**Department of Education**

**and Early Development**

Departement of Education and Ealry Development logo

## Glossary for the Alaska Computer Science Standards

### A

**Abstraction:** The process of reducing complexity by focusing on the main idea. By hiding details irrelevant to the question at hand and bringing together related and useful details, abstraction reduces complexity and allows one to focus on the problem.

**Accessibility:** The design of products, devices, services, or environments for people who experience disabilities. Accessibility standards that are generally accepted by professional groups include the Web Content Accessibility Guidelines (WCAG) 2.0 and Accessible Rich Internet Applications (ARIA) standards.

**Algorithm:** A step‐by‐step process to complete a task.

**App**: A type of application software designed to run on a mobile device, such as a smartphone or tablet computer. Also known as a mobile application.

**Artifact**: Anything created by a human. See computational artifact for the definition used in computer science.

**Artificial Intelligence (A.I.)**: The branch of computer science dealing with the simulation of intelligent behavior in computers.

**Audience:** Expected end users of a computational artifact or system.

**Authentication:** The verification of the identity of a person or process.

### B

**Binary**: A numeric system that only uses two digits — 0 and 1. Computers operate in binary, meaning they store data and perform calculations using only zeros and ones.

**Block‐based**: Using pre‐programmed “blocks” of instructions to teach kids coding.

### C

**Code**: Any set of instructions expressed in a programming language.

**Complexity:** The minimum amount of resources, such as memory, time, or messages, needed to solve a problem or execute an algorithm.

**Component:** An element of a larger group. Usually, a component provides a particular service or group of related services. Computational: Relating to computers or computing methods.

**Computational artifact**: Anything created by a human using a computational thinking process and a computing device. A computational artifact can be, but is not limited to, a program, image, audio, video, presentation, or web page file.

**Computational thinking**: The human ability to formulate problems so that their solutions can be represented as computational steps or algorithms to be executed by a computer.

**Computer:** A machine or device that performs processes, calculations, and operations based on instructions provided by a software or hardware program.

**Computer science**: The study of computers and algorithmic processes, including their principles, their hardware and software designs, their implementation, and their impact on society.

**Computing**: Any goal‐oriented activity requiring, benefiting from, or creating algorithmic processes.

**Computing device**: A physical device that uses hardware and software to receive, process, and output information. Computers, mobile phones, and computer chips inside appliances are all examples of computing devices.

**Computing system**: A collection of one or more computers or computing devices, together with their hardware and software, integrated for the purpose of accomplishing shared tasks. Although a computing system can be limited to a single computer or computing device, it more commonly refers to a collection of multiple connected computers, computing devices, and hardware.

**Conditional**: A feature of a programming language that performs different computations or actions depending on whether a programmer‐specified Boolean condition evaluates to true or false.

**Configuration:**

* (process): Defining the options that are provided when installing or modifying hardware and software or the process of creating the configuration (product).
* (product): The specific hardware and software details that tell exactly what the system is made up of, especially in terms of devices attached, capacity, or capability.

**Connection:** A physical or wireless attachment between multiple computing systems, computers, or computing devices. Connectivity: A program’s or device’s ability to link with other programs and devices.

**Control structure:** A programming (code) structure that implements control. Conditionals and loops are examples of control structures.

**Culture:** A human institution manifested in the learned behavior of people, including their specific belief systems, language(s), social relations, technologies, institutions, organizations, and systems for using and developing resources.

**Cybersecurity:** The protection against access to, or alteration of, computing resources through the use of technology, processes, and training.

### D

**Data:** Information that is collected and used for reference or analysis. Data can be digital or nondigital and can be in many forms, including numbers, text, show of hands, images, sounds, or video.

**Data structure**: A particular way to store and organize data within a computer program to suit a specific purpose so that it can be accessed and worked with in appropriate ways.

**Data type**: A classification of data that is distinguished by its attributes and the types of operations that can be performed on it. Some common data types are integer, string, Boolean (true or false), and floating‐point.

**Debug/debugging**: The process of finding and correcting errors (bugs) in programs.

**Decompose/decomposition**: Breaking down a problem or system into components.

**Device:** A unit of physical hardware that provides one or more computing functions within a computing system. It can provide input to the computer, accept output, or both.

**Digital citizenship**: The norms of appropriate, responsible behavior with regard to the use of technology.

### E

**Efficiency:** A measure of the amount of resources an algorithm uses to find an answer. It is usually expressed in terms of the theoretical computations, the memory used, the number of messages passed, the number of disk accesses, etc.

**Encryption:** The conversion of electronic data into another form, called ciphertext, which cannot be easily understood by anyone except authorized parties.

**Event:** Any identifiable occurrence that has significance for system hardware or software. User‐generated events include keystrokes and mouse clicks; system‐generated events include program loading and errors.

**Execute:** To carry out (or “run”) an instruction or set of instructions (program, app, etc.).

### G

**Gigabyte**: A unit of computer memory. One gigabyte equals 1,000,000,000 bytes.

### H

**Hacking:** Gaining unauthorized access to other computers.

**Hardware**: The physical components that make up a computing system, computer, or computing device.

### I

**Implementation**: The process of expressing the design of a solution in a programming language (code) that can be made to run on a computing device.

**Inference:** A conclusion reached on the basis of evidence and reasoning. Input: The signals or instructions sent to a computer.

**Internet**: The global collection of computer networks and their connections, all using shared protocols to communicate.

**Internet of Things** (IoT): An umbrella term that refers to anything connected to the Internet. It includes traditional computing devices, but also includes a growing list of other devices, including home appliances, automobiles, wearable electronics and security cameras.

**Iterative**: Involving the repeating of a process with the aim of approaching a desired goal, target, or result.

### K

**Kilobyte**: A unit of computer memory. One kilobyte equals 1,000 bytes.

### L

**Loop:** A programming structure that repeats a sequence of instructions as long as a specific condition is true.

### M

**Megabyte:** A unit of computer memory. One megabyte equals 1,000,000 bytes. Memory: Temporary storage used by computing devices.

**Model**: A representation of some part of a problem or a system. Note: This definition differs from that used in science.

**Modularity:** The characteristic of a software/web application that has been divided (decomposed) into smaller modules. An application might have several procedures that are called from inside its main procedure. Existing procedures could be reused by recombining them in a new application.

**Module**: A software component or part of a program that contains one or more procedures. One or more independently developed modules make up a program.

### N

**Network**: A group of computing devices (personal computers, phones, servers, switches, routers, etc.) connected by cables or wireless media for the exchange of information and resources.

### O

**Operation**: An action, resulting from a single instruction that changes the state of data.

### P

**Packet**: The unit of data sent over a network.

**Procedure**: An independent code module that fulfills some concrete task and is referenced within a larger body of program code. The fundamental role of a procedure is to offer a single point of reference for some small goal or task that the developer or programmer can trigger by invoking the procedure itself.

**Process:** A series of actions or steps taken to achieve a particular outcome.

**Program:** A set of instructions that the computer executes to achieve a particular objective.

**Programming**: The craft of analyzing problems and designing, writing, testing, and maintaining programs to solve them.

**Protocol**: The special set of rules used by endpoints in a telecommunication connection when they communicate. Protocols specify interactions between the communicating entities.

### R

**Reliability**: An attribute of any system that consistently produces the same results, preferably meeting or exceeding its requirements.

**Remix:** The process of creating something new from something old. Originally a process that involved music, remixing involves creating a new version of a program by recombining and modifying parts of existing programs, and often adding new pieces, to form new solutions.

**Router**: A device or software that determines the path that data packets travel from source to destination.

### S

**Scalability**: The capability of a network to handle a growing amount of work or its potential to be enlarged to accommodate that growth. [Wikipedia]

**Software:** Programs that run on a computing system, computer, or other computing device.

**Storage**:

* (place) A place, usually a device, into which data can be entered, in which the data can be held, and from which the data can be retrieved at a later time.
* (process) A process through which digital data is saved within a data storage device by means of computing technology. Storage is a mechanism that enables a computer to retain data, either temporarily or permanently.

**String:** A sequence of letters, numbers, and/or other symbols. A string might represent, for example, a name, address, or song title. Some functions commonly associated with strings are length, concatenation, and substring.

**Structure:** A general term used in the framework to discuss the concept of encapsulation without specifying a particular programming methodology.

**Switch:** A high‐speed device that receives incoming data packets and redirects them to their destination on a local area network (LAN).

**System:** A collection of elements or components that work together for a common purpose.

### T

**Test case:** A set of conditions or variables under which a tester will determine whether the system being tested satisfies requirements or works correctly.

**Topology**: The physical and logical configuration of a network; the arrangement of a network, including its nodes and connecting links. A logical topology is the way devices appear connected to the user. A physical topology is the way they are actually interconnected with wires and cables.

**Troubleshooting**: A systematic approach to problem solving that is often used to find and resolve a problem, error, or fault within software or a computing system.

### U

**User Interface (U.I.)**: The means in which a person controls a software application or hardware device.

### V

**Variable:** A symbolic name that is used to keep track of a value that can change while a program is running. Variables are not just used for numbers; they can also hold text, including whole sentences (strings) or logical values (true or false). A variable has a data type and is associated with a data storage location; its value is normally changed during the course of program execution. Note: This definition differs from that used in math.