

**Theory and Foundation**

**Overall Expectations**

- TFV.01I • describe the stages in the software design process
- TFV.02I • define and explain the fundamental programming constructs
- TFV.03I • describe the functions of internal and external computer components
- TFV.04I • describe the relationship among networks, operating systems, and application software and explain their uses

**Problem Solving, Logic, and Design**

- TF1.01I – use input, processing, and output correctly as a model for solving problems using a computer
- TF1.02I – explain how clarity at each step in the problem-solving process determines the quality and effectiveness of the final product
- TF1.03I – define a problem by identifying the required result, the necessary user inputs, and the steps required to produce the result

**Hardware, Interaces, and Networking Systems**

- TF2.01I – use correct terminology to describe hardware concepts
- TF2.02I – identify the basic internal components of the computer
- TF2.03I – identify the functions of peripheral devices
- TF2.04I – describe operating system functions that meet user needs
- TF2.05I – describe networking system functions that meet user needs
- TF2.06I – compare and contrast application, programming, and systems software

**Programming Concepts**

- TF3.01I – use correct terminology to describe programming concepts
- TF3.02I – describe the types of data that computers store, including numbers and characters
- TF3.03I – define constants, variables, expressions, and assignment statements, including the order in which the operations are performed
- TF3.04I – explain the need for decision and repetition structures, and how they can be expressed in different programming languages
- TF3.05I – explain the difference between logic and syntax errors
- TF3.06I – explain the role of internal documentation in ensuring program correctness and clarity

**Skills and Processes**

**Overall Expectations**

- SPV.01I • apply a problem-solving model
- SPV.02I • select software to solve specific problems
- SPV.03I • use proper programming practice
- SPV.04I • create computer programs using fundamental programming constructs
- SPV.05I • use correctly an operating system that includes a local network to perform management tasks

**Problem Solving, Logic and Design**

- SP1.01I – state problems accurately to gain a clear understanding of what is required for a solution
- SP1.02I – identify and resolve ambiguities and missing information in a problem statement
- SP1.03I – design a simple method to obtain clear and precise information from the user
- SP1.04I – state the steps required to take user input and produce correct output
- SP1.05I – solve and verify solutions to simple problems using application software, calculators, and computer programs

**SP1.06I** – compare and contrast a variety of tools, such as application software, calculators, and computer programs, based on ease of use and time required

### Hardware, Interfaces, and Networking

**SP2.01I** – use correctly file management techniques to create, name, copy, move, delete, and organize files

**SP2.02I** – use correctly networking services to log on and off and access shared files and devices

**SP2.03I** – use correctly Internetworking services to access and navigate global information resources

**SP2.04I** – develop resources to share information locally and globally

**SP2.05I** – maintain backup copies of program files on different media

### Programming Concepts

**SP3.01I** – write numbers and characters in such a way that computers recognize them (e.g., place quotation marks around characters)

**SP3.02I** – use correctly constants, variables, expressions, and assignment statements to store and manipulate numbers and characters in a program

**SP3.03I** – use descriptive naming conventions for constants, variables, and expressions

**SP3.04I** – write input and output statements that conform to a program design

**SP3.05I** – write programs that compare data using constants, variables, and expressions

**SP3.06I** – write a program that uses a decision structure involving two or more alternatives

**SP3.07I** – write a program that uses a counted repetition structure

**SP3.08I** – use appropriate sequences, decisions, and loops to conform to a program design

**SP3.09I** – incorporate internal documentation to a specific set of standards to ensure clarity and maintainability

**SP3.10I** – trace the execution of programs to find and correct logic and syntax errors

**SP3.11I** – validate a program using appropriate data

## Impact and Consequences

### Overall Expectations

**ICV.01I** • describe the evolution of programming languages

**ICV.02I** • identify the social impact of computers and associated technologies

**ICV.03I** • identify related computer careers

### Specific Expectations

**IC1.01I** – describe the evolution of different levels of programming languages

**IC1.02I** – describe the need to translate higher-level languages to machine code to make a computer operable

**IC1.03I** – explain major developments in information technology and anticipate future changes

**IC1.04I** – describe software-related careers

**IC1.05I** – describe how computers change the ways in which information is collected and used and explain how this affects people's privacy and access to information

**IC1.06I** – describe how computers change the ways in which people live, work, and communicate

**IC1.07I** – comply with acceptable computer use policies

**IC1.08I** – use appropriate strategies to prevent potential health and safety problems associated with computer use, such as posture problems, eye strain, and musculoskeletal injuries