Exercises (Functions)

RETURNING A SINGLE VALUE

1) Consider the following function definition

```
double funcTest(string name, char u, int num, double y)
{
      ...
      ...
      ...
}
```

Which of the following are correct function declarations of the function `funcTest`?

a) `double funcTest(string, char, int, double)`
b) `double funcTest(string n, char ch, int x, double t)`
c) `double funcTest(name, u, num, y)`
d) `int funcTest(string, char, int, double)`

(reproduced from Qn 5, Cp 6, D. S. Malik, 8th ed).

2) Consider the following statements

```
int num1, num2, num3;
double length, width, height;
double volume;

num1 = 6; num2 = 7; num3 = 4;
length = 6.2; width = 2.3; height = 3.4;
```

and the function declaration:

```
double box(double, double, double);
```

Which of the following statements are valid? If they are invalid, explain why.

a) `volume = box(length, width, height);`
b) `volume = box(length, 3.8, height);`
c) `cout << box(num1, num3, num2) << endl;`
d) `cout << box(length, width, 7.0) << endl;`
e) `volume = box(length, num1, height);`
f) `cout << box(6.2, ,height) << endl;`
g) `volume = box(length + width, height);`
h) `volume = box(num1, num2 + num3);`

(reproduced from Qn 7, Cp 6, D. S. Malik, 8th ed).

3) Write a function `findMin` that returns the smaller between two values, `x` and `y`.

4) Modify Qn 3 so that the value of `x` and `y` are entered when the program is running.

5) Modify Qn 4 to repeat the process for n times (determined by user, for example, $n = 3$).

6) Modify Qn 3 to determine the smallest between 3 values.

7) Write a function that takes as input an integer and outputs two times the number if it is even; otherwise it outputs five times the number (adapted from Qn 23, Cp 6, D. S. Malik, 8th ed).

8) Write a C++ program named `cylvol` of type double that accepts a cylinder's radius r and its length h and return its volume v. Test the function by passing various data to it and verifying the returned value using the formula $v = \pi r^2 h$ (adapted from Qn 5, Ex 6.2, G. Bronson, 9th ed).

9) Write a C++ program named `spherevol` of type double that accepts a sphere's radius r and then displays its volume r. Then, rewrite your program by making `spherevol` a function of type void. Test the function by passing various data to it and verifying the returned value using the formula $v = 4\pi r^3/3$ (adapted from Qn 10, Ex 6.1, G. Bronson, 9th ed).

10) Write a function that `oddEvenFun` that accepts an integer, which then determines whether the passed integer is even or odd (Hint, use the % module operator) (adapted from Qn 12, Ex 6.1, G. Bronson, 9th ed).

11) Write a function definition for `whole` that returns the integer part of any number passed to the function. This can be done by assigning the passed argument (value) to an integer variable. Then, write a full C++ program. Test the function by passing various data to it and verifying the returned value (adapted from Qn 12, Ex 6.2, G. Bronson, 9th ed).

12) Using the `whole` function used in Qn 11, write a C++ program named `fracpart` that returns the fractional part of any number passed to it. For example, if the number 256.879 is passed to `fracpart`, the number 0.879 should be returned (adapted from Qn 13, Ex 6.2, G. Bronson, 9th ed).

13) Write a program that reads a word and outputs the number of vowel appears in it. Your program must contain a function with one of its parameters as a string variable (to read the word), and return the number of vowel appears in it. (Note that if `str` is a variable of type string, then `str.at(i)` returns the character at the ith position. The position of the first character is 0. Also, `str.length()` returns the length of the `str`, that is, the number of characters in `str`.) (adapted from Qn 8, Cp 7, D. S. Malik, 5th ed).

14) Modify Qn 13 to count and print the number of consonants in a word.

15) Modify Qns 3 to 14 using void functions to print the output.

RETURNING MULTIPLE VALUES

16) Write parameter declarations for the following
    a) A parameter named `amount` that will be a reference to a double-precision value.
    b) A parameter named `price` that will be a reference to a double-precision number.
    c) A parameter named `minutes` that will be a reference to an integer number.
    d) A parameter named `key` that will be a reference to a character.
    e) A parameter named `yield` that will be a reference to a double-precision number.

17) Write a function `findMax()`which stores the maximum value of the two passed numbers called `max`, where the value of `max` should be set from within `findMax()`. (Hint: A reference to `max` has to be accepted by `findMax()`(adapted from Qn 3, Ex 6.3, G. Bronson, 9th ed).

18) [Similar to Qns 13 and 14] Write a function `classifyWord` that reads a word and outputs the numbers of vowels and consonants appear in it. These two values should be set from within the function.

19) Write a function named `time()` that has an integer parameter named `seconds` and three integer reference parameters named `hours`, `mins`, and `secs`. The function is to convert the passed number

of seconds into an equivalent number of hours, minutes, and seconds. Using the reference parameters, the function should alter the arguments in the calling function (from Qn 5, Ex 6.3, G. Bronson, 9th ed).

20) Use functions to write the program that determines the number of odds (variable `odds`), evens (variable `evens`) and zeros (variable `zeros`) from a given list of integers. Note that in this question, we take 0 as an even number (please see the sample run below for more details). The functions will do the followings:

i. Initialize the variables to 0. This is done by a function `initialize`.

ii. Read a number. This is done by a function `getNumber`.

iii. If the number is even, increment the even count, and if the number is also zero, increment the zero count; else, increment the odd count. This is done by a function `classifyNumber.` This function also increments the appropriate count.

iv. Repeat Steps ii and iii for each number in the list.

v. Print the results. This is done by a function `printResults`.

Sample Run: In this sample run, the user input is shaded.

```
Please enter 20 integers.
0 0 12 23 45 7 -2 -8 -3 -9 4 0 1 0 -7 23 -24 0 0 12
There are 12 evens, which includes 6 zeros
The number of odd numbers is: 8
```

(reproduced from Programming Example, Cp 7, D. S. Malik, 5th ed).