Employee

Programmer Manual
Employee

1. Problem Description

The Employee class consists of information about a single employee, including first and last names, social security number, age, number of dependents, employee code, pay rate, and the number of hours worked. The class also contains overloaded operators to allow for comparisons necessary to be placed into a binary search tree. Accessors and mutator functions also exist to allow for access to and manipulation of employee data.

2. Class Employee

Private data members:

char fName[]	first name
char lName[]	last name
char ssn[]	social security number
int age	age
int dependents	number of dependents
char code	employee code
double payRate	pay rate
double hoursWorked	number of hours worked

Public member functions:

Employee	constructor for an Employee object

Operator overloads:

operator>>	stream extraction operator overload
operator<<	stream insertion operator overload
operator=	assignment operator overload
operator<	less than operator overload
operator>	greater than operator overload
operator<=	less than or equal to operator overload
operator>=	greater than or equal to operator overload
operator==	equality operator overload
operator!=	inequality operator overload

Mutators:

setFName	mutator for fName
setLName	mutator for LName
setSSN	mutator for ssn
setAge	mutator for age
setDependents	mutator for dependents
setCode	mutator for code
setPayRate	mutator for payRate
setHoursWorked	mutator for hoursWorked

Accessors:

getLastName	accessor for lName
getCode	accessor for code
getPayRate	accessor for payRate
getHoursWorked	accessor for hoursWorked

3. High Level Program Solution

Employee

set all employee information to default values

operator>>

get employee information from an input stream
set information to the current employee
return input stream

operator<<

print out each set of information for the employee
return output stream

operator=

copy the information from a right operand employee into the left operand employee
return left operand employee

operator<

if the left operand employee last name is less than the right operand last name, return true
otherwise return false

operator>

if the left operand employee last name is greater than the right operand last name, return true
otherwise return false

operator<=

if the left operand employee last name is less than or equal to the right operand last name, return true
otherwise return false

operator>=

if the left operand employee last name is greater than or equal to the right operand last name, return true
otherwise return false

operator==

if the left operand employee last name is equal to the right operand last name, return true
otherwise return false

operator!=
    if the left operand employee last name is not equal to the right operand last name, return true
    otherwise return false

setFName
    copy the input string into the employee's first name

setLName
    copy the input string into the employee's last name

setSSN
    copy the input string into the employee's social security number

setAge
    copy the input int into the employee's age

setDependents
    copy the input int into the employee's number of dependents

setCode
    copy the input char into the employee's employee code

setPayRate
    copy the input double into the employee's pay rate

setHoursWorked
    copy the input double into the employee's number of hours worked

getLastName
    return employee's last name

getHoursWorked
    return employee's number of hours worked

getPayRate
    return employee's pay rate

getCode
    return employee's employee code