Glossary

Symbols

!= (is not equal to): a boolean operator placed between two variables that returns false if the two variables have the same value and true otherwise [Chapter 5]

&& (logical and): an operator placed between two boolean statements that returns true if both statements are true and false otherwise [Chapter 5]

! (logical not): a boolean operator placed before a boolean statement that returns the negation of the statement's value [Chapter 5]

|| (logical or): an operator placed between two boolean statements that returns false if both statements are false and true otherwise [Chapter 5]

A

abstraction: a declaration in a superclass that all subclasses must either override the superclass method or declare the subclass method as abstract [Chapter 11]

accessor methods: methods used to obtain a value from a data field [Chapter 7]

array: an ordered list of the same data type [Chapter 8]

ArrayIndexOutOfBoundsException: a run-time error caused by all calling of an index array that is either negative or greater than the highest index of the array [Chapter 8]

ArrayList object: an object of a type that is a subclass of List used on the AP Computer Science A Exam [Chapter 9]

assignment operator: a single equals sign ("=") indicating that an identifier should be assigned a particular value [Chapter 3]

B

base case: the indication that a recursive method should stop executing and return to each prior recursive call [Chapter 12]

binary: a system of 1's and 0's [Chapter 3]

binary search: a search of a sorted array that uses searches beginning in the middle and determines in which direction to sort [Chapter 9]

blocking: a series of statements grouped by { } that indicate a group of statements to be executed when a given condition is satisfied [Chapter 5]

boolean: a primitive data type that can have the value true or false [Chapter 3]

boolean operator ==: an operator that returns true if it is between two variables with the same value and false otherwise [Chapter 5]

C

casting: forcing a data type to be recognized by the compiler as another data type [Chapter 3]

character: a primitive data type representing any single text symbol, such as a letter, digit, space, or hyphen [Chapter 3]

class: a group of statements, including control structures, assembled into a single unit [Chapter 7]

columns: portions of 2D-arrays that are depicted vertically. These are the elements of each array with the same index. [Chapter 9]

commenting: including portions of the program that do not affect execution but are rather used to make notes for the programmer or for the user
[Chapter 3]

compile-time error: a basic programming error identified by the interpreter during compiling [Chapter 3]

compiling: translating a programming language into binary code [Chapter 3]

compound condition: a complicated condition that includes at least one boolean operator [Chapter 5]
Computer Science: different aspects of computing, usually development [Chapter 3]
concatenation operator: a plus sign ("+") used between two strings indicating that the two string values be outputted next to each other [Chapter 3]
condition: a boolean statement used to determine flow control [Chapter 5]
conditional statement: a statement that is executed only if some other condition is met [Chapter 5]
constructor: a method in an object class used to build the object [Chapter 7]

D

decrement operator (--): a symbol used after a variable to decrease its value by 1 [Chapter 3]
double: a primitive data type for a number that can include decimals [Chapter 3]
driver class: a class that is created to control a larger program [Chapter 7]
dynamically sized: having the ability to change length and to insert and remove elements [Chapter 9]

E

enhanced-for loop: a for loop with a simplified call used to span an array [Chapter 8]
escape sequence: a small piece of coding beginning with a backslash to indicate special characters [Chapter 3]

F

flow control: indication of which lines of programming should be executed in which conditions [Chapter 5]
for loops: a loop with not only a condition but also an initializer and incrementer to control the truth value of the condition [Chapter 6]
full: all values of the array are assigned [Chapter 8]

H

header: the "title" of a method used to indicate its overall function [Chapter 7]

I

identifiers: names given to indicate data stored in memory [Chapter 3]
if: a reserved word in Java indicating the condition by which a statement will be executed [Chapter 5]
increment operator (++): a symbol used after a variable to increase its value by 1 [Chapter 3]
index numbers: integers, beginning with 0, used to indicate the order of the elements of an array [Chapter 8]
infinite loop: a programming error in which a loop never terminates because the condition is never false [Chapter 6]
inheritance: the quintessential way to create relationships between classes [Chapter 11]
inheritance hierarchy: the quantification of relationships between classes by using "parent" and "child" classes [Chapter 11]
initializer list: known values used to assign the initial values of all the elements of an array [Chapter 8]
in-line or short comments: comments preceded by two forward slashes ("//") indicating that the rest of the line of text will not be executed [Chapter 3]
insertion sort: a sorting algorithm in which smaller elements are inserted before larger elements [Chapter 9]
instance data/data fields: variables, listed immediately after the class heading, of an object that are accessible by all of the object's methods [Chapter 7]

integer: a primitive data type for positive numbers, negative numbers, or 0 with no fractions or decimals [Chapter 3]

interpreter: the part of the developer environment that enables the computer to understand the Java code [Chapter 3]

L

list object: an object form of an array with a dynamic size that can store multiple types of data [Chapter 9]

logical error: an error that lies in the desired output/purpose of the program rather than in the syntax of the code itself [Chapter 3]

long comments: comments that use ("/*") to indicate the beginning of the comment and ("*/") to indicate the end [Chapter 3]

loop: a portion of code that is to be executed repeatedly [Chapter 6]

M

merge sort: a sorting algorithm in which an array is divided and each half of the array is sorted and later merged into one array [Chapter 9]

methods: a group of code that performs a specific task [Chapter 7]

multiple inheritance: an illegal activity in Java in which a subclass inherits from more than one superclass [Chapter 11]

mutator methods: methods used to change the value of a data field [Chapter 7]

N

null: the default value of an object that has not yet been assigned a value [Chapter 8]

NullPointerException: a run-time error caused by calling an object with a null value [Chapter 8]

O

object class: a class that houses the "guts" of the methods that the driver class calls [Chapter 7]

overloading: using two methods with the same names but with different numbers and/or types of parameters [Chapter 7]

override (overridden): used a method in a subclass that possessed the same as a method in the superclass, causing the subclass method to be executed [Chapter 11]

P

parameters: what types of variables, if any, will be inputted to a method [Chapter 7]

planning: outlining the steps of a proposed program [Chapter 7]

polymorphism: the ability of a subclass object to also take the form as an object of its superclass [Chapter 11]

precedence: the order by which Java will execute mathematical operations, starting with parentheses, followed by multiplication and division from left to right, followed by addition and subtraction from left to right [Chapter 3]

primitive data: one of the most basic types of data in Java [Chapter 3]

programming style: a particular approach to using a programming language [Chapter 3]

R

recursion: a flow control structure in which a method calls itself [Chapters 9, 12]

recursive call: the command in which a method calls itself [Chapter 12]

return type: what type of data, if any, will be outputted by a method after its commands are executed [Chapter 7]

rows: the portions of 2D-arrays that are depicted horizontally and can be seen as their own arrays [Chapter 9]

run-time error: an error that occurs in the execution of the program [Chapter 3]

S

search algorithms: methods of finding a particular element in an array [Chapter 9]

selection sort: a sorting algorithm in which the lowest remaining element is swapped with the element at the lowest unsorted index [Chapter 9]

sequential search: a search of all the elements of an array in order until the desired element is found [Chapter 9]

short-circuited: the second condition of an and statement is skipped when the first condition is false, rendering the statement false [Chapter 6]

sorting algorithms: methods of ordering the elements within an array [Chapter 9]

source code: a .java file that defines a program's actions [Chapter 7]

span: using a loop to use or change all elements of an array [Chapter 8]

string literal (string): one or more characters combined in a single unit [Chapter 3]

strongly typed: characteristic of a language in which a variable will always keep its type until it is reassigned [Chapter 3]

subclass: a "child" class, with more specific forms of the superclass [Chapter 11]

superclass: a "parent" class, the most general form of a class hierarchy [Chapter 11]

super keyword: a keyword used to call an overridden method [Chapter 11]

T

truth value: the indication of whether a statement is true or false [Chapter 5]

typed ArrayList: an ArrayList that allows only one type of data to be stored [Chapter 9]

V

variable: an identifier associated with a particular value [Chapter 3]

W

while loop: a loop that cycles again and again while a condition is true [Chapter 6]

white space: empty space intended to enhance readability [Chapter 3]