OOP With C++ (Assignment -3)
CSE, 3rd Semester
Deadline August 30, 2020

Prepared By: Deepak Uniyal (Assistant Professor CSE, GEU)

Note -

- Create your GitHub profile as taught in lectures and then push all your programs to folders named according to assignment. Example - when you push codes of this assignment, they should be inside Assignment3 folder.
- Please keep in mind that you don't commit all the codes together. Keep on committing codes module wise or question wise whatever seems available.

1. Modify the Complex class program to enable input and output of complex numbers via overloaded >> and << operators, respectively.
2. Overload the multiplication operator to enable multiplication of two complex numbers as in algebra.
3. Overload the == and != operators to allow comparisons of complex numbers.
4. Write a complete program that prompts the user for the radius of a sphere, and calculates and prints the volume of that sphere. Use an inline function sphereVolume that returns the result of the following expression: (4.0 / 3.0 * 3.14159 * pow(radius, 3)).
5. A parking garage charges a $2.00 minimum fee to park for up to three hours. The garage charges an additional $0.50 per hour for each hour or part thereof in excess of three hours. The maximum charge for any given 24-hour period is $10.00. Assume that no car parks for longer than 24 hours at a time. Write a program that calculates and prints the parking charges for each of three customers who parked their cars in this garage yesterday. You should enter the hours parked for each customer. Your program should print the results in a neat tabular format and should calculate and print the total of yesterday's receipts. The program should use the function calculateCharges to determine the charge for each customer. Your outputs should appear in the following format:

Car	Hours	Charge
1	1.5	2.00
2	4.0	2.50
3	24.0	10.00
Total	29.5	14.50

6. Create a class called Invoice that a hardware store might use to represent an invoice for an item sold at the store. An Invoice should include four data members—a part number (type string), a part description (type string), a quantity of the item being purchased (type int) and a price per item (type int). Your class should have a constructor that initializes the four data members. Provide a set and a get function for each data member. In addition, provide a member function named getInvoiceAmount that calculates the invoice amount (i.e., multiplies the quantity by the price per item), then returns the amount as an int

value. If the quantity is not positive, it should be set to 0. If the price per item is not positive, it should be set to 0. Write a test program that demonstrates class Invoice's capabilities.

7. Modify the Program-6 to separate the interface and implementation for the purpose of reusability. You should break the program into three different parts as discussed in lectures.
8. A right triangle can have sides that are all integers. A set of three integer values for the sides of a right triangle is called a Pythagorean triple. These three sides must satisfy the relationship that the sum of the squares of two of the sides is equal to the square of the hypotenuse. Find all Pythagorean triples for side1, side2 and hypotenuse all no larger than 500. Use a triple-nested for loop that tries all possibilities. This is an example of brute force computing. You'll learn in more advanced computer science courses that there are many interesting problems for which there's no known algorithmic approach other than sheer brute force.
9. A prime integer is any integer that is evenly divisible only by itself and 1. The Sieve of Eratosthenes is a method of finding prime numbers. It operates as follows:
    a. Create an array with all elements initialized to 1 (true). Array elements with prime sub- scripts will remain 1. All other array elements will eventually be set to zero. You'll ignore elements 0 and 1 in this exercise.
    b. Starting with array subscript 2, every time an array element is found whose value is 1, loop through the remainder of the array and set to zero every element whose subscript is a multiple of the subscript for the element with value 1. For array subscript 2, all elements beyond 2 in the array that are multiples of 2 will be set to zero (subscripts 4, 6, 8, 10, etc.); for array subscript 3, all elements beyond 3 in the array that are multiples of 3 will be set to zero (subscripts 6, 9, 12, 15, etc.); and so on.

    When this process is complete, the array elements that are still set to one indicate that the subscript is a prime number. These subscripts can then be printed. Write a program that uses an array of 1000 elements to determine and print the prime numbers between 2 and 9999. Ignore element 0 of the array.
10. A palindrome is a string that is spelled the same way forward and backward. Examples of palindromes include "radar" and "able was i ere i saw elba." Write a recursive function testPalindrome that returns true if a string is a palindrome, and false otherwise. Note that like an array, the square brackets ([]) operator can be used to iterate through the characters in a string.
11. Write a recursive function printArray that takes an array, a starting sub- script and an ending subscript as arguments, returns nothing and prints the array. The function should stop processing and return when the starting subscript equals the ending subscript.
12. Write a recursive function stringReverse that takes a string and a starting subscript as arguments, prints the string backward and returns nothing. The function should stop processing and return when the end of the string is encountered. Note that like an array the square brackets ([]) operator can be used to iterate through the characters in a string.
13. Write a recursive function recursiveMinimum that takes an integer array, a starting subscript and an ending subscript as arguments, and returns the smallest element of the array. The function should stop processing and return when the starting subscript equals the ending subscript.

Reference - C++ How To Program By Deitel and Deitel