actions that should be considered cyberbullying, how a victim of cyberbullying should respond, and how a cyberbully should be punished by the school or his or her parents.

Topic 10: Evaluating and Using Information from Networked Sources

Topic Description:
A vast amount of information is available over the Internet, as well as from specialized networks that may be available at libraries. For many students, “research” is equivalent to “web search.” Students need to know how to evaluate and use information acquired from networked sources effectively and ethically.

Background Information:
With the wide availability and access to the Internet, many students assume that they know how to use search engines and that information obtained from these search engines is always accurate.

Certain issues arise when evaluating information from any source (not just digital, networked ones). These include questions about the nature of knowledge, reliability, author bias, and whether the information could have been modified or corrupted during transmission. Using digital and networked information can exacerbate these concerns. The identity or background of the author is often unknown, or may not be what it is purported to be. Also, digital information can be modified without a trace.

Materials and Supplies:
A computer with Internet access, and a search engine.

Student Learning Objectives	
A K–2 student will be able to:	
1.	Identify keywords given a topic.
2.	Use a teacher-selected search engine.
A 3–5 student will be able to:	
1.	Verify accuracy of information.
2.	Use a variety of search engines.
3.	State the probable type of information to be found at a web site based on the domain.
A 6–8 student will be able to:	
1.	Evaluate web sources for relevancy and reliability.
2.	Use web resources in a legally acceptable manner.

Focus	**Sample Activity**
Grades K–2	
Identifying keywords	Students pick a topic of study from a list and with help from the teacher, create a list of good keywords for a search.
Using a search engine	Students open a teacher-selected search engine and enter a keyword. Students follow five links and determine if the content returned is related to the keyword selected.
Grades 3–5	
Accuracy of Internet information	Students search for information relating to a history or science topic that is currently under study. Students follow one link and compare the information in these sources to that in the textbook and their notes, looking for both differences and similarities in the information.
Search engine performance	Students perform the same search using three or more search engines. They identify the differences in the order in which the references are listed. Students identify the number of documents located and the types of information provided.
Grades 6–8	
Evaluating relevancy of web sources	Students perform a web search and then evaluate the responses for relevancy to the research topic. What percentage of the sources is useful? What happens to the relevancy as the student moves through the various listed sources? Students perform a second search within the results to narrow the scope of the information presented. What happens to the relevancy of the information as the search is refined? Are some relevant sources eliminated or moved to later in the list based on the refinement?
Using relevant information	Students prepare a paper on an assigned topic using both library and web sources.

Focus	**Sample Activity**
Copyright	Students are assigned to make a multi-media presentation about a state or country, with the special provision that they must ensure and document that all resources such as text, images, and music that they did not create themselves are used legally. They may have to justify “fair use,” quote a web site’s license or terms of service, or write to a publisher to get permission.

Topic 11: Human Computer Interaction

Topic Description:
Computer hardware and software systems are novel and complex inventions which are often successful only to the extent that they interact well with humans. The term “user-friendly” is often used to describe a system that users find easy to learn and efficient to use.

Background Information:
Here are several important aspects of usability.

- The system is easy for a new user to learn.
- The system accommodates users with a variety of ages, skill levels, and physical challenges.
- Errors are prevented or detected.
- Experienced users can perform actions quickly.
- People interact with the system in different ways, such as a keyboard and a mouse.

Materials and Supplies:
A computer with several software applications.

Student Learning Objectives	
A K–2 student will be able to:	
1.	State differences in the user interfaces of two software applications.
2.	Sit in an ergonomically correct position when using a computer.
A 3–5 student will be able to:	
1.	Apply the principles of ergonomics when working on a computer.
2.	Modify display settings for personal preference.
A 6–8 student will be able to:	
1.	Evaluate the usability of a user interface.
2.	Create a user-centered design.
3.	Explain why different software settings affect the ease of use of the interface for certain users, such as physically challenged or elderly users.

Focus	**Sample Activity**
Grades K–2	
Software application differences	Students use several application programs and create a chart showing differences that make some easier to use than others.
Ergonomics	Students are shown how to sit correctly and use proper body position when using the computer.
Grades 3–5	
Evaluating software ease of use	In small groups, students use an application program. Each group evaluates the program's ease of use for a specific type of user, based on age, training, or physical challenge.
Selecting a good user interface	Given several options of computer interface settings (colors, fonts, etc.), students choose the one that is most appealing and explain why it is the best choice.
Ergonomics	Arrange the work environment to permit proper posture when keyboarding, appropriate distances for monitors, and proper lighting.
Grades 6–8	
Interface evaluation	Students study the user interface of a non-digital device (e.g. a bicycle, a combination lock, a pen) and list aspects that make it easy to use or hard to use. Students study the user interface of an electronic device and list aspects that make it easy to use or difficult to use.

Focus	**Sample Activity**
User-centered design	Students design a new software application, such as a website. They identify their target users, ask those users about their needs, and create a design on paper. The user reviews the design and the student updates the design based upon the user feedback.
User differences	The teacher demonstrates some built-in usability features in the operating system and in application software. Students describe why some users might find these features helpful.
Deciding between check boxes and radio buttons	Students find software applications or web pages that have check boxes and radio buttons, and observe the differences in functionality. Students design the user interface for a new application for choosing a dog at a pound, and determine whether to use check boxes or radio buttons for selecting characteristics such as breed, gender, age, and color.
Accessibility	Students discuss ways in which technology helps people with physical challenges (for example, eye glasses), and analyze the usability designs of these technologies. Students research and, if possible, use hardware and software that helps physically challenged people use the computer.

Topic 12: Computers in Society

Topic Description:
Computers have had a significant impact on our society. Students need to consider the effects of computers upon society, understand appropriate ways of using computers, and imagine the impact of future technological innovations. Job descriptions have changed as technology has evolved. An understanding of the role of computers in various careers is important.

Background Information:
As computer technology has evolved, it has affected society in many ways from ethics, to ergonomics, to workflow. For example, while it was common in the past for business workers to use secretaries for typing memos and taking dictation, most office staff now send their own e-mail and use grammar and spell checkers as they type their own letters.

The goal of this topic is for students to consider the effects of computers upon society, understand appropriate ways of using computers, and imagine the impact of future technological innovations. In order to do these things, students must learn about the history of computing and its rapid change. They must also understand the diversity of today's computing devices from laptops, to desktops, to ATMs to cell phones. Finally, students should understand the context of ethical decisions relating to technology—both its use and its creation.

Materials and Supplies:
No special resources required.

Student Learning Objectives
A K–2 student will be able to:
1. List ways in which people use computers at work and in their daily lives.
A 3–5 student will be able to:
1. Identify ethical and unethical computing behaviors.
2. List major events in computer history and state how each event changed daily life.
A 6–8 student will be able to:
1. Behave ethically when using the computer.
2. Describe imagined future changes in technology and how they may affect daily life.
3. Name and describe the contributions of two or more computer scientists.
4. Identify at least two computing careers.

Focus	Sample Activity
Grades K–2	
Computers at work	Students role play or discuss the use of computers in jobs such as police work, office work, the supermarket, the doctor's office, and the classroom.
Grades 3–5	
Ethical and unethical behaviors	The teacher presents several open-ended scenarios and asks the students to describe ethical and unethical behaviors in each situation. For example, a student has figured out how to hack into the school server and change grades. Students describe why the behavior is unethical and identify the victims of the crime.
Computer history	Students create a timeline of major technological innovations. In small groups, they write a description of life before and after the innovation. For example, they could describe the ways in which access to mobile phones has changed families.
Grades 6–8	
Reading an acceptable use policy	Students are divided into teams. Each team is assigned a section of the school's acceptable use policy and prepares and delivers a presentation about its section.
Future changes	In small groups, students brainstorm a future information technology they would like to see. They determine how it will work and how it will be used. Students then write descriptive essays describing life after the technology becomes commonly available. For example, they might describe life after every car has a built-in GPS system.
History of computer science	Students are each assigned an important computer scientist or invention about which they write a short biographical essay.

Focus	**Sample Activity**
Social impacts of computerization	Students interview their parents or other adults to find out about society and life before a selected digital technology came into use.
Computing careers	The teacher invites speakers from the community to discuss their jobs as computer scientists, engineers, information technologists, network administrators, etc. The speakers should discuss what they do every day, their educational background, and provide some idea of career growth over time.

Appendix A

The International Society for Technology in Education (ISTE) publishes standards for K–12 technology education. Their 1998 National Educational Technology Standards for Students (NETS•S) was a basis for the Level I curriculum as presented in the *ACM Model Curriculum for K-12 Computer Science.* NETS•S was updated in 2007, and emphasizes student abilities in six areas: Creativity and Innovation; Communications and Collaboration; Research and Information Fluency; Critical Thinking, Problem Solving, and Decision Making; Digital Citizenship; and Technology Operations and Concepts. The following table shows how each Focus area in the *Level 1 Objectives and Outlines* promotes learning and understanding in one or more of the NETS•S categories.

Topic	**Focus**	**Creativity and Innovation**	**Communication and Collaboration**	**Research and Information Fluency**	**Critical Thinking, Problem Solving, and Decision Making**	**Digital Citizenship**	**Technology Operations and Concepts**
1. Parts of a Computer	Vocabulary (K-2) (3-5)						X
	Roles of PC components (K-2) (3-5) (6-8)				X		X
	Starting up and shutting down a PC (K-2)				X		X
	Keyboard and mouse (K-2) (3-5) (6-8)						X
	Evaluating features (3-5)				X		X
	Volatile and non-volatile memory (3-5)						X
	Roles of components in electronic devices (6-8)			X			X
	Units of measurement (6-8)						X
	Cables and ports (6-8)						X
	History of computers (6-8)			X			X
2. Standard Software	Locating software programs (K-2)						X
	Use of software (K-2)	X	X				X
	Exposure to software applications (3-5)	X					X
	Choosing and using software (3-5)			X	X		X
	Knowledge of software suites (3-5)	X	X				X
	Using built-in help (3-5)	X					X
	Multiple software applications can be used to complete the same task (6-8)			X	X		X

Topic	Focus	Creativity and Innovation	Communication and Collaboration	Research and Information Fluency	Critical Thinking, Problem Solving, and Decision Making	Digital Citizenship	Technology Operations and Concepts
	Critical thinking about software applications (6-8)	X			X		
3. Operating Systems	Logging on to the computer (K-2)						X
	Desktop icons and software applications (K-2)						X
	Good file names (K-2)				X		
	Standard OS dialogs (K-2)						X
	Menus and buttons (K-2)						X
	File name extensions (3-5)				X		
	Distinguishing proprietary and non-proprietary file types (3-5)				X		
	Understanding files and directories (folders) (3-5)				X		
	Selecting an appropriate place to save a file (3-5)				X		
	Manipulating windows (3-5)						X
	OS utilities (3-5)						X
	Navigating between running applications (3-5)						X
	Comparing operating systems (6-8)				X		
	Understanding directories (6-8)				X		
	Exploring multi-tasking (6-8)				X		
	Role of the OS (6-8)						X
	Using the clipboard (6-8)						X
4. Networks	Understanding networks (K-2)		X		X		X
	Working with local and remote resources (K-2)		X		X		X
	Protocols (3-5)		X		X		X
	Hostnames and network topologies (3-5)		X		X		X
	Network topologies (3-5)		X		X		X
	Client-server architecture (3-5)		X		X		X
	Error correction (6-8)		X		X		X
	Data communication over networks (6-8)		X		X		X
	Network topologies (6-8)		X	X	X		X

Topic	Focus	Creativity and Innovation	Communication and Collaboration	Research and Information Fluency	Critical Thinking, Problem Solving, and Decision Making	Digital Citizenship	Technology Operations and Concepts
5. The World Wide Web and Communicating over Networks	Accessing the WWW (K-2)						X
	Using a web site (K-2)			X		X	X
	Finding and saving web data for later use (3-5)			X			X
	Editing a web page template that includes HTML tags (3-5)	X	X		X		X
	Search techniques (3-5)		X	X	X		X
	Top level domains (3-5) (6-8)				X		X
	Creating a web site that conforms to standards (6-8)	X	X		X	X	X
	Social networks (6-8)		X	X	X	X	
6. Representing Information Digitally	Digitizing information (K-2)	X			X		X
	Coding information (K-2)	X	X		X		X
	Understanding binary values (K-2)		X		X		X
	ASCII (3-5)	X	X				X
	Range of values in a byte (3-5)	X	X		X		X
	Red-green-blue system for specifying color values (3-5)	X	X		X		X
	Challenges of modeling information digitally (6-8)	X	X		X		X
	ASCII and Unicode (6-8)			X			X
	Digitizing information (6-8)	X	X		X		X
	Hexadecimal (6-8)						
	Data compression (6-8)	X	X		X		X
	Digital data representation (6-8)		X	X	X		X
7. Problem Solving and Algorithms	Understanding the word “algorithm” (K-2)				X		X
	Describing an algorithm (K-2)		X		X		X
	Writing a precise algorithm (3-5)	X	X		X		X
	Understanding instructions with AND or OR (3-5)				X		X
	Combinatorics (3-5)		X	X	X		
	Building algorithms (6-8)	X	X		X		X
	Algorithm efficiency (6-8)				X		X

Topic	Focus	Creativity and Innovation	Communication and Collaboration	Research and Information Fluency	Critical Thinking, Problem Solving, and Decision Making	Digital Citizenship	Technology Operations and Concepts
	Relationship between Boolean Algebra and circuits (6-8)				X		X
	Simulations (6-8)	X	X	X	X		
	Parallel processing (6-8)	X	X	X	X		
8. Computer Programming	Terminology (K-2)		X				X
	Becoming aware of computer programs (K-2)		X		X		X
	Knowing about a variety of programming languages (3-5)		X	X	X		X
	Familiarity with a computer programming language (3-5)		X		X		X
	Computer programming (3-5)	X	X		X		X
	Converting an algorithm to a computer program (6-8)	X	X		X		X
	Planning, writing, and testing computer programs (6-8)	X	X		X		X
9. Privacy and Security	Safe computer use (K-2)					X	X
	Choosing good passwords(K-2)					X	X
	Internet safety (3-5)			X		X	X
	Computer security (3-5)					X	X
	Digital theft (3-5)					X	X
	Keeping it personal (3-5)		X			X	X
	Backing up files (3-5)					X	X
	Safe web surfing (3-5)		X			X	X
	Determining the appropriateness of email (3-5)		X		X	X	
	Dangers of revealing personal information (6-8)		X	X	X	X	X
	Making friends on the Internet (6-8)		X			X	X
	How viruses are spread (6-8)		X			X	X
	Types of malicious software (6-8)		X	X		X	X
	Staying safe in our electronic world (6-8)		X		X	X	

Topic	Focus	Creativity and Innovation	Communication and Collaboration	Research and Information Fluency	Critical Thinking, Problem Solving, and Decision Making	Digital Citizenship	Technology Operations and Concepts
10. Evaluating and Using Information from Networked Sources	Indentifying keywords (K-2)			X	X		X
	Using a search engine (K-2)			X	X		X
	Accuracy of Internet information (3-5)			X	X		X
	Search engine performance (3-5)			X	X		X
	Evaluating relevancy of web sources (6-8)			X	X		X
	Using relevant information(6-8)			X	X		X
	Copyright (6-8)			X		X	
11. Human Computer Interaction	Software application differences (K-2)				X		X
	Ergonomics (K-2) (3-5)						X
	Evaluating software ease of use (3-5)			X	X		X
	Selecting a good user interface (3-5)				X		X
	Interface evaluation (6-8)				X		X
	User-centered design (6-8)	X	X		X		X
	User differences (6-8)				X		X
	Deciding between check boxes and radio buttons (6-8)	X			X		X
	Accessibility (6-8)			X		X	X
12. Computers in Society	Computers at work (K-2)					X	
	Ethical and unethical behaviors (3-5)				X	X	
	Computer history (3-5)		X		X	X	
	Reading an acceptable use policy (6-8)				X	X	
	Future changes (6-8)				X	X	
	History of computer science (6-8)				X	X	
	Social impacts of computerization (6-8)				X	X	
	Computing careers (6-8)				X	X	

Appendix B

Level I Curriculum: Facets[1] of Computer Science

The Level II and Level III Curriculums presented, in an Appendix, an alternate approach to organizing the K-12 computer science material. In this approach, computer science can be seen as comprising four facets:

- Computer Hardware
- Using Computer Software
- Solving Problems by Developing Software
- Computers, People, and Society

Each of these facets has principal concepts and underlying themes, which are described below.

Computer Hardware

The development of the electronic computer has been one of the technological marvels of the last century. Research and development of computers and peripherals actively continues.

Principal concepts and themes of this facet are:

1. At a fundamental level, all computers are collections of circuits.
2. The most common architecture for computers is based on a central processor, memory, and peripherals.
3. Memory storage devices (including punch cards, paper tape, cassette tapes, hard disk drives, floppy disks, CD-ROMs and DVDs, memory sticks, RAM, ROM, cache, and video memory) have a variety of characteristics.
4. Memory storage devices usually store information in units called bytes, and each byte has a numeric address.
5. Computer processors (chips) are almost ubiquitous in cars, cell phones, traffic signal controllers, and other embedded devices.
6. Peripherals serve two main functions, input and output. Specialized peripherals are used in non-PC devices such as robots, satellites, cell phones, and digital cameras.
7. Computers are connected in networks via a wide variety of communication media, including telephones (with modems), Ethernet cable, and wireless.
8. Various tools (screwdrivers, Allen wrenches) are used to open computer cases and add or remove components.
9. Safety concerns must be kept in mind when assembling or fixing computers or any electronic equipment.

[1] These categories were defined by Daniel Frost, Donald Bren School of Information and Computer Sciences, University of California, Irvine and initially used in *A Model Curriculum for K-12 Computer Science: Level II Objectives and Outlines* and *A Model Curriculum for K-12 Computer Science: Level III Objectives and Outlines*.

Using Computer Software

Some software is written to be tightly integrated with specific hardware, as in a cell phone or a digital camera. In other cases, a software application, such as a word processor or a Web browser, is primarily designed to run on standard hardware. Sometimes software works "behind the scenes" and can be almost invisible, as in the Internet or many parts of an operating system. Familiarity with a variety of computer software programs and with the basic concepts underlying many of them is a prerequisite for many jobs and for understanding a large part of 21st century culture.

Principal concepts of this facet are:
1. Software may be tightly or loosely coupled with the computer hardware on which it runs.
2. By representing information in digital format, computers can store, manipulate, and transmit that information.
3. The instructions a computer follows are in software, which means that they can be changed easily.
4. Related data is often stored in a "file." A file (usually) corresponds to a particular range of locations in a memory storage device. The data in the file often has a specific format. A file has a name, and often the name has an extension that is associated with a specific program that knows how to use the contents of the file.
5. Files are organized in a hierarchy of folders or directories.
6. Files should be backed up periodically.
7. The same data can often be displayed in multiple ways.
8. Personal computers' operating systems often have graphical user interfaces.
9. Commercial "shrink-wrapped" software includes categories such as word processors, Web browsers, presentation and slide managers, spreadsheets, databases, graphic and music content creators and editors, and e-mail clients.

Solving Problems by Developing Software

Computer software solutions are created by identifying a need or opportunity, analyzing how it can be addressed with software, designing and coding the program, carefully testing the program, and in many cases writing documentation and training the users. Gaining a basic understanding of how software is created gives students a deeper understanding of what computers can do.

Principal concepts of this facet are:
1. Creating software involves several common phases:
   (a) Identifying the requirements from the user's perspective
   (b) Planning how to write the program (particularly important when the program is large or more than one programmer will collaborate)
   (c) Following the plan by writing code in a computer programming language
   (d) Testing the program to assure it meets the original requirements, runs acceptably quickly, is user-friendly, and has other desired qualities
   (e) Turning the program over to the user, which involves training, documentation, and designing procedures

   Often, steps (d) and (e) are repeated after beta-testing by end users.
2. Computers follow programs, which are written by humans.
3. Computer programs are written in computer languages, which have rigid syntax utilizing a limited number of key words and symbols.

4. Computers have no "understanding" beyond what is explicitly coded into a computer program.
5. For a problem to be solved by a computer, every step must be defined in detail.
6. Information used by the computer must be represented as digital data.
7. Writing a computer program involves selecting or creating algorithms and data structures, and analyzing their performance and other characteristics. Algorithms to solve a specific problem vary widely, and often involve different trade-offs of space and time.
8. Algorithms implemented in computer programs are made up of elementary control structures, such as conditionals, loops, and subroutines.
9. Computer science has been developed in the 20th and 21st centuries, but the philosophic roots of the "laws of thought" and algorithmic thinking originated with Plato and the pre-Socratics.

Computers, People, and Society

Technical advances have driven social changes throughout history, and tools have shaped culture in many ways. The rapid development of computers, networks, and peripherals has an ongoing impact on society.

Principal concepts of this facet are:
1. Computer technology and software changes more quickly than ethics and laws, thus creating a constant tension in society.
2. The ubiquity of data in digital format presents new issues of privacy and security.
3. Computerized data is often copied and rarely deleted, raising issues of privacy, ethics, and ownership rights.
4. Humans are best at recognition, making connections between similar things, and learning by doing. Computers are best at following small instruction steps and processing digital data quickly and consistently.
5. A human-computer interface is the meeting point of the human and computer realms. A good interface minimizes the human's short-term memory load, is compatible with a diverse set of users, and prevents errors (Schneiderman 1998).
6. Computers are tools with several functions: to process data (to compute), to store data, to acquire and display data, and to move data from one computer to another (to communicate).

The following table is a cross-reference between these facets of computer science and the Level I Curriculum's topics and focuses.

Topic	Focus	Computer Hardware	Using Software	Developing Solutions	People & Society
1. Parts of a Computer	Vocabulary (K-2) (3-5)	X	X		
	Roles of PC components (K-2) (3-5) (6-8)	X	X		
	Starting up and shutting down a PC (K-2)	X	X		
	Keyboard and mouse (K-2) (3-5) (6-8)	X			
	Evaluating features (3-5)	X	X	X	

Topic	**Focus**	**Computer Hardware**	**Using Software**	**Developing Solutions**	**People & Society**
	Volatile and non-volatile memory (3-5)	X			
	Roles of components in electronic devices (6-8)	X			
	Units of measurement (6-8)	X	X		
	Cables and ports (6-8)	X			
	History of computers (6-8)				X
2. Standard Software	Locating software programs (K-2)		X		
	Use of software (K-2)		X		
	Exposure to software applications (3-5)		X	X	
	Choosing and using software (3-5)		X	X	
	Knowledge of software suites (3-5)		X	X	
	Using built-in help (3-5)		X		
	Multiple software applications can be used to complete the same task (6-8)		X	X	
	Critical thinking about software applications (6-8)		X	X	
3. Operating Systems	Logging on to the computer (K-2)		X	X	
	Desktop icons and software applications (K-2)		X	X	
	Good file names (K-2)		X	X	
	Standard OS dialogs (K-2)		X	X	
	Menus and buttons (K-2)		X	X	
	File name extensions (3-5)		X	X	
	Distinguishing proprietary and non-proprietary file types (3-5)		X	X	
	Understanding files and directories (folders) (3-5)		X	X	
	Selecting an appropriate place to save a file (3-5)		X	X	
	Manipulating windows (3-5)		X	X	
	OS utilities (3-5)		X	X	
	Navigating between running applications (3-5)		X	X	
	Comparing operating systems (6-8)		X	X	
	Understanding directories (6-8)		X	X	
	Exploring multi-tasking (6-8)		X	X	
	Role of the OS (6-8)		X	X	
	Using the clipboard (6-8)		X	X	
4. Networks	Understanding networks (K-2)	X		X	
	Working with local and remote resources (K-2)	X		X	
	Protocols (3-5)	X		X	
	Hostnames and network topologies (3-5)	X		X	

Topic	Focus	Computer Hardware	Using Software	Developing Solutions	People & Society
	Network topologies (3-5)	X		X	
	Client-server architecture (3-5)	X		X	
	Error correction (6-8)	X		X	
	Data communication over networks (6-8)	X		X	
	Network topologies (6-8)	X		X	
5. The World Wide Web and Communicating over Networks	Accessing the WWW (K-2)		X		
	Using a web site (K-2)		X		
	Finding and saving web data for later use(3-5)		X	X	
	Editing a web page template that includes HTML tags (3-5)		X	X	
	Search techniques (3-5)		X		
	Top level domains (3-5) (6-8)		X		
	Creating a web site that conforms to standards (6-8)		X	X	X
	Social networks (6-8)		X		X
6. Representing Information Digitally	Digitizing information (K-2)	X		X	
	Coding information (K-2)			X	
	Understanding binary values (K-2)		X	X	
	ASCII (3-5)		X		
	Range of values in a byte (3-5)	X	X		
	Red-green-blue system for specifying color values (3-5)	X	X		
	Challenges of modeling information digitally (6-8)		X	X	
	ASCII and Unicode (6-8)		X	X	
	Digitizing information (6-8)		X	X	
	Hexadecimal (6-8)		X	X	
	Data compression (6-8)		X		
	Digital data representation (6-8)		X	X	
7. Problem Solving and Algorithms	Understanding the word “algorithm” (K-2)		X	X	
	Describing an algorithm (K-2)		X	X	
	Writing a precise algorithm (3-5)		X	X	
	Understanding instructions with AND or OR (3-5)		X	X	
	Combinatorics (3-5)		X	X	
	Building algorithms (6-8)		X	X	
	Algorithm efficiency (6-8)		X	X	
	Relationship between Boolean Algebra and circuits (6-8)		X	X	

Topic	Focus	Computer Hardware	Using Software	Developing Solutions	People & Society
	Simulations (6-8)		X	X	
	Parallel Processing (6-8)		X	X	
8. Computer Programming	Terminology (K-2)		X		
	Becoming aware of computer programs (K-2)		X		
	Knowing about a variety of programming languages (3-5)		X	X	
	Familiarity with a computer programming language (3-5)		X	X	
	Computer programming (3-5)		X	X	
	Converting an algorithm to a computer program (6-8)		X	X	
	Planning, writing, and testing computer programs (6-8)		X	X	
9. Privacy and Security	Safe computer use (K-2)			X	X
	Choosing good passwords(K-2)			X	X
	Internet safety (3-5)			X	X
	Computer security (3-5)			X	X
	Digital theft (3-5)			X	X
	Keeping it personal (3-5)			X	X
	Backing up files (3-5)			X	X
	Safe web surfing (3-5)			X	X
	Determining the appropriateness of email (3-5)			X	X
	Dangers of revealing personal information (6-8)			X	X
	Making friends on the Internet (6-8)			X	X
	How viruses are spread (6-8)			X	X
	Types of malicious software (6-8)			X	X
	Staying safe in our electronic world (6-8)			X	X
10. Evaluating and Using Information from Networked Sources	Indentifying keywords (K-2)		X	X	
	Using a search engine (K-2)		X	X	
	Accuracy of Internet information (3-5)			X	X
	Search engine performance (3-5)		X	X	X
	Evaluating relevancy of web sources (6-8)		X	X	X
	Using relevant information(6-8)		X	X	X
	Copyright (6-8)		X	X	X
11. Human Computer Interaction	Software application differences (K-2)		X	X	X
	Ergonomics (K-2) (3-5)		X	X	X

Topic	Focus	Computer Hardware	Using Software	Developing Solutions	People & Society
	Evaluating software ease of use (3-5)		X	X	X
	Selecting a good user interface (3-5)		X	X	X
	Interface evaluation (6-8)		X	X	X
	User-centered design (6-8)		X	X	X
	User differences (6-8)		X	X	X
	Deciding between check boxes and radio buttons (6-8)		X	X	X
	Accessibility (6-8)	X	X	X	X
12. Computers in Society	Computers at work (K-2)	X	X	X	X
	Ethical and unethical behaviors (3-5)			X	X
	Computer history (3-5)	X	X	X	X
	Reading an acceptable use policy (6-8)				X
	Future changes (6-8)	X	X	X	X
	History of computer science (6-8)		X	X	X
	Social impacts of computerization (6-8)	X	X	X	X
	Computing careers (6-8)		X	X	X

References

International Society for Technology in Education (ISTE), *National Educational Technology Standards for Students,* 1998.

International Society for Technology in Education (ISTE), *National Educational Technology Standards for Students, Second Edition*, 2007.

Madden, B., Verno, A., Carter, D., Cooper, S., Cortina, T., Cudworth, R., Ericson, B., Parys, E. *A Model Curriculum for K-12 Computer Science: Level III Objectives and Outlines.* Computer Science Teachers Association, 2007. http://csta.acm.org/Curriculum/sub/Implementation.html

Shneiderman, Ben. *Designing the User Interface,* Third Edition. Addison Wesley, 1998.

Tucker, A., Deek, F., Jones, J., McCowan, D., Stephenson, C., and Verno, A. *A Model Curriculum for K-12 Computer Science: Final Report of the ACM K-12 Task Force Curriculum Committee, 2nd Ed.* Association for Computing Machinery (ACM), New York, New York, 2006. http://csta.acm.org/Curriculum/sub/ACMK12CSModel.html

Verno, A., Carter, D., Cutler, R., Hutton, M., Pitt, L. *A Model Curriculum for K-12 Computer Science: Level 2 Objectives and Outlines.* Computer Science Teachers Association, 2004. http://csta.acm.org/Curriculum/sub/Implementation.html