Lab Exercise 2

Exercise 1

Write a value-returning function, *isVowe*l, that returns the value true if a given character is a vowel and otherwise returns false.

Exercise 2

Write a function, *reverseDigit*, that takes an integer as a parameter and returns the number with its digits reversed. For example, the value of *reverseDigit*(12345) is 54321; the value of *reverseDigit*(5600) is 65; the value of *reverseDigit*(7008) is 8007; and the value of *reverseDigit*(-532) is -235.

Exercise 3

Write a program that prompts the user to input a string. The program then uses the function substr to remove all the vowels from the string. For example, if str = "There", then after removing all the vowels, str = "Thr". After removing all the vowels, output the string. Your program must contain a function to remove all the vowels and a function to determine whether a character is a vowel.

Exercise 4

Write a C++ program that declares an array alpha of 50 components of type double. Initialize the array so that the first 25 components are equal to the square of the index variable, and the last 25 components are equal to three times the index variable. Output the array so that 10 elements per line are printed.

Exercise 5

Write a program that reads students' names followed by their test scores. The program should output each student's name followed by the test scores and the relevant grade. It should also find and print the highest test score and the name of the students having the highest test score.
Student data should be stored in a struct variable of type studentType, which has four components: studentFName and studentLName of type string, testScore of type int (testScore is between 0 and 100), and grade of type char. Suppose that the class has 20 students. Use an array of 20 components of type studentType.
Your program must contain at least the following functions:
a. A function to read the students' data into the array.
b. A function to assign the relevant grade to each student.
c. A function to find the highest test score.
d. A function to print the names of the students having the highest test score.
Your program must output each student's name in this form: last name followed by a comma, followed by a space, followed by the first name; the name must be left justified. Moreover, other than declaring the variables and opening the input and output files, the function main should only be a collection of function calls.