**Lab 05 – Operator Overloading (Contd.)**

1. **Objectives**

The objective of this lab is to teach the students, the use of special binary operators i.e. comparison and compound assignment operators for user defined classes i.e. operator overloading.

1. **Outcome** 
   1. At the end of this lab student will know the purpose of Operator Overloading.
   2. Student will be able to use of special binary operators for user defined classes.
2. **Introduction**

One of the nice features of C++ is that you can give special meanings to operators, when they are used with user-defined classes. This is called operator overloading. You can implement C++ operator overloads by providing special member-functions on your classes that follow a particular naming convention. For example, to overload the + operator for your class, you would provide a member-function named operator+ on your class.

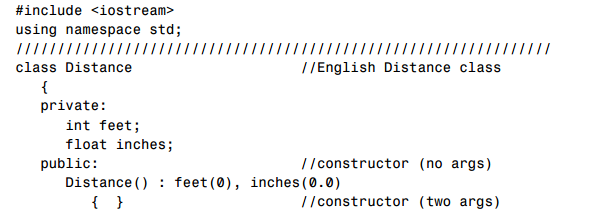
The following set of operators is commonly overloaded for user-defined classes:

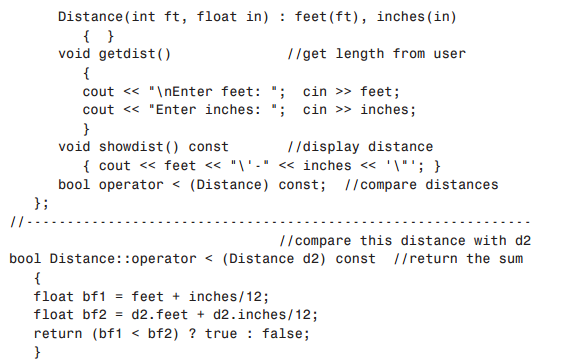
1. **Examples**

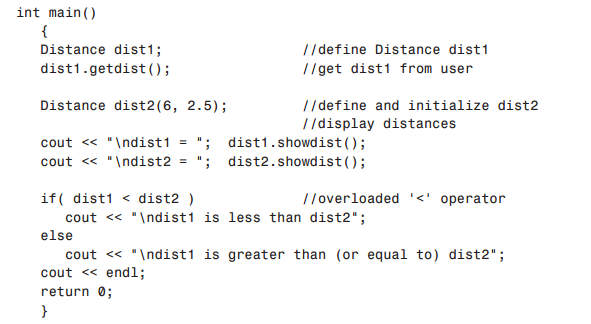
|  |  |
| --- | --- |
| ++ -- (Increment and Decrement Operators) | Unary Operators |
| = (Assignment Operator)  + - \* (Binary Arithmetic Operators) |  |
| += -= \*= (Compound Assignment Operators)  == != < > (Comparison Operators) | Binary Operators |

* 1. **Examples of comparison Operators Overloading:**

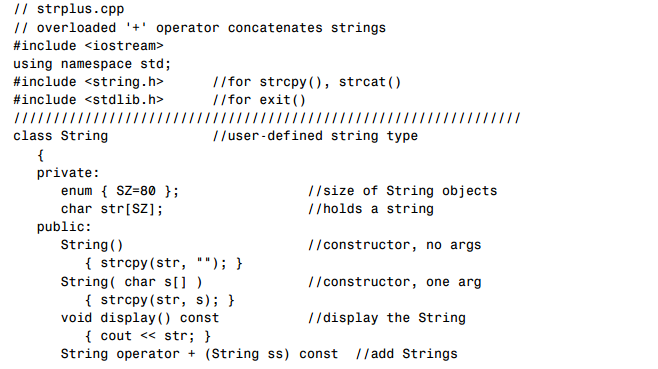
In our **first example** we’ll overload the *less than* operator (<) in the Distance class so that we can compare two distances.

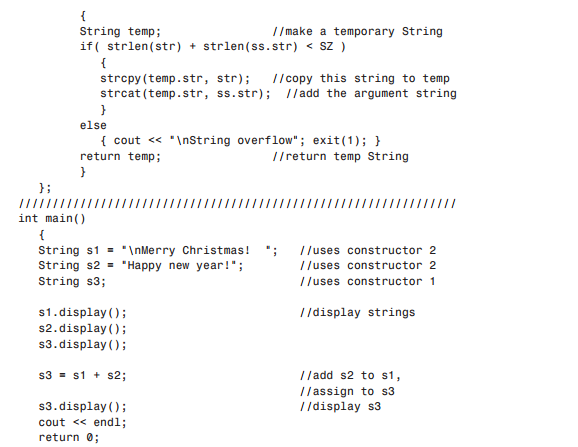






**4.2 Example of Concatenating Strings:**





1. **In-Lab Tasks**

**5.1** Use the Arithmetic Assignment operator (+=) for a Distance class to add one distance to a second, leaving the result in the first. This is similar to example shown earlier, but there is a subtle difference.

**5.2** Write a program that substitutes an overloaded += operator for the overloaded + operator in the example given above. This operator should allow statements like s1 += s2; where s2 is added (concatenated) to s1 and the result is left in s1. The operator should also permit the results of the operation to be used in other calculations, as in s3 = s1 += s2;

**5.3** Write a program for class String that uses an overloaded == operator for comparing two strings together. This operator should allow statements like s1 == s2; and display the results showing whether the strings entered by user are same or not.

1. **Post Lab Tasks:**
   1. Write a program for class time which has the ability to subtract two time values using the overloaded (- ) operator, and to multiply a time value by a number of type float, using the overloaded (\*) operator.
   2. Create a class Cartesian such that it compares the x and y co-ordinates of two points in a plain using an overloaded operator and after comparison print out which one of the point will be the head of the vector and which one will be tail if a vector would be formed between these two points keeping that a vector will go from smallest to largest point. Also check if both points are at the same place using overloaded operator then print out an error message “Both points are same…Resultant vector can’t be created”.