9.8 Practice Programs

Sample solutions to these problems will be included by the beginning of 2013 at the following website:

http://www.cs.ucf.edu/~dmarino/ucf/transparency/cop3223/book/

Each of the following problems asks you to write a function. In order to test your function, you must write a main function in addition that calls the function you've written. Make sure to test the function you've written with a comprehensive set of test cases.

1) Write a function that takes in a positive integer and returns 1 if its prime and 0 otherwise. The prototype is given below. Note that the first few prime numbers are 2, 3, 5 and 7. Thus, if n is 1, your function should return 0.

```
int isPrime(int n);
```

2) Write a function that converts its input parameter (a double) from a temperature in Fahrenheit to a temperature in Celsius. The prototype is given below:

```
double convertToCelsius(double fahr);
```

3) Write a function that converts its input parameter (a double) from a temperature in Celsius to a temperature in Fahrenheit.

```
double convertToFahrenheit(double fahr);
```

4) Write a function that calculates how many times a particular positive integer, base, divides evenly into a second positive integer, total. For example the integer 2 divides into 96 five times since $2^5 = 32$ and 96 is divisible by 32, and since $2^6 = 64$ and 96 is NOT divisible by 64. Use the function prototype given below:

```
int numTimesFactor(int base, int total);
```

5) Write a function that takes in a single positive integer n and returns n! (n factorial). Note that n! = 1x2x3x4...xn. You may assume $n \leq 12$. Use the function prototype given below:

```
int factorial(int n);
```

6) The Fibonacci sequence is defined as follows: F(0)=0, F(1)=1, F(n)=F(n-1)+F(n-2), for all integers n > 1. Thus, the first few terms of the sequence are: 0, 1, 1, 2, 3, 5, 8, etc. Write a function that takes in a non-negative integer n and returns the n^{th} Fibonacci number. You may assume $n \leq 46$. Use the function prototype given below:

```
int fibonacci(int n);
```

7) Write a function that takes in the coefficients a (not 0), b, and c (all doubles) to a quadratic equation and returns the smaller of the two roots (a double) as the result. You may assume that the roots of the quadratic are real. The formula for the roots of a quadratic is given below:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

8) Write a void function that prints out a design of a specified character in the shape of a diamond. A diamond is specified by a positive odd integer, *n*. The diamond should have *n* rows, where the middle row has *n* stars. The first and last rows have 1 star each and each row in the first half of the diamond has two more stars in it than the previous row, while each row in the second half of the diamond has two fewer stars in it than the previous row. Here is the diamond that should be printed for n = 7:

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

Here is the function prototype:

```
void printDiamond(char c, int n);
```

9) Write a function that in two positive integers: *numProblems* and *maxNumber*, and plays a multiplication game with *numProblems* multiplication problems that are randomly generated where each number in the multiplication problems are in between 1 and *maxNumber*, inclusive. Your function should return the number of responses the user gets correct. In writing the function, assume that the random number generator has already been seeded and that you can just use the rand() function. The prototype is given below:

```
int multGame(int numProblems, int maxNumber);
```

10) Write a function that takes in six integer parameters: x1, y1, r1, x2, y2 and r2, where (x1,y1) is the center of a circle with radius r1 and (x2, y2) is the center of a circle with radius r2. Your program should return the number of intersection points between the two circles. (Hint: This number will always be 0, 1 or 2.) The function prototype is included for you below:

```
int numCross(int x1, int y1, int r1, int x2, int y2, int r2);
```