



Chapter 15

Inheritance, Polymorphism, Virtual Functions

Spring 2013

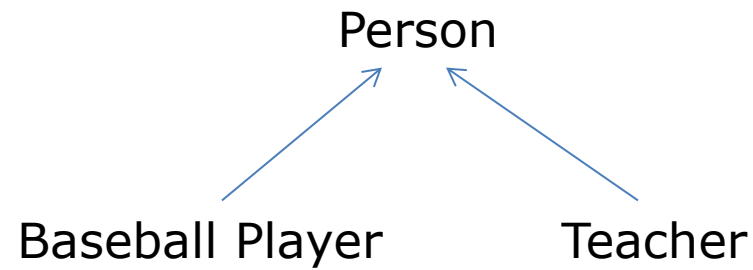
Key Terminology

- Inheritance
- Superclass
- Subclass
- Base class
- Derived class
- is-a relationship
- Composition
- has-a relationship

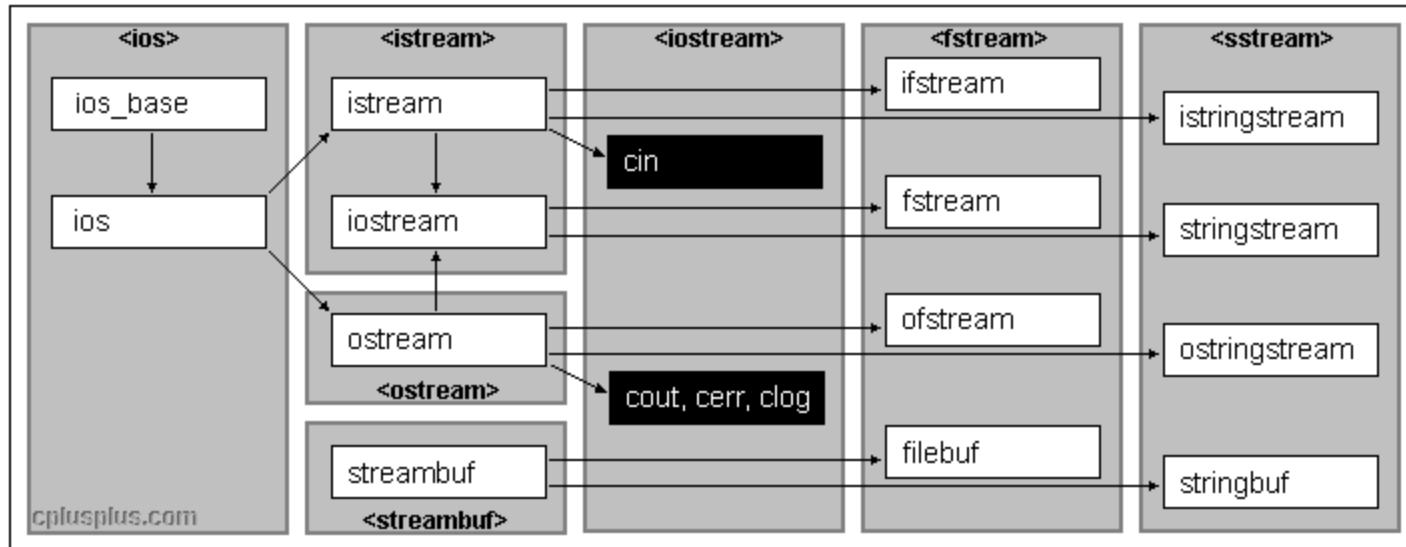
Inheritance

- Allows a new class to be based on an existing class (reusability)
- The new class inherits
 - all member variables
 - all member functions (excluding the constructors and destructor)
- The new class then adds new functionality

Simple Inheritance



More Complex Inheritance



<http://www.cplusplus.com/reference/ios/>

Inheritance and the is-a relationship

- A car is-a vehicle
- A rectangle is-a shape
- An athlete is-a person
- A football player is-a athlete

OOP Terminology

- Inheritance is used to create an is-a relationship
- Inheritance involves a base class (also called a superclass or parent class) and a derived class (also called a subclass or a child class)
- base class is the existing class
- derived class is the new class with inherited members and additional behaviors

Simple Inheritance Example

```
class Employee
{
    public:
        Employee (string = "", string = "");
        string getName () const;
        string getSSN () const;
        void setName (string);
        void setSSN (string);
        void print (ostream &) const;

    private:
        string mName;
        string mSSN;
};
```


HourlyEmployee

- An hourly employee is an employee that
 - earns an hourly wage rate
 - works a certain number of hours
- What additional member variables and behavior is necessary for an hourly employee?

Public Inheritance

- Every derived class object is also an object of the superclass.
- As an example, if the superclass is "Vehicle" then a subclass might be "Cars" and "Trucks." Cars inherit the members and behaviors of a Vehicle and add other behaviors and members
- Members of a subclass cannot directly access the private members of a superclass

Let's Play

1. Grab the solution Inheritance from CS250 Public and let's go through the initial setup
2. Write additional set and get methods for HourlyEmployee
3. Create a subclass (derived class) called SalariedEmployee for an employee that works on a yearly salary model. Write the interface and implementation for SalariedEmployee.

Let's Play

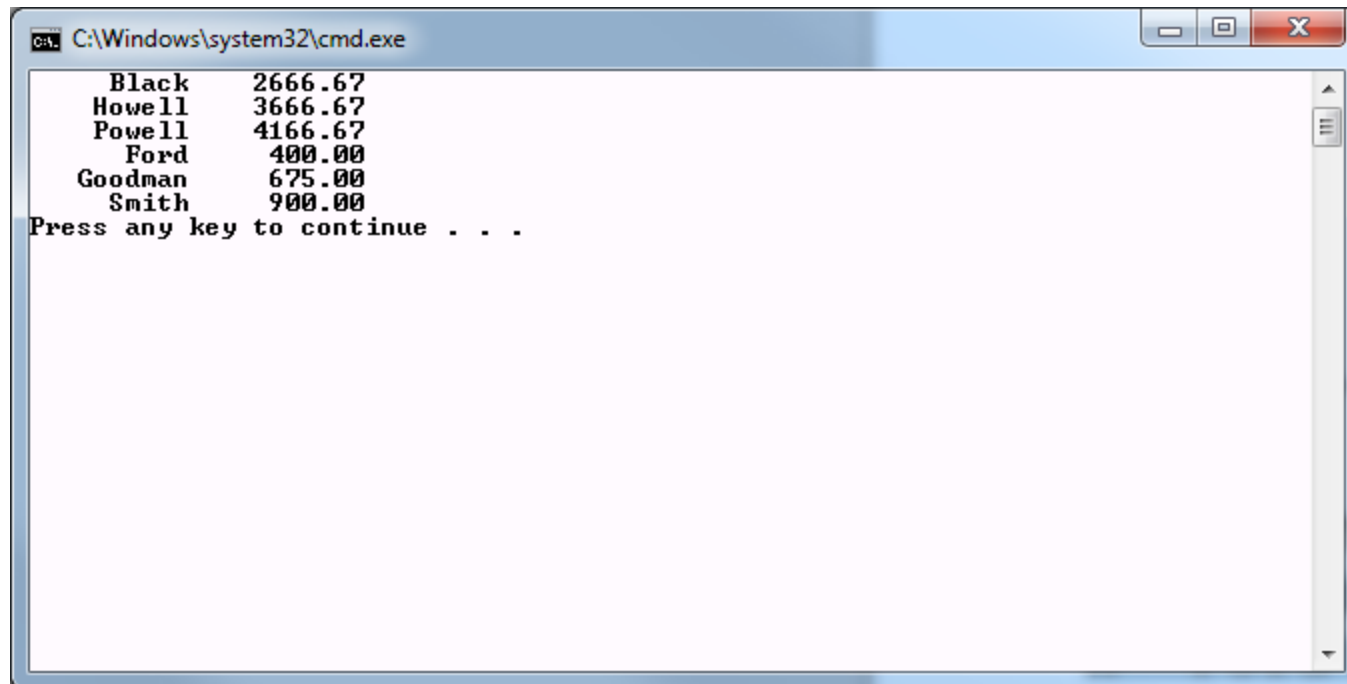
4. Create the following data file `employees.txt`:

```
S Black 32000.0
H Ford 10.0 40.0
H Goodman 15.0 45.0
S Howell 44000.0
S Powell 50000.0
H Smith 22.5 40
```

5. Create an array of `HourlyEmployees` and `SalariedEmployees`. Read the data from the data file into the appropriate array and print out each person's monthly gross income.

Let's Play

6. Results



A screenshot of a Windows command prompt window. The title bar reads "C:\Windows\system32\cmd.exe". The window contains the following text:

```
Black    2666.67
Howell   3666.67
Powell   4166.67
  Ford    400.00
Goodman  675.00
Smith    900.00
Press any key to continue . . .
```