SECTION 9.2 QUESTIONS

1. What is meant by the scope of a variable?
2. Describe one advantage to using local variables.
3. Where must a variable be declared in order for it to be local?
4. Where must a variable be declared in order for it to be global?
5. Describe what happens when a variable is passed by value?
6. When should you pass variables by reference?
7. What method is used to pass character arrays to functions?

PROBLEM 9.2.1

Modify *SERIES2.CPP* so that the calls to **get_choice** and **handle_choice** are in a do while loop. Add an item to the menu numbered 0 (zero) that exits the program. Have the loop continually redisplay the menu until zero is chosen. *Note:* Make sure you change the do while loop in the **get_choice** function so that zero is a valid input. Save the source code as *SERIES3.CPP*.

PROBLEM 9.2.2

Add a function to the *CTOF.CPP* program that asks the user for the temperature in Celsius and uses passing by reference to return the temperature to the main function. Save the source code as *CTOF2.CPP*.

CHAPTER 9, SECTION 3

Library Functions

C++ compilers include pre-written, ready-to-use functions to make programming easier. The number and type of functions available to you will vary depending on your compiler. The functions that come with your compiler are called ***library functions.*** This section shows you how to use some of the more common library functions.

USING LIBRARY FUNCTIONS

Library functions are just like functions you create and may be used in the same way. The difference is that the source code for library functions does not appear in your program. The prototypes for library functions are provided to your program using the **#include** compiler directive.

Let's examine a common C++ library function, **pow()**, which is used to raise a value (x) by the power (y). The **pow()** function prototype is shown below.

```
double pow(double x, double y);
```


The function pow receives two values or expressions of type double and returns the result as a double. Below is an example of a call to the pow function.

```
x_to_the_y = pow(x, y);
```

In order to use the pow function, you must include the **math.h** header file using the compiler directive below. A ***header file*** is a text file that provides the prototypes of a group of library functions. The linker uses the information in the header file to properly link your program with the function you want to use.

```
#include<math.h>
```

EXERCISE 9-9 USING LIBRARY FUNCTIONS

1. Write a program that prompts the user for two values of type double and uses the **pow** function to calculate x^y. Be sure to include the header files for **iostream.h** and **math.h**.
2. Add appropriate comments to your program to describe the parts of your program.
3. Save the program as *POWER.CPP*, compile, run, and test your program.
4. When you have completed the program and verified that it is working properly, close the source code file.

POPULAR MATH FUNCTIONS

Many C++ compilers provide basic math functions, such as the one you used to calculate x^y. Table 9-1 describes some basic math functions, and shows their prototypes and their purpose.

Note

All of the functions in Table 9-1 require that **math.h** *be included in the calling program.*

FUNCTION	PROTOTYPE	DESCRIPTION
abs	int abs(int x);	Returns the absolute value of an integer
labs	long int labs(long int x);	Returns the absolute value of a long integer
fabs	double fabs(double x);	Returns the absolute value of a floating-point number
ceil	double ceil(double x);	Rounds up to a whole number
floor	double floor(double x);	Rounds down to a whole number
hypot	double hypot(double a, double b);	Calculates the hypotenuse (c) of a right triangle where $c^2 = a^2 + b^2$
pow	double pow(double x, double y);	Calculates x to the power of y
pow10	double pow10(int x);	Calculates 10 to the power of x
sqrt	double sqrt(double x);	Calculates the positive square root of x

TABLE 9-1

EXERCISE 9-10 USING sqrt

1. Write a program that uses the **sqrt** function to calculate the circumference of a circle, given its area. Use the formula $2\sqrt{\pi * area}$.
2. Save the program as *CIRC.CPP* and close the source code when you have completed the exercise.