**Operator Overloading**

**Implement the following exercises.**

1. Write a class **Time** which represents time. The class should have three fields for hours, minutes and seconds. It should have constructor to initialize the hours, minutes and seconds. A method printTime() to print the current time.

Overload the following operators:

plus operator (+) (add two time objects based on 24 hour clock)  
and < (compare two time objects)

1. Define a class **Set** that stores a finite set of integers. Supply add ( ) and remove ( ) member functions  to add and remove set elements. Overload the | and & operators to compute the union and  intersection of the set.
2. Complete the following tasks:

a. Design a **Meal** class with two fields—one that holds the name of the entrée, the other that holds a calorie count integer. Include a constructor that sets a Meal’s fields with parameters, or uses default values when no parameters are provided.

b. Include an overloaded insertion operator function that displays a Meal’s values.

c. Include an overloaded extraction operator that prompts a user for an entrée name and calorie count for a meal.

d. Include an overloaded operator+()function that allows you to add two or more Meal

objects. Adding two Meal objects means adding their calorie values and creating a summary Meal object in which you store “Daily Total” in the entrée field.

e. Write a main()function that declares four Meal objects named breakfast, lunch,

dinner, and total. Provide values for the breakfast, lunch, and dinner objects.

Include the statement total = breakfast + lunch + dinner; in your program,

then display values for the four Meal objects. Save the file as **Meal.cpp**.

f. Write a main()function that declares an array of 21 Meal objects. Allow a user to enter values for 21 Meals for the week. Total these meals and display the calorie total for the end of the week. (*Hint*: You might find it useful to create a constructor for the Meal class.) Save the file as **Meal2.cpp**.

1. Complete the following tasks:

a. Design a PhoneCall class that holds a phone number to which a call is placed, the length

of the call in minutes, and the rate charged per minute. Overload extraction and insertion

operators for the class.

b. Overload the == operator to compare two PhoneCalls. Consider one PhoneCall to be

equal to another if both calls are placed to the same number.

c. Create a main()function that allows you to enter 10 PhoneCalls into an array. If a

PhoneCall has already been placed to a number, do not allow a second PhoneCall to the

same number. Save the file as **PhoneCall.cpp**.

1. Complete the following tasks:

a. Design a SoccerPlayer class that includes three integer fields: a player’s jersey number,

number of goals, and number of assists. Overload extraction and insertion operators

for the class.

b. Include an operator>()function for the class. One SoccerPlayer is considered greater

than another if the sum of goals plus assists is greater.

c. Include overloaded stream operators to read and display the player instance.

d. Include operator =( ) function to create temporary object.

e. Create an array of 11 SoccerPlayers, then use the > operator to find the player who has

the greatest total of goals plus assists. Save the file as **SoccerPlayer.cpp**.