

### Algorithm Workbench

1. Write a `while` loop that lets the user enter a number. The number should be multiplied by 10, and the result assigned to a variable named `product`. The loop should iterate as long as `product` is less than 100.
2. Write a `while` loop that asks the user to enter two numbers. The numbers should be added and the sum displayed. The loop should ask the user if he or she wishes to perform the operation again. If so, the loop should repeat, otherwise it should terminate.
3. Write a `for` loop that uses the `range` function to display all odd numbers between 1 and 100.
4. Starting with a variable `text` containing an empty string, write a loop that prompts the user to type a word. Add the user's input to the end of `text` and then print the variable. The loop should repeat while the length of `text` is less than 10 characters.
5. Write a loop that calculates the total of the following series of numbers:

$$\frac{1}{30} + \frac{2}{29} + \frac{3}{28} + \cdots \frac{30}{1}$$

6. Rewrite the following statements using augmented assignment operators.
  - a. `x = x + 1`
  - b. `x = x * 2`
  - c. `x = x / 10`
  - d. `x = x - 100`
7. Write a set of nested loops that displays the first ten values of the multiplication tables from 1 to 10. The code should print 100 lines in total, starting at "`1 × 1 = 1`" and ending with "`10 × 10 = 100`".
8. Write code that prompts the user to enter a positive nonzero number and validates the input.
9. Write code that prompts the user to enter a number in the range of 1 through 100 and validates the input.

## Programming Exercises

### 1. Bug Collector

A bug collector collects bugs every day for five days. Write a program that keeps a running total of the number of bugs collected during the five days. The loop should ask for the number of bugs collected for each day, and when the loop is finished, the program should display the total number of bugs collected.

### 2. Calories Burned

Running on a particular treadmill you burn 4.2 calories per minute. Write a program that uses a loop to display the number of calories burned after 10, 15, 20, 25, and 30 minutes.

### 3. Lap Times

Write a program that asks the user to enter the number of times that they have run around a racetrack, and then uses a loop to prompt them to enter the lap time for each of their laps. When the loop finishes, the program should display the time of their fastest lap, the time of their slowest lap, and their average lap time.



VideoNote  
The Bug Collector  
Problem

#### 4. Distance Traveled

The distance a vehicle travels can be calculated as follows:

$$\text{distance} = \text{speed} \times \text{time}$$

For example, if a train travels 40 miles per hour for three hours, the distance traveled is 120 miles. Write a program that asks the user for the speed of a vehicle (in miles per hour) and the number of hours it has traveled. It should then use a loop to display the distance the vehicle has traveled for each hour of that time period. Here is an example of the desired output:

What is the speed of the vehicle in mph? **40**

How many hours has it traveled? **3**

Hour	Distance Traveled
1	40
2	80
3	120

#### 5. Average Rainfall

Write a program that uses nested loops to collect data and calculate the average rainfall over a period of years. The program should first ask for the number of years. The outer loop will iterate once for each year. The inner loop will iterate twelve times, once for each month. Each iteration of the inner loop will ask the user for the inches of rainfall for that month. After all iterations, the program should display the number of months, the total inches of rainfall, and the average rainfall per month for the entire period.

#### 6. Miles to Kilometers Table

Write a program that displays a table of distances in miles and their equivalent distances in kilometers, rounded to 2 decimal places. One mile is equivalent to 1.60934 kilometers. The table should be generated using a loop, and should include values in 10 mile increments from 10 to 80.

#### 7. Pennies for Pay

Write a program that calculates the amount of money a person would earn over a period of time if his or her salary is one penny the first day, two pennies the second day, and continues to double each day. The program should ask the user for the number of days. Display a table showing what the salary was for each day, then show the total pay at the end of the period. The output should be displayed in a dollar amount, not the number of pennies.

#### 8. Average Word Length

Write a program with a loop that repeatedly asks the user to enter a word. The user should enter nothing (press Enter without typing anything) to signal the end of the loop. Once the loop ends, the program should display the average length of the words entered, rounded to the nearest whole number.

#### 9. Ocean Levels

Assuming the ocean's level is currently rising at about 1.6 millimeters per year, create an application that displays the number of millimeters that the ocean will have risen each year for the next 25 years.

**10. Tuition Increase**

At one college, the tuition for a full-time student is \$8,000 per semester. It has been announced that the tuition will increase by 3 percent each year for the next 5 years. Write a program with a loop that displays the projected semester tuition amount for the next 5 years.

**11. Sleep Debt**

A “sleep debt” represents the difference between a person’s desirable and actual amount of sleep. Write a program that prompts the user to enter how many hours they slept each day over a period of seven days. Using 8 hours per day as the desirable amount of sleep, determine their sleep debt by calculating the total hours of sleep they got over the seven-day period and subtracting that from the total hours of sleep they should have got. If the user does not have a sleep debt, display a message expressing your jealousy.

**12. Calculating the Factorial of a Number**

In mathematics, the notation  $n!$  represents the factorial of the nonnegative integer  $n$ . The factorial of  $n$  is the product of all the nonnegative integers from 1 to  $n$ . For example,

$$7! = 1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 = 5,040$$

and

$$4! = 1 \times 2 \times 3 \times 4 = 24$$

Write a program that lets the user enter a nonnegative integer then uses a loop to calculate the factorial of that number. Display the factorial.

**13. Population**

Write a program that predicts the approximate size of a population of organisms. The application should use text boxes to allow the user to enter the starting number of organisms, the average daily population increase (as a percentage), and the number of days the organisms will be left to multiply. For example, assume the user enters the following values:

Starting number of organisms: 2

Average daily increase: 30%

Number of days to multiply: 10

The program should display the following table of data:

Day	Approximate Population
1	2
2	2.6
3	3.38
4	4.394
5	5.7122
6	7.42586
7	9.653619
8	12.5497
9	16.31462
10	21.209

**14. Write a program that uses nested loops to draw this pattern:**

```

*****
*****
*****
****
***
**
*

```

**15. Write a program that uses nested loops to draw this pattern:**

```

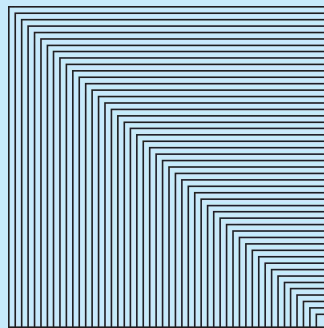
###
# #
# #
#  #
#   #
#    #

```

**16. Turtle Graphics: Repeating Squares**

In this chapter, you saw an example of a loop that draws a square. Write a turtle graphics program that uses nested loops to draw 100 squares, to create the design shown in Figure 4-13.

**Figure 4-13** Repeating squares

**17. Turtle Graphics: Star Pattern**

Use a loop with the turtle graphics library to draw the design shown in Figure 4-14.

**Figure 4-14** Star pattern

