

# Project Summary

One of the special features of this text is the diversity of its projects. Project subject matter spans six broad academic areas, as this short table shows:

<b>abbreviation</b>	<b>description</b>	<b>easy</b>	<b>moderate</b>	<b>difficult</b>	<b>total</b>
CS	Computer Science and Numerical Methods	14	12	6	32
Business	Business and Accounting	10	11	3	24
Sociology	Social Sciences and Statistics	6	7	5	18
Math & Phys	Math and Physics	9	5	3	17
Engineering	Engineering and Architecture	2	8	5	15
Biol & Ecol	Biology and Ecology	0	2	4	6
	<b>totals</b>	<b>41</b>	<b>45</b>	<b>26</b>	<b>112</b>

The abbreviation in the first column above will be used in a larger table below as a brief identification of a particular academic area. The four right-side columns in the above table indicate the number of projects in various categories. Of course, the highest number of projects (32) occurs in the area of computer science and numerical methods. The 26 easy and moderate CS projects are typical CS introductory programming problems. The 6 difficult CS projects provide gentle introductions to some advanced topics like link list operations, database operations, and simulated annealing.

In addition, there are 24 projects in business and accounting, which include miscellaneous financial calculations, simple bookkeeping problems, and cost-accounting applications. There are 18 projects in social sciences and statistics, which include applications in sociology and political science, as well as general experience. There are 17 projects in math and physics, which include applications in both classical and chaotic mechanics. There are 15 projects in engineering and architecture, which include applications in heating ventilating and air conditioning (HVAC), electrical circuits, and structures. Finally, there are 6 projects in biology and ecology, which include realistic growth and predator-prey simulations. Although we've associated each project with one primary academic area, many of these projects can fit into other academic areas as well.

Because many of these projects apply to disciplines outside the field of computer science, we do not expect that the average reader will already know about all of these "other" topics. Therefore, in our problem statements we usually take considerable time to explain the topic as well as the problem. And we often explain how to go about solving the problem—in layman's terms. Therefore, working many of these projects will be like implementing computer solutions for customers who are not programmers themselves but understand their subject matter and know what they want you (the programmer) to do for them. They will explain their problem and how to go about solving it. But then they will expect you to create the program that actually solves that problem.

Because our project explanations frequently take considerable printed space, instead of putting them in the book itself, we put them on our Web site:

<http://www.mhhe.com/dean>

The following table provides a summary of the projects on the Web site. The table lists all of the book's projects in a sequence that matches the book's sequence. The first column identifies the first point in the

book at which you should be able to do the project, by chapter and section, in the form: ChapterNumber.SectionNumber. The second column is a unique project number for the chapter in question. The third column identifies the project's primary academic area with an abbreviation that's explained in the shorter table above. The fourth column indicates the approximate number of pages of code that our solution contains. The fifth column indicates the difficulty relative to where you are in your study of Java. For example, you can see that what we call "easy" involves progressively more pages of code as you progress through the book. The last two columns provide a title and brief description of each project.

**Project Summary**

Ch./Sec	Proj.	Academic Area	Sol. Pages	Difficulty	Title	Brief Description
2.7	1	Business	0.6	easy	Annual Bonus— (Flowchart)	Draw a flowchart for an algorithm that computes an annual bonus.
2.7	2	Business	0.3	easy	Annual Bonus— (Pseudocode)	Write pseudocode for an algorithm that computes an annual bonus.
2.7	3	Business	0.6	easy	Number of Stamps— (Flowchart)	Draw a flowchart for an algorithm that calculates the number of stamps needed for an envelope. Use one stamp for every five sheets of paper.
2.7	4	Business	0.3	easy	Number of Stamps— (Pseudocode)	Write pseudocode for an algorithm that calculates the number of stamps needed for an envelope. Use one stamp for every five sheets of paper.
2.7	5	Biol & Ecol	0.5	moderate	Five Kingdoms— (Pseudocode)	Write pseudocode for an algorithm that identifies a biological kingdom from a set of characteristics.
2.7	6	Math & Phys	0.6	easy	Speed of Sound— (Flowchart)	Draw a flowchart for an algorithm that provides the speed of sound in a particular medium.
2.7	7	Math & Phys	0.4	easy	Speed of Sound— (Pseudocode)	Write pseudocode for an algorithm that provides the speed of sound in a particular medium.
2.7	8	Business	0.6	moderate	Stock Market Return— (Flowchart)	Draw a flowchart for an algorithm that prints the type of market and its probability given a particular rate of return.
2.7	9	Business	0.4	moderate	Stock Market Return— (Pseudocode)	Write pseudocode for an algorithm that prints the type of market and its probability given a particular rate of return.
2.8	10	Business	0.3	moderate	Bank Balance— (Pseudocode)	Write pseudocode for an algorithm that determines the number of years until a growing bank balance reaches a million dollars.
2.9	11	Engineering	1.0	moderate	Loop Termination by User Query— (Flowchart)	Draw a flowchart for an algorithm that calculates the overall miles per gallon for a series of miles and gallons user inputs.

Project Summary						
Ch./Sec	Proj.	Academic Area	Sol. Pages	Difficulty	Title	Brief Description
2.9	12	Engineering	0.5	easy	Loop Termination by User Query— (Pseudocode)	Write pseudocode for an algorithm that calculates the overall miles per gallon for a series of miles and gallons user inputs.
2.9	13	Engineering	0.4	moderate	Loop Termination by Sentinel Value— (Pseudocode)	Write pseudocode for an algorithm that calculates the overall miles per gallon for a series of miles and gallons user inputs.
2.9	14	Engineering	0.3	easy	Loop Termination by Counter— (Pseudocode)	Write pseudocode for an algorithm that calculates the overall miles per gallon for a series of miles and gallons user inputs.
2.10	15	CS	0.4	moderate	Average Weight— (Pseudocode)	Write pseudocode for an algorithm that determines average weight for a group of items.
3.2	1	CS	NA	easy	Hello World Experimentation	Experiment with the <code>Hello.java</code> program to learn the meanings of typical compile-time and runtime error messages.
3.3	2	CS	NA	moderate	Research	Study Sun's Java Coding Conventions.
3.3	3	CS	NA	moderate	Research	Study Appendix 5 "Java Coding-Style Conventions."
3.16 3.23	4	Engineering	2.5	difficult	Truss Analysis	Given the load in the center of a bridge and the weights of all truss members, compute the compression or tension force in each truss member.
3.17	5	CS	1.0	easy	Sequence of Commands	Trace a sequence of commands and write a program that executes those commands.
3.17 3.23	6	CS	1.7	moderate	Computer Speed	Given a simple set of hardware and software characteristics, write a program that estimates the total time to run a computer program.
3.17 3.23	7	Engineering	2.7	moderate	HVAC Load	Calculate the heating and cooling loads for a typical residence.
3.17 3.23	8	Sociology	3.5	difficult	Campaign Planning	Write a program to help organize estimates of votes, money, and labor.
3.22	9	CS	1.0	easy	String Processing	Trace a set of string processing operations and write a program that implements them.
3.23	10	CS	1.2	easy	Swapping	Trace an algorithm that swaps the values in two variables, and write a program that implements that algorithm.
3.23	11	Math & Phys	1.0	easy	Circle Parameters	Write a program that generates and prints circle-related values.

(continued)

Project Summary						
Ch./Sec	Proj.	Academic Area	Sol. Pages	Difficulty	Title	Brief Description
3.23	12	Sociology	0.4	easy	One-Hundredth Birthday	Write a program that prompts the user for his/her birthday month, day, and year and prints the date of the user's one-hundredth birthday.
4.3	1	Math & Phys	1.7	easy	Stopping Distance	Write a program which determines whether a vehicle's tailgating distance is safe, given the speed of the vehicle, the vehicle's tailgating distance, and a formula that gives the distance required to stop the vehicle.
4.3 4.9	2	Engineering	1.9	moderate	Column Safety	Write a program that determines whether a structural column is thick enough to support the column's expected load.
4.3	3	Business	1.1	easy	Economic Policy	Write a program that reads in growth rate and inflation values and outputs a recommended economic policy.
4.8	4	Business	2.0	moderate	Bank Balance	Write a program that determines the number of years until a growing bank balance reaches a million dollars.
4.9 4.12	5	CS	2.6	difficult	Game of NIM	Implement the game of NIM. Start the game with a user-specified number of stones in a pile. The user and the computer take turns removing either one or two stones from the pile. The player who takes the last stone loses.
4.12	6	Math & Phys	1.0	easy	Triangle	Write a program that generates an isosceles triangle made of asterisks, given user input for triangle size.
4.12	7	Sociology	0.8	easy	Mayan Calendar	Implement an algorithm that determines the number of Tzolkins and the number of Haabs in one Calendar Round.
4.12	8	CS	0.9	easy	Input Validation	Implement an algorithm that repeatedly prompts for inputs until they fall within an acceptable range and computes the average of valid inputs.
4.14	9	Business	2.6	moderate	Tax Preparation	Write a program that calculates customers' income taxes using the following rules: <ul style="list-style-type: none"> <li>• The amount of taxes owed equals the taxable income times the tax rate.</li> <li>• Taxable income equals gross income minus \$1,000 for each exemption.</li> <li>• The taxable income cannot be less than zero.</li> </ul>

Project Summary						
Ch./Sec	Proj.	Academic Area	Sol. Pages	Difficulty	Title	Brief Description
4.14	10	CS	1.7	moderate	Text Parsing	Write a program that converts words to Pig Latin.
5.3	1	Math & Phys	1.2	easy	Trigonometric Functions	Write a demonstration program that asks the user to select one of three possible inverse functions, arcsin, arccos, or arctan, and input a trigonometric ratio. It should generate appropriate output, with diagnostics.
5.3	2	Math & Phys	0.7	easy	Combining Decibels	Determine the acoustical power level produced by the combination of two sound sources.
5.5	3	CS	1.5	moderate	Variable Name Checker	Write a program that checks the correctness of a user-entered variable name, i.e., whether it is: (1) illegal, (2) legal, but poor style, or (3) good style. Assume that “good style” variable names use letters and digits only, and use a lowercase letter for the first character.
5.6	4	CS	1.0	moderate	Phone Number Dissector	Implement a program that reads phone numbers, and for each phone number, it displays the phone number’s three components—country code, area code, and local number.
5.6	5	CS	1.1	difficult	Phone Number Dissector—robust version	Implement a more robust version of the above phone number program. Allow for shortened phone numbers—phone numbers that have just a local digit group and nothing else, and phone numbers that have just a local digit group and an area code and nothing else.
5.8	6	Business	1.0	moderate	Net Present Value Calculation	Write a program that computes the net present value of a proposed investment, given a discount rate and an arbitrary set of future cash flows.
6.4	1	Biol & Ecol	1.5	moderate	Plant Germination Observation	Write a program that: (1) creates an object called <code>tree</code> from the <code>MapleTree</code> class; (2) calls a <code>plant</code> method to record the planting of the seed; (3) calls a <code>germinate</code> method to record the first observation of a seedling and record its height; (4) calls a <code>dumpData</code> method to display the current values of all instance variables.

(continued)

Project Summary						
Ch./Sec	Proj.	Academic Area	Sol. Pages	Difficulty	Title	Brief Description
6.4	2	Business	0.5	easy	Bank Account	Given the code for a <code>BankAccount</code> class, provide a driver that tests that class by instantiating an object and calling its methods— <code>setCustomer</code> , <code>setAccountNum</code> , and <code>printAccountInfo</code> .
6.8	3	Math & Phys	1.5	moderate	Logistic Equation	Exercise the logistic equation: $\text{nextX} = \text{presentX} + r \times \text{presentX} \times (1 - \text{presentX})$ , where $\text{presentX} = (\text{present } x) / (\text{maximum } x)$ , and $r$ is a growth factor.
6.9	4	Math & Phys	0.9	easy	Circle	Given the code for a <code>CircleDriver</code> class, write a <code>Circle</code> class that defines a <code>radius</code> instance variable, a <code>setRadius</code> method, and a <code>printAndCalculateCircleData</code> method that uses the circle's radius to calculate and print the circle's diameter, circumference, and area.
6.10	5	Engineering	2.0	moderate	Digital Filter	Given a formula for a “Chebyshev second-order low-pass” filter or a “Butterworth second-order low-pass” filter, with appropriate parameter values, write a program that asks the user to supply a sequence of raw input values and generates the corresponding filtered output.
6.10	6	Sociology	3.1	difficult	Vending Machine	Write a program that mimics the operations of a vending machine. The program should read amounts of money inserted into the vending machine, ask the user to select an item, and then print the change that's returned to the user.
6.12	7	Math & Phys	1.1	easy	Rectangle	Implement a <code>Rectangle</code> class that defines a rectangle with length and width instance variables, mutator and accessor methods, and a boolean <code>isSquare</code> method.
6.12	8	Biol & Ecol	4.0	difficult	Predator-Prey Dynamics	Write a program that models a species that could be either predator or prey or both. Run a simulation that includes predators, prey, and limited renewable sustenance for the prey.
6.13	9	Math & Phys	2.1	moderate	Guitar Mechanics	Write a program that simulates the motion of a plucked guitar string.

Project Summary						
Ch./Sec	Proj.	Academic Area	Sol. Pages	Difficulty	Title	Brief Description
7.5 7.9	1	CS	3.5	difficult	Linked List	Given the code for a driver, implement a <code>Recipe</code> class that creates and maintains a linked list of recipes. The problem assignment specifies all instance variables and methods in UML class diagrams.
7.7	2	CS	2.5	easy	Automobile Description	Use method-call chaining to help display properties of automobiles.
7.7 7.9	3	Biol & Ecol	4.6	difficult	Carbon Cycle	Given the code for a driver, write a pair of classes for a program that models the carbon cycle in an ecosystem. Use two generic classes. One class, <code>Entity</code> , defines things. The other class, <code>Relationship</code> , defines interactions.
7.8	4	CS	1.4	easy	IP Address	Implement an <code>IpAddress</code> class that stores an IP address as a dotted-decimal string and as four octet ints.
7.9	5	Math & Phys	4.5	moderate	Fraction Handler	Given the main method of a driver class, write a <code>Fraction</code> class. Include the following instance methods: <code>add</code> , <code>multiply</code> , <code>print</code> , <code>printAsDouble</code> , and a separate accessor method for each instance variable.
7.10	6	Engineering	2.8	moderate	Electric Circuit	Write branch and node classes for lumped-circuit elements. A branch carries current through a resistor in series with an inductor. A node holds voltage on a capacitor connected to a common ground. Driver code is provided in the problem assignment.
7.10	7	Business	5.1	difficult	Cost Accounting	Write an object-oriented program that demonstrates cost accounting in a manufacturing plant.
7.10	8	Sociology	6.4	difficult	Political Campaign	Write a program to help organize estimates of votes, money, and labor. This is an object-oriented version of Project 8 in Chapter 3.
8.4	1	CS	1.6	easy	Input Validation	Implement an algorithm that repeatedly prompts for inputs until they fall within an acceptable range and computes the average of valid inputs. This is an object-oriented version of Project 8 in Chapter 4.

(continued)

Project Summary						
Ch./Sec	Proj.	Academic Area	Sol. Pages	Difficulty	Title	Brief Description
8.4	2	Engineering	4.0	difficult	HVAC Load	Calculate the heating and cooling loads for a typical residence. This is an object-oriented version of Project 7 in Chapter 3.
8.6	3	Sociology	2.6	moderate	Elevator Control	Write a program that mimics the operations of the inside of an elevator. The program should simulate what happens when the user chooses to go to a particular floor and when the user pulls the fire alarm.
8.9	4	CS	2.0	easy	Prototype Restructuring	Consider the NestedLoopRectangle program in Figure 4.17 in Section 4.12 to be a prototype. Using top-down methodology, restructure it into OOP format.
9.3	1	Sociology	2.7	easy	Person Class	Define a class that simulates the creation and display of <code>Person</code> objects.
9.4	2	Sociology	2.7	moderate	Homework Scores	Write a program that handles homework scores. Use instance variables for actual and maximum points on a particular homework, and use class variables for actual total and maximum total points on all homeworks combined.
9.3	3	Sociology	3.9	difficult	Political Approval Rating	Write a program that determines the mean and standard deviation of statistical samples.
9.4	4	Engineering	5.7	difficult	Solar Input for HVAC and Solar Collectors	Write a program that keeps track of where the sun is and determines how much solar energy penetrates a glass window of any orientation, at any place and time.
9.6	5	Business	2.7	moderate	Net Present Value Calculation	Write a program that computes the net present value of a proposed investment, given a discount rate and an arbitrary set of future cash flows. This is an OOP version of Project 6 in Chapter 5.
9.7	6	Math & Phys	7.0	difficult	Three-Body Problem	Write a program to model the three-body problem in which two equally sized moons circle the earth in different orbits. This illustrates chaotic dynamic motion.
10.4	1	Biol & Ecol	5.0	difficult	Demographic Projections	Write a program that projects future world population and average individual wealth as a function of fertility rates and resource extraction rates, and includes effects of governmental taxation and spending.

Project Summary						
Ch./Sec	Proj.	Academic Area	Sol. Pages	Difficulty	Title	Brief Description
10.6	2	CS	3.3	moderate	Dice-Throwing Simulator	Write a program that simulates the rolling of a pair of dice and prints a histogram showing the frequencies of possible results.
10.6	3	CS	5.1	difficult	Simulated Annealing—the Traveling Salesman Problem	Write a program that uses simulated annealing to solve the intractable problem of finding the shortest itinerary that visits all of the world’s major cities exactly one time.
10.7	4	Sociology	2.1	easy	Party Guest List	Write a program that creates a <code>Party</code> object, adds guests to the party, and prints party information.
10.9	5	Sociology	2.7	easy	Vowel Counter	Write a program that counts the number of uppercase and lowercase vowels in user-entered lines of text and prints a summary report of vowel counts.
10.9	6	Math & Phys	7.6	difficult	Solution of Simultaneous Algebraic Equations	Write a program that loads a set of simultaneous algebraic equations into two-dimensional arrays and solves the equations by Lower-Upper Decomposition.
10.9	7	Math & Phys	2.5	moderate	Linear Regression	Write a program that computes a linear regression by fitting a straight line to a series of random data.
10.10	8	Business	3.4	moderate	Purchase Vouchers	Write a program that creates business vouchers that record purchases, displays current voucher information, and records payments for those purchases.
10.11	9	Sociology	1.1	easy	Deck of Cards	Write a class that uses an <code>ArrayList</code> to hold a deck of cards.
10.13	10	Business	1.9	easy	Bookstore	Write a program that models the storing and retrieving of books based on title.
11.13	1	Biol & Ecol	5.5	difficult	Game of Spawn	Model a “game” that simulates reproduction and growth in a rectangular grid of cells. An X indicates life. A dead cell comes to life when it has exactly three living neighbor cells. A living cell remains alive only when surrounded by two or three living neighbor cells.
11.3	2	CS	0.7	easy	ASCII Table	Write a program that prints the 128-character ASCII table. It should print the table in eight tab-separated columns.

(continued)

Project Summary						
Ch./Sec	Proj.	Academic Area	Sol. Pages	Difficulty	Title	Brief Description
11.7	3	CS	0.8	easy	Circular Queue	A given program implements a circular-array queue. Rewrite the <code>isFull</code> , <code>remove</code> , and <code>showQueue</code> methods by replacing conditional operators, embedded assignments, and embedded increment operators with simpler, more understandable code.
11.7	4	Math & Phys	4.1	moderate	Polynomial Interpolation	Fit a polynomial to points on either side of a pair of points in an array of data and use that to estimate the value at a position between the pair of points.
11.9	5	CS	1.4	moderate	Bitwise Operations	Use arithmetic and logical shifting to display the binary values of numbers.
11.11	6	CS	3.5	moderate	Heap Sort	Use the heap-sort algorithm to sort data. (This is a robust in-place sorting algorithm with a computational complexity of $N\log N$ .)
12.2	1	Business	1.7	easy	Savings Accounts	Compute and display savings account balances that accumulate with compound interest.
12.4	2	Math & Phys	13.4	difficult	Statistics Functions	Write a program that generates values for the Gamma, Incomplete Gamma, Beta, Incomplete Beta, and Binomial statistical functions.
12.5	3	Business	3.3	easy	Car Program	Using inheritance, write a program that keeps track of information about new and used cars.
12.10	4	Sociology	16.4	difficult	Game of Hearts	Write a program that simulates a basic game of hearts with an arbitrary number of players. Give all players an identical set of good strategies which optimize the chance of winning.
13.7	1	Business	9.0	difficult	Grocery Store Inventory	Write an inventory program that keeps track of various kinds of food items. Use different methods in an <code>Inventory</code> class to process heterogeneous objects representing generic and branded food items. Store the objects together in a common <code>ArrayList</code> .

Project Summary						
Ch./Sec	Proj.	Academic Area	Sol. Pages	Difficulty	Title	Brief Description
13.7	2	Engineering	8.7	difficult	Electric Circuit Analysis	Write a program that calculates the steady-state currents in a two-loop electric circuit that has an arbitrary combination of discrete resistors, inductors, capacitors, and voltage sources in the legs of the circuit. Include methods to perform addition, subtraction, multiplication, and division of complex numbers—numbers that have real and imaginary parts.
13.8	3	Business	5.4	moderate	Payroll	Use polymorphism to write an employee payroll program that calculates and prints the weekly payroll for a company. Assume three types of employees—hourly, salaried, and salaried plus commission. Assume each type of employee gets paid using a different formula. Use an abstract base class.
13.8	4	Business	2.9	moderate	Bank Accounts	Write a bank account program that handles bank account balances for an array of bank accounts. Use two types of bank accounts, checking and savings, derived from an abstract class named <code>BankAccount</code> .
14.4	1	Sociology	4.0	moderate	Body Mass Index	Write a program that prompts the user for height and weight values and displays the associated body mass index.
14.5	2	CS	6.4	difficult	Storage and Retrieval of Objects in an Array	Search for a match with the key value in a relational table, using two different search algorithms, a sequential search and a hashed search.
14.9	3	CS	2.5	moderate	Date Formatting	Create a class named <code>Date</code> that stores date values and prints out the date in either a numeric format or an alphabetic format.
14.9	4	CS	5.5	difficult	Input Utility	Write a utility class that reads inputs from the keyboard and parses the following datatypes: <code>String</code> , <code>char</code> , <code>double</code> , <code>float</code> , <code>long</code> , and <code>int</code> . It should do input approximately like <code>Scanner</code> does.

(continued)

Project Summary						
Ch./Sec	Proj.	Academic Area	Sol. Pages	Difficulty	Title	Brief Description
15.4	1	Engineering	3.7	moderate	Road Use Survey	Model traffic flowing on a highway past a particular place, store observations, and read file later for analysis.
15.4	2	Business	2.9	moderate	Mail Merge	Write a program that reads a form letter from a text file and modifies custom fields.
15.5 15.9	3	CS	5.0	moderate	File Converter	Write a program that changes whitespace in text files.
15.8	4	CS	1.5	easy	Appending Data to an Object File	Implement code needed to append data to an object file.
16.12	1	Engineering	4.1	moderate	Animated Garage Door	Write a program that simulates the operation of an automatic garage door and its controls and visually display its position as it operates.
16.14	2	Sociology	3.0	moderate	Color Memorization	Write a program that tests the user's ability to memorize a sequence of colors.
16.14	3	Business	8.7	difficult	Grocery Inventory GUI	Write a GUI version of the Grocery Store Inventory project in Chapter 13.
16.15	4	Sociology	4.2	moderate	Word Order Game	Create a simple interactive game that helps kids practice their alphabetic skills.
16.16	5	Business	3.8	moderate	Airline Reservations	Write a GUI program that assigns seats on airline flights.
17.3	1	CS	1.7	easy	Changing Color and Alignment	Write an interactive program that modifies the color and position of buttons in a GUI window.
17.6	2	CS	1.9	easy	Click Tracker	Write an interactive program that modifies the borders and labels of buttons in a GUI window.
17.10	3	Sociology	3.4	moderate	Tic-Tac-Toe	Create an interactive Tic-Tac-Toe game.
17.10	4	Sociology	4.3	moderate	Word Order Game, revisited	Modify Chapter 16's Word Order Game program so it uses embedded layout managers.
17.10	5	Engineering	7.5	difficult	Thermal Diffusion in a Ground-Source Heat Pump's Well	Write a program that calculates temperatures in the earth around a ground-source heat pump's well. Display results in a color-coded plot of temperature as a function of distance from well center and time of year.