REVIEW EXERCISES

■ **R2.1** Write declarations for storing the following quantities. Choose between integers and floating-point numbers. Declare constants when appropriate.

a. The number of days per week

b. The number of days until the end of the semester

c. The number of centimeters in an inch

d. The height of the tallest person in your class, in centimeters

■ **R2.2** What is the value of `mystery` after this sequence of statements?

```
int mystery = 1;
mystery = 1 - 2 * mystery;
mystery = mystery + 1;
```

■ **R2.3** What is wrong with the following sequence of statements?

```
int mystery = 1;
mystery = mystery + 1;
int mystery = 1 - 2 * mystery;
```

■■ **R2.4** Write the following mathematical expressions in C++.

$$s = s_0 + v_0 t + \frac{1}{2} g t^2 \qquad \mathrm{FV} = \mathrm{PV} \cdot \left(1 + \frac{\mathrm{INT}}{100}\right)^{\mathrm{YRS}}$$

$$G = 4\pi^2 \frac{a^3}{p^2(m_1 + m_2)} \qquad c = \sqrt{a^2 + b^2 - 2ab\cos\gamma}$$

■■ **R2.5** Write the following C++ expressions in mathematical notation.

a. `dm = m * (sqrt(1 + v / c) / sqrt(1 - v / c) - 1);`

b. `volume = PI * r * r * h;`

c. `volume = 4 * PI * pow(r, 3) / 3;`

d. `z = sqrt(x * x + y * y);`

■■ **R2.6** What are the values of the following expressions? In each line, assume that

```
double x = 2.5;
double y = -1.5;
int m = 18;
int n = 4;
```

a. `x + n * y - (x + n) * y`

b. `m / n + m % n`

c. `5 * x - n / 5`

d. `1 - (1 - (1 - (1 - (1 - n))))`

e. `sqrt(sqrt(n))`

■ **R2.7** What are the values of the following expressions? In each line, assume that

```
string s = "Hello";
string t = "World";
```

a. `s.length() + t.length()`

b. `s.substr(1, 2)`

c. `s.substr(s.length() / 2, 1)`

d. `s + t`

e. `t + s`

■ **R2.8** Find at least five *compile-time* errors in the following program.

```
#include iostream

int main();
{
   cout << "Please enter two numbers:"
   cin << x, y;
   cout << "The sum of << x << "and" << y
      << "is: " x + y << endl;
   return;
}
```

■■ **R2.9** Find at least four *run-time* errors in the following program.

```
#include <iostream>

using namespace std;

int main()
{
   int total;
   int x1;
   cout << "Please enter a number: ";
   cin >> x1;
   total = total + x1;
   cout << "Please enter another number: ";
   int x2;
   cin >> x2;
   total = total + x1;
   double average = total / 2;
   cout << "The average of the two numbers is "
      << average << "endl";
   return 0;
}
```

■ **R2.10** Explain the differences between `2`, `2.0`, `"2"`, and `"2.0"`.

■ **R2.11** Explain what each of the following program segments computes.

a.
```
int x = 2;
int y = x + x;
```

b.
```
string s = "2";
string t = s + s;
```

■■ **R2.12** Write pseudocode for a program that reads a word and then prints the first character, the last character, and the characters in the middle. For example, if the input is `Harry`, the program prints `H y arr`.

■■ **R2.13** Write pseudocode for a program that reads a name (such as `Harold James Morgan`) and then prints a monogram consisting of the initial letters of the first, middle, and last names (such as `HJM`).

■■■ **R2.14** Write pseudocode for a program that computes the first and last digits of a number. For example, if the input is `23456`, the program should print out `2` and `6`. *Hint:* `%`, `log10`.

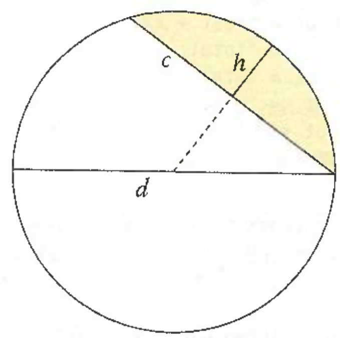
■ **R2.15** Modify the pseudocode for the program in How To 2.1 so that the program gives change in quarters, dimes, and nickels. You can assume that the price is a multiple of 5 cents. To develop your pseudocode, first work with a couple of specific values.

■■ **R2.16** A cocktail shaker is composed of three cone sections.

© Media Bakery.

The volume of a cone section with height h and top and bottom radius r_1 and r_2 is $V = \pi \frac{\left(r_1^2 + r_1 r_2 + r_2^2\right)h}{3}$. Compute the total volume by hand for one set of realistic values for the radii and heights. Then develop an algorithm that works for arbitrary dimensions.

■■■ **R2.17** You are cutting off a piece of pie like this, where c is the length of the straight part (called the chord length) and h is the height of the piece.



There is an approximate formula for the area: $A \approx \frac{2}{3}ch + \frac{h^3}{2c}$

However, h is not so easy to measure, whereas the diameter d of a pie is usually well-known. Calculate the area where the diameter of the pie is 12 inches and the chord length of the segment is 10 inches. Generalize to an algorithm that yields the area for any diameter and chord length.

■■ **R2.18** The following pseudocode describes how to obtain the name of a day, given the day number (0 = Sunday, 1 = Monday, and so on.)

Define a string called names containing "SunMonTueWedThuFriSat".
Compute the starting position as 3 x the day number.
Extract the substring of names at the starting position with length 3.

Check this pseudocode, using the day number 4. Draw a diagram of the string that is being computed, similar to Figure 4.

■■■ **R2.19** The following pseudocode describes how to swap two letters in a word.

We are given a string str and two positions i and j. (i comes before j)
Set first to the substring from the start of the string to the last position before i.
Set middle to the substring from positions i + 1 to j - 1.
Set last to the substring from position j + 1 to the end of the string.
Concatenate the following five strings: first, the string containing just the character at position j, middle, the string containing just the character at position i, and last.

Check this pseudocode, using the string `"Gateway"` and positions 2 and 4. Draw a diagram of the string that is being computed, similar to Figure 4.

■■ **R2.20** Run the following program, and explain the output you get.

```
#include <iostream>

using namespace std;

int main()
{
   int total;
   cout << "Please enter a number: ";
   double x1;
   cin >> x1;
   total = total + x1;
   cout << "total: " << total << endl;
   cout << "Please enter a number: ";
   double x2;
   cin >> x2;
   total = total + x2;
   cout << "total: " << total << endl;
   total = total / 2;
   cout << "total: " << total << endl;
   cout << "The average is " << total << endl;
   return 0;
}
```

Note the **trace messages** (in blue) that are inserted to show the current contents c the `total` variable. How do you fix the program? (The program has two separate errors.)

■■ **R2.21** How do you get the first character of a string? The last character? How do you remove the first character? The last character?

■■ **R2.22** For each of the following computations in C++, determine whether the result is exact, an overflow, or a roundoff error.

a. `2.0 - 1.1`

b. `1.0E6 * 1.0E6`

c. `65536 * 65536`

d. `1000000 * 1000000`

■■■ **R2.23** Write a program that prints the values

```
3 * 1000 * 1000 * 1000
3.0 * 1000 * 1000 * 1000
```

Explain the results.

■ **R2.24** This chapter contains a number of recommendations regarding variables and constants that make programs easier to read and maintain. Briefly summarize these recommendations.

PRACTICE EXERCISES

■ **E2.1** Write a program that displays the dimensions of a letter-size (8.5 × 11 inches) shee of paper in millimeters. There are 25.4 millimeters per inch. Use constants and con ments in your program.

■ **E2.2** Write a program that computes and displays the circumference of a letter-size (8.5 × 11 inches) sheet of paper and the length of its diagonal.

■ **E2.3** Write a program that reads a number and displays the square, cube, and fourth power. Use the `pow` function only for the fourth power.

■ **E2.4** Write a program that prompts the user for two integers and then prints
- The sum
- The difference
- The product
- The average

■■ **E2.5** Write a program that prompts the user for two integers and then prints
- The distance (absolute value of the difference)
- The maximum (the larger of the two)
- The minimum (the smaller of the two)

Hint: The `max` and `min` functions are defined in the `<algorithm>` header.

■■ **E2.6** Write a program that prompts the user for a measurement in meters and then converts it to miles, feet, and inches.

■ **E2.7** Write a program that prompts the user for a radius and then prints
- The area and circumference of a circle with that radius
- The volume and surface area of a sphere with that radius

■■ **E2.8** Write a program that asks the user for the lengths of the sides of a rectangle and then prints
- The area and perimeter of the rectangle
- The length of the diagonal (use the Pythagorean theorem)

■ **E2.9** Improve the program discussed in How To 2.1 to allow input of quarters in addition to bills.

■■ **E2.10** Write a program that asks the user to input
- The number of gallons of gas in the tank
- The fuel efficiency in miles per gallon
- The price of gas per gallon

Then print the cost per 100 miles and how far the car can go with the gas in the tank.

■ **E2.11** *File names and extensions.* Write a program that prompts the user for the drive letter (`C`), the path (`\Windows\System`), the file name (`Readme`), and the extension (`txt`). Then print the complete file name `C:\Windows\System\Readme.txt`. (If you use UNIX or a Macintosh, skip the drive name and use `/` instead of `\` to separate directories.)

■■ **E2.12** Write a program that reads a number between 1,000 and 999,999 from the user and prints it *with a comma separating the thousands.* Here is a sample dialog; the user input is in color:

```
Please enter an integer between 1000 and 999999: 23456
23,456
```

■■■ E2.13 Write a program that reads a number between 1,000 and 999,999 from the user, where the user enters a comma in the input. Then print the number without a comma. Here is a sample dialog; the user input is in color:

```
Please enter an integer between 1,000 and 999,999: 23,456
23456
```

Hint: Read the input as a string. Measure the length of the string. Suppose it conta *n* characters. Then extract substrings consisting of the first $n - 4$ characters and th last three characters.

■ E2.14 *Printing a grid.* Write a program that prints the following grid to play tic-tac-toe.

```
+--+--+--+
|  |  |  |
+--+--+--+
|  |  |  |
+--+--+--+
|  |  |  |
+--+--+--+
```

Of course, you could simply write seven statements of the form

```
cout << "+--+--+--+";
```

You should do it the smart way, though. Define string variables to hold two kinds (patterns: a comb-shaped pattern

```
+--+--+--+
|  |  |  |
```

and the bottom line. Print the comb three times and the bottom line once.

■■ E2.15 Write a program that reads an integer and breaks it into a sequence of individual digits. For example, the input 16384 is displayed as

```
1 6 3 8 4
```

You may assume that the input has no more than five digits and is not negative.

■■ E2.16 Write a program that reads two times in military format (0900, 1730) and prints the number of hours and minutes between the two times. Here is a sample run. User input is in color.

```
Please enter the first time: 0900
Please enter the second time: 1730
8 hours 30 minutes
```

Extra credit if you can deal with the case where the first time is later than the second:

```
Please enter the first time: 1730
Please enter the second time: 0900
15 hours 30 minutes
```

■■■ E2.17 *Writing large letters.* A large letter H can be produced like this:

```
*   *
*   *
*****
*   *
*   *
```

It can be defined as a string constant like this:

```
const string LETTER_H =
   "*   *\n*   *\n*****\n*   *\n*   *\n";
```

(The `\n` character is explained in Special Topic 1.1.) Do the same for the letters `E`, `L`, and `O`. Then write the message

```
H
E
L
L
O
```

in large letters.

© José Luis Gutiérrez/iStockphoto.

■■ E2.18 Write a program that transforms numbers `1`, `2`, `3`, …, `12` into the corresponding month names `January`, `February`, `March`, …, `December`. *Hint:* Make a very long string `"January February March ..."`, in which you add spaces such that each month name has *the same length*. Then use `substr` to extract the month you want.

PROGRAMMING PROJECTS

■■ P2.1 Easter Sunday is the first Sunday after the first full moon of spring. To compute the date, you can use this algorithm, invented by the mathematician Carl Friedrich Gauss in 1800:

1. Let `y` be the year (such as 1800 or 2001).
2. Divide `y` by `19` and call the remainder `a`. Ignore the quotient.
3. Divide `y` by `100` to get a quotient `b` and a remainder `c`.
4. Divide `b` by `4` to get a quotient `d` and a remainder `e`.
5. Divide `8 * b + 13` by `25` to get a quotient `g`. Ignore the remainder.
6. Divide `19 * a + b - d - g + 15` by `30` to get a remainder `h`. Ignore the quotient.
7. Divide `c` by `4` to get a quotient `j` and a remainder `k`.
8. Divide `a + 11 * h` by `319` to get a quotient `m`. Ignore the remainder.
9. Divide `2 * e + 2 * j - k - h + m + 32` by `7` to get a remainder `r`. Ignore the quotient.
10. Divide `h - m + r + 90` by `25` to get a quotient `n`. Ignore the remainder.
11. Divide `h - m + r + n + 19` by `32` to get a remainder `p`. Ignore the quotient.

Then Easter falls on day `p` of month `n`. For example, if `y` is `2001`:

```
a = 6            g = 6          m = 0    n = 4
b = 20, c = 1    h = 18         r = 6    p = 15
d = 5, e = 0     j = 0, k = 1
```

Therefore, in 2001, Easter Sunday fell on April 15. Write a program that prompts the user for a year and prints out the month and day of Easter Sunday.

■■■ P2.2 In this project, you will perform calculations with triangles. A triangle is defined by the *x*- and *y*-coordinates of its three corner points.

Your job is to compute the following properties of a given triangle:

- the lengths of all sides
- the angles at all corners
- the perimeter
- the area

Supply a program that prompts a user for the corner point coordinates and produ a nicely formatted table of the triangle properties.

■ Business P2.3 A video club wants to reward its best members with a discount based on the member's number of movie rentals and the number of new members referred by the member. The discount is in percent and is equal to the sum of the rentals and the referrals, but it cannot exceed 75 percent. (*Hint:* The `min` function in the `<algorithm>` header.) Write a program to calculate the value of the discount.

Here is a sample run:

```
Enter the number of movie rentals: 56
Enter the number of members referred to the video club: 3
The discount is equal to:     59.00 percent.
```

■■■ P2.4 Write a program that helps a person decide whether to buy a hybrid car. Your program's inputs should be:

- The cost of a new car
- The estimated miles driven per year
- The estimated gas price
- The estimated resale value after 5 years

Compute the total cost of owning the car for 5 years. (For simplicity, we will not take the cost of financing into account.) Obtain realistic prices for a new and used hybrid and a comparable car from the Web. Run your program twice, using today's gas price and 15,000 miles per year. Include pseudocode and the program runs with your assignment.

© asiseeit/iStockphoto.

■■ Business P2.5 The following pseudocode describes how a bookstore computes the price of an order from the total price and the number of the books that were ordered.

Read the total book price and the number of books.
Compute the tax (7.5% of the total book price).
Compute the shipping charge ($2 per book).
The price of the order is the sum of the total book price, the tax, and the shipping charge.

Print the price of the order. Translate this pseudocode into a C++ program.

■■ Business P2.6 The following pseudocode describes how to turn a string containing a ten-digit phone number (such as `"4155551212"`) into a more readable string with parentheses and dashes, like this: `"(415) 555-1212"`.

Take the substring consisting of the first three characters and surround it with "(" and ")". This is the area code.

Concatenate the area code, the substring consisting of the next three characters, a hyphen, and the substring consisting of the last four characters. This is the formatted number.

Translate this pseudocode into a C++ program that reads a telephone number into a string variable, computes the formatted number, and prints it.

■■ Business P2.7 The following pseudocode describes how to extract the dollars and cents from a price given as a floating-point value. For example, a price of 2.95 yields values 2 and 95 for the dollars and cents.

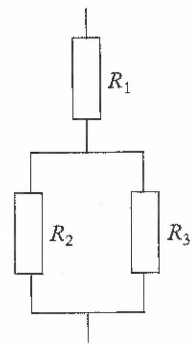
Assign the price to an integer variable dollars.
Multiply the difference price - dollars by 100 and add 0.5.
Assign the result to an integer variable cents.

Translate this pseudocode into a C++ program. Read a price and print the dollars and cents. Test your program with inputs 2.95 and 4.35.

■■ Business P2.8 *Giving change.* Implement a program that directs a cashier how to give change. The program has two inputs: the amount due and the amount received from the customer. Display the dollars, quarters, dimes, nickels, and pennies that the customer should receive in return.

© Captainflash/iStockphoto.

■ Engineering P2.9 Consider the following circuit.



Write a program that reads the resistances of the three resistors and computes the total resistance, using the rules for series and parallel resistors that are derived from Ohm's law.

■■ Engineering P2.10 The dew point temperature T_d can be calculated (approximately) from the relative humidity RH and the actual temperature T by

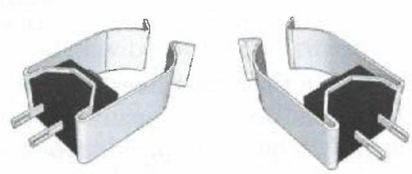
$$T_d = \frac{b \cdot f(T,RH)}{a - f(T,RH)}$$

$$f(T,RH) = \frac{a \cdot T}{b + T} + \ln(RH)$$

where $a = 17.27$ and $b = 237.7°$ C.

Write a program that reads the relative humidity (between 0 and 1) and the temperature (in degrees C) and prints the dew point value. Use the C++ `log` function to compute the natural logarithm.

■■■ Engineering P2.11 The pipe clip temperature sensors shown here are robust sensors that can be clipped directly onto copper pipes to measure the temperature of the liquids in the pipes.



Each sensor contains a device called a *thermistor*. Thermistors are semiconduct devices that exhibit a temperature-dependent resistance described by:

$$R = R_0\, e^{\beta\left(\frac{1}{T} - \frac{1}{T_0}\right)}$$

where R is the resistance (in Ω) at the temperature T (in °K), and R_0 is the resist (in Ω) at the temperature T_0 (in °K). β is a constant that depends on the material to make the thermistor. Thermistors are specified by providing values for R_0, T and β.

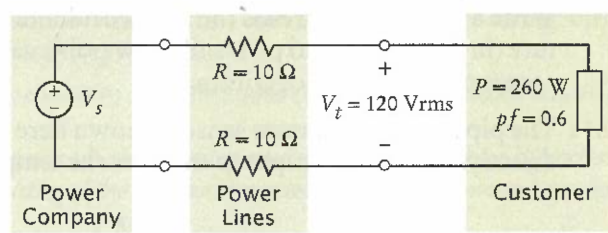
The thermistors used to make the pipe clip temperature sensors have $R_0 = 1075$ $T_0 = 85$ °C, and $\beta = 3969$ °K. (Notice that β has units of °K. Recall that the temp ture in °K is obtained by adding 273 to the temperature in °C.) The liquid temp ture, in °C, is determined from the resistance R, in Ω, using

$$T = \frac{\beta T_0}{T_0 \ln\left(\frac{R}{R_0}\right) + \beta} - 273$$

Write a C++ program that prompts the user for the thermistor resistance R and prints a message giving the liquid temperature in °C.

■■■ **Engineering P2.12** The circuit shown below illustrates some important aspects of the connection between a power company and one of its customers. The customer is represented by three parameters, V_t, P, and pf. V_t is the voltage accessed by plugging into a wall outlet. Customers depend on having a dependable value of V_t in order for their appliances to work properly. Accordingly, the power company regulates the value of V_t carefully. P describes the amount of power used by the customer and is the primary factor in determining the customer's electric bill. The power factor, pf, is less familiar. (The power factor is calculated the cosine of an angle so that its value will always be between zero and one.) In problem you will be asked to write a C++ program to investigate the significan the power factor.

© TebNad/iStockphoto.

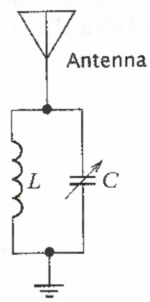


In the figure, the power lines are represented, somewhat simplistically, as resist in Ohms. The power company is represented as an AC voltage source. The sou voltage, V_s, required to provide the customer with power P at voltage V_t can be determined using the formula

$$V_s = \sqrt{\left(V_t + \frac{2RP}{V_t}\right)^2 + \left(\frac{2RP}{pf V_t}\right)^2 \left(1 - pf^2\right)}$$

(V_s has units of Vrms.) This formula indicates that the value of V_s depends on the value of *pf*. Write a C++ program that prompts the user for a power factor value and then prints a message giving the corresponding value of V_s, using the values for P, R, and V_t shown in the figure above.

▪▪▪ Engineering P2.13 Consider the following tuning circuit connected to an antenna, where C is a variable capacitor whose capacitance ranges from C_{min} to C_{max}.



The tuning circuit selects the frequency $f = \dfrac{1}{2\pi\sqrt{LC}}$. To design this circuit for a given frequency, take $C = \sqrt{C_{\text{min}} C_{\text{max}}}$ and calculate the required inductance L from f and C. Now the circuit can be tuned to any frequency in the range

$$f_{\text{min}} = \frac{1}{2\pi\sqrt{LC_{\text{max}}}} \text{ to } f_{\text{max}} = \frac{1}{2\pi\sqrt{LC_{\text{min}}}}.$$

Write a C++ program to design a tuning circuit for a given frequency, using a variable capacitor with given values for C_{min} and C_{max}. (A typical input is f = 16.7 MHz, C_{min} = 14 pF, and C_{max} = 365 pF.) The program should read in f (in Hz), C_{min} and C_{max} (in F), and print the required inductance value and the range of frequencies to which the circuit can be tuned by varying the capacitance.

▪ Engineering P2.14 According to the Coulomb force law, the electric force between two charged particles of charge Q_1 and Q_2 Coulombs, that are a distance r meters apart, is

$F = \dfrac{Q_1 Q_2}{4\pi\varepsilon r^2}$ Newtons, where $\varepsilon = 8.854 \times 10^{-12}$ Farads/meter. Write a program that calculates the force on a pair of charged particles, based on the user input of Q_1 Coulombs, Q_2 Coulombs, and r meters, and then computes and displays the electric force.

▪▪ Engineering P2.15 According to Newton's law of gravitation, the gravitational force between two masses M_1 and M_2 (in kilograms), that are a distance r meters apart, is $F = GM_1M_2 / r^2$, where $G = 6.67 \times 10^{-11} N(\text{m/kg})^2$. Write a program that calculates the force on a pair of masses, based on the user input of M_1, M_2, and r.

■■ Engineering P2.16 The relationship between angles measured in units of degrees and units of radians is π radians = 180°. Write a program that converts angles in degrees to angles in radians. The input is a positive integer.

Note: Make sure that numerical output is not greater than 2π radians or 360°. For example, 370° = 0.174533 radian.

■■ Engineering P2.17 The equation for the distance traveled by a freely falling object is $y = 1/2gt^2$, where t is time in seconds and g is the acceleration of gravity near the surface of the Earth, $g = 9.8$ m/s^2. Notice that this equation is independent of the mass of the object.

Create a program that outputs a sentence expressing the distance traveled in the given time. For example, an input of 10 seconds should have the output sentence "After the first 10 seconds, the object has fallen 490 meters."