**CH6 Writing your own function**

**6.1 How does a function differ from a module?**

When a module finishes, the program merely returns back to the part of the program that called the module, and execution resumes at that point. When a function finishes, it returns a value back to the part of the program that called it.

**6.2 What is a library function?**

A prewritten function that comes with a programming language.

**6.3 Why are library functions like “black boxes”?**

The term "black box" is used to describe any mechanism that accepts input, performs some operation that cannot be seen on the input, and produces output. A library function can be regarded as a black box because you cannot see the code inside the function. The function accepts input, performs an operation on the input, and produces output.

**6.6 What is the purpose of the Return statement in a function?**

The Return statement specifies the value that the function returns to the part of the program that called the function. When the Return statement is executed, it causes the function to terminate and return the specified value.

**6.8 What is an IPO chart?**

A chart, or table, that describes a function's input, processing, and output.

**6.9 What is a Boolean function?**

A function that returns either true or false.

This type of error occurs when you try to assign a value of one data type to a variable of another type. (**type mismatch error)**

This is a string inside of another string. **(substring)**

**CH7 Input Validation**

**7.1 What does the phrase “garbage in, garbage out” mean?**

It means that if bad data (garbage) is provided as input to a program, the program will produce bad data (garbage) as output.

**7.2 Give a general description of the input validation process.**

When input is given to a program, it should be inspected before it is processed. If the input is invalid, then it should be discarded and the user should be prompted to enter the correct data.

**7.3 Describe the steps that are generally taken when an input validation loop is used to validate data.**

The input is read, and then a pretest loop is executed. If the input data is invalid, the body of the loop executes. In the body of the loop, an error message is displayed so the user will know that the input was invalid, and then the input read again. The loop repeats as long as the input is invalid.

**7.4 What is a priming read? What is its purpose?**

It is the input operation that takes place just before an input validation loop. The purpose of the priming read is to get the first input value.

**7.5 If the input that is read by the priming read is valid, how many times will the input validation loop iterate?**

None.

GIGO stands for (**garbage in, garbage out**)

The integrity of a program’s output is only as good as the integrity of the program’s (**input**).

Validation loops are also known as (**error traps**).

The term *empty input* describes what happens when (**an input operation attempts to read data, but there is no data to read**).

**CH8 Arrays**

**8.1 Can you store a mixture of data types in an array?**

No, you cannot. All of the items in an array must be of the same data type.

**8.2 What is an array size declaratory?**

A nonnegative integer that specifies the size of an array.

**8.3 In most languages, can the size of an array be changed while the program is running?**

No

**8.4 What is an array element?**

An individual storage location in an array.

**8.5 What is subscript?**

A number that identifies a specific element in an array.

**8.6 What is usually the first subscript in an array?**

0

**8.8 What does “array bounds checking” mean?**

Many languages support arrays bounds checking, which means they do not allow a program to use an invalid array subscript.

**8.9 What is an off-by-one error?**

An off-by-one error occurs when a loop iterates one time too many, or one time too few.

**8.10 What is search algorithm?**

An algorithm developed for the purpose of locating a specific item in a larger collection of data, such as an array.

**8.11 Which array element does the sequential search algorithm first look at?**

The first element in the array.

**8.12 What does the loop do in the sequential search algorithm? What happens when the value being searched for is found?**

The loop sequentially steps through each element in the array, comparing the elements to the value being searched for. When the value is found, the loop stops.

**8.13 How many elements does the sequential search algorithm look at in the case that the search value is not found in the array?**

It looks at every element in the array.

**8.14 How do you look for a partial string match when searching an array of strings for a value?**

You use a function similar to the *contains* function described in this chapter.

The *contains* function returns true if a string is found inside another string, or false otherwise.

**8.15 briefly describe how you calculate the total of the values in an array.**

To calculate the total of the values in an array, you use a loop with an accumulator variable. The loop steps through the array, adding the value of each array element to the accumulator.

**8.16 briefly describe how you get the average of the values in an array.**

The first step in calculating the average of the values in an array is to get the sum of the values. You use the algorithm for totaling the values in an array to perform this.

The second step is to divide the sum by the number of elements in the array.

**8.17 describe the algorithm for finding the highest value in an array.**

You create a variable to hold the highest value. In the examples shown in this book the variable is named highest. Then, you assign the value at element 0 to the highest variable. Next, you use a loop to step through the rest of the array elements, beginning at element 1. Each time the loop iterates, it compares an array element to the highest variable. If the array element is greater than the highest variable, then the value in the array element is assigned to the highest variable. When the loop finishes, the highest variable will contain the highest value in the array.

8.18 **describe the algorithm for finding the lowest value in an array.**

You create a variable to hold the lowest value. In the examples shown in this book the variable is named lowest. Then, you assign the value at element 0 to the lowest variable. Next, you use a loop to step through the rest of the array elements, beginning at element 1. Each time the loop iterates, it compares an array element to the lowest variable. If the array element is less than the lowest variable, then the value in the array element is assigned to the lowest variable. When the loop finishes, the lowest variable will contain the lowest value in the array.

**8.19 How do you copy the contents of one array to another array?**

You assign the individual elements of the array that you are copying to the elements of the other array. This is usually best done with a loop.

**8.20 How do you establish a relationship between the data stored in two parallel arrays.**

You use the same subscript to access data items in the two arrays.

**8.24 write a pseudocode declaration for a two-dimensional array initialized with the following table of data: 12 24 32 21 42**

**14 67 87 65 90**

**19 1 24 12 8**

Constant Integer ROWS = 3

Constant Integer COLS = 5

Declare Integer table[ROWS][COLS] = 12, 24, 32, 21, 42,

14, 67, 87, 65, 90,

19, 1, 24, 12, 8

**8.25 assume a program has the following declaration:**

**Constant Integer ROWS = 100**

**Constant Integer COLS = 50**

**Declare Integer info [ROWS] [CLOS]**

**Write pseudocode with a set of nested loops that store the value 99 in each element of the info array.**

Declare Integer row

Declare Integer col

For row = 0 To ROWS – 1

For col = 0 to COLS – 1

Set info[row][col] = 99

End For

End For

To make programs easier to maintain, many programmers use these to specify the size of an array. **(named constants)**

**CH10 Files**

**10.1 Where are files normally stored?**

On the computer's disk

**10.2 What is an output file?**

A file that a program writes data to. It is called an output file because the program sends output to it.

**10.3 What is an input file?**

A file that a program reads data from. It is called an input file because the program receives input from it.

**10.4 What three steps must be taken by a program when it uses a file?**

(1) Open the file (2) Process the file (3) Close the file

**10.5 In general, what are the two types of files? What is the difference between these two types of files?**

Text and binary. A text file contains data that has been encoded as text, using a scheme such as Unicode. Even if the file contains numbers, those numbers are stored in the file as a series of characters. As a result, the file may be opened and viewed in a text editor such as Notepad. A binary file contains data that has not been converted to text. As a consequence, you cannot view the contents of a binary file with a text editor.

**10.6 What are the two types of file access? What is the difference between these two?**

Sequential and direct access. When you work with a sequential access file, you access data from the beginning of the file to the end of the file. When you work with a direct access file, you can jump directly to any piece of data in the file without reading the data that comes before it.

**10.7 when writing a program that performs an operation on a file, what two file- associated names do you have to work with in your code?**

The file's external name and internal name. The external name is the file name that identifies the file on the disk. The internal name is like a variable name. It identifies the file in your program code.

**10.8 In most programming languages, if a file already exists what happens to it if you try to open it as an output file?**

The file's contents are erased.

**10.9 What is the purpose of opening a file?**

Opening a file creates a connection between the file and the program. It also creates an association between the file and its internal name.

**10.10** **What is the purpose of closing a file?**

Closing a file disconnects the program from the file.

**10.11 Generally speaking, what is a delimiter分隔符? How are delimiters typically used in files?**

A predefined character or set of characters that marks the end of piece of data. In many languages, a delimiter is written after each item that is stored in a file.

**10.12 In many systems, what is written at the end of a file?**

A special character, or set of characters, known as the end-of-file marker

**10.13 What is a file’s read position? Initially, where is the read position when an input file is opened?**

A file’s read position marks the location of the next item that will be read from the file. When an input file is opened, its read position is initially set to the first item in the file.

**10.14 In what mode do you open a file if you want to write data to it, but you do not want to erase the file’s existing contents? When you write data to such a file, to what part of the file is the data written?**

You open the file in append mode. When you write data to a file in append mode, the data is written to the end of the file's existing contents.

**10.15 design an algorithm that use a For loop to write the numbers 1 through 10 to a file.**

Declare Integer counter

Declare OutputFile myFile

Open myFile "myfile.dat"

For counter = 1 To 10

Write myFile, counter

End For

Close myFile

**10.16 What is the purpose of the eof function?**

The eof function determines whether the end of a file has been reached.

**10.17 is it acceptable for a program to attempt to read beyond the end of a file?**

No, this usually causes an error.

**10.18 What would it mean if the expression eof (myFile) were to return True?**

It would mean that the program has reached the end of the file associated with the name myFile.

**10.20 What is a record? What is a field?**

A record is a complete set of data that describes one item, and a field is a single piece of data within a record.

**10.21 describe the way that you use a temporary file in a program that modifies a record in a sequential access file.**

You copy all of the original file's records to the temporary file, but when you get to the record that is to be modified, you do not write its old contents to the temporary file. Instead, you write its new, modified values to the temporary file.

Then, you finish copying any remaining records from the original file to the temporary file.

**10.22 describe the way that you use a temporary file in a program that deletes a record from a sequential access file.**

You copy all of the original file's records to the temporary file, except for the record that is to be deleted. The temporary file then takes the place of the original file. You delete the original file and rename the temporary file, giving it the name that the original file had on the computer's disk.

This is a small “holding section” in memory that many systems write data to before writing the data to a file. (**buffer**)

This is a character or set of characters that marks the end of a piece of data. (**delimiter**)

This is a character or set of characters that marks the end of a file. (**EOF marker**)

This marks the location of the next item that will be read from a file. (**read position**)

**CH9 Sorting and Searching Arrays**

**9.1 Which of the sorting algorithms discussed makes several passes through an array and causes the larger values to move gradually toward the end of the array with each pass?**

The bubble sort

**9.2 One of the sorting algorithms discussed works like this: it begins by sorting the first two elements of the array, which becomes a sorted subset. Then the third element is moved to its correct position relative to the first two elements. At that point the first three elements become the sorted subset. This process continues with the fourth and subsequent elements until the entire array is sorted. Which algorithm is this?**

The insertion sort algorithm

**9.3 One of the sorting algorithms discussed works like this: the smallest value in the array is located and moved to element 0. Then the next smallest value is located and moved to element 1. This process continues until all of the elements have been placed in their proper order. Which algorithm is this?**

The selection sort algorithm

**9.4 describe the difference between a sequential search and a binary search.**

The sequential search algorithm simply uses a loop to step through each element of an array, comparing each element’s value with the value being searched for.

The binary search algorithm, which requires the values in the array to be sorted in order, starts searching at the element in the middle of the array. If the middle element’s value is greater than the value being searched for, the algorithm next tests the element in the middle of the first half of the array. If the middle element’s value is less than the value being searched for, the algorithm next tests the element in the middle of the last half of the array. Each time the array tests an array element and does not find the value being searched for, it eliminates half of the remaining portion of the array. This method continues until the value is found, or there are no more elements to test. The binary search is more efficient than the sequential search

**9.6 With an array of 1,000 elements, what is the maximum number of comparisons a binary search will perform?**

10

**CH14 Object-Oriented Programming**

**13.1 What is an object?**

An object is a software entity that contains both data and procedures.

**13.2 What is encapsulation封装?**

Encapsulation is the combining of data and code into a single object.

**13.3 Why is an object’s internal data usually hidden from outside code?**

When an object’s internal data is hidden from outside code and access to that data is restricted to the object’s methods, the data is protected from accidental corruption. In addition, the programming code outside the object does not need to know about the format or internal structure of the object’s data.

**13.4 What are public methods? What are private methods?**

Public methods can be accessed by entities outside the object. Private methods cannot be accessed by entities outside the object. They are designed to be accessed internally.

**13.7 What is an access specifier?**

A key word that specifies how code outside a class can access a field or method.

**13.8 What access specifier is commonly used with a class’s fields?**

Private

**13.9 When a class variable is said to reference an object, what is actually stored in the class variable?**

The memory address of the object that it references.

**13.10 What does the *New* key word do?**

It creates an object in the computer's memory.

**13.11 What is an accessor? What is a mutator?**

An accessor is a method that gets a value from a class’s field but does not change it. A mutator is a method that stores a value in a field or changes the value of a field in some other way.

**13.12 What is a constructor? When does a constructor execute?**

A constructor is a method that typically initializes an object's fields. A constructor executes when an object is created.

**13.13 What is a default constructor?**

If you do not write a constructor in a class, most languages automatically provide one when the class is compiled. The constructor that is automatically provided is usually known as the default constructor. Typically, the default constructor assigns default starting values to the object's fields.

**13.14 The typical UML diagram for a class has three sections. What appears in these three sections?**

The top section is where you write the name of the class. The middle section holds a list of the class’s fields. The bottom section holds a list of the class’s methods.

**13.15 Suppose a class has a field named *description*. The field’s data type is *String*. How would you indicate the field’s data type in a ULM diagram?**

By writing a colon followed by String after the name of the field. Here is an example: *description : String*

**13.16 What symbols do you use in a UML diagram to indicate private and public access specification?**

You use a minus sign (-) to indicate private specification, and a plus sign (+) to indicate public specification.

**13.17 What is a problem domain description?**

A written description of the real-world objects, parties, and major events related to the problem.

**13.18 What technique was described in this section for finding the classes in a particular problem?**

First, identify the nouns, pronouns, and pronoun phrases in the problem domain description. Then, refine the list to eliminate duplicates, items that you do not need to be concerned with in the problem, items that represent objects instead of classes, and items that represent simple values that can be stored in variables.

**13.19 What are classes’ responsibilities?**

The things that the class is responsible for knowing, and the actions that the class is responsible for doing.

**13.20 What causes an item of data to become stale陈旧?**

When the value of an item is dependent on other data and that item is not updated when the other data is changed, it is said that the item has become stale.

**13.21 Which is the general class and which is the specialized class?**

The superclass is the general class and the subclass is the specialized class.

**13.22 What does it mean to say there is an “is a” relationship between two objects?**

When an “is a” relationship exists between objects, it means that the specialized object has all of the characteristics of the general object, plus additional characteristics that make it special

**13.23 What does a subclass inherit from its superclass?**

The superclass's fields and methods, except those that are private.

**13.24 *Class Canary Extends Bird***

**What is the name of the superclass? What is the name of the subclass?**

Bird is the superclass and Canary is the subclass.

The **polymorphis**m characteristic of object-oriented programming allows a superclass variable to reference a subclass object.

**CH15 GUI Applications and Event-Driven Programming**

**14.1 What is a user interface?**

The part of a computer and its operating system that the user interacts with.

**14.2 How does a command line interface work?**

A command line interface typically displays a prompt, and the user types a command which is then executed.

**14.3 When the user runs a program in a text-based environment, such as the command line, what determines the order in which things happen?**

The program.

**14.4 What is an even-driven program?**

A program that responds to events that take place, such as the user clicking a button.

**14.5 What is a user interface flow diagram?**

A diagram that shows how the program flows from one window to the next as the user interacts with it.

**14.6 Why was GUI programming complex and time consuming in the early days of GUI technology?**

Because programmers had to write code that constructed the windows, created graphical elements such as icons and buttons, and set each element's color, position, size, and other properties. Even a simple GUI program that displays a message such as "Hello world" required the programmer to write a hundred or more lines of code. Furthermore, the programmer could not actually see the program's user interface until the program was compiled and executed.

**14.7 In an IDE that allows you to visually construct a window, how do you place an item such as a button in the window?**

You drag the item from a "toolbox" to the window editor.

**14.8 What is a component?**

An item that appears in a program's graphical user interface.

**14.9 Why must components be assigned names?**

A component's name identifies the component in the program, in the same way that a variable's name identifies the variable.

**14.10 What do a component’s properties do?**

A component's properties determine how the component appears on the screen.

**14.11 What is an event?**

An action that takes place within a program, such as the clicking of a button.

**14.12 What is an event handler?**

A module that automatically executes when a specific event occurs.