Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

Programming in Modern C++

Module M24: Inheritance: Part 4: Phone Hierarchy

NPTEL

Partha Pratim Das

Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

ppd@cse.iitkgp.ac.in

All url's in this module have been accessed in September, 2021 and found to be functional

Module Recap

Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

- Understood the need and use of `protected` access specifier
- Discussed the Construction and Destruction process of class hierarchy and related Object Lifetime

NPTEL

Module Objectives

Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

- Model a hierarchy of phones using inheritance

NPTEL

Module Outline

Module M24

Partha Pratim Das

Objectives & Outlines

ISA Hierarchy Design by Inheritance

Helper Classes

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

Landline Phone

Mobile Phone

Smart Phone

Refactoring

Hierarchy Integration

Extended Hierarchy of Phones

Module Summary

1. Objectives & Outlines
2. ISA Hierarchy Design by Inheritance
3. Helper Classes
4. Hierarchy of Phones by Interfaces
5. Interfaces & State Variables of Phones
   - Landline Phone
   - Mobile Phone
   - Smart Phone
6. Refactoring
7. Hierarchy Integration
   - Extended Hierarchy of Phones
8. Module Summary

NPTEL

Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

NPTEL

ISA Hierarchy Design by Inheritance

Approach to Modeling Hierarchy

Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

- Identify the **Concepts and their ISA relationships** to define the hierarchy: model with public inheritance
- Identify and model **Helper classes** - lower level UDTs to define components
- Identify the **Interface** of each concept: signatures of public member functions
- Identify the **State Variables** of each concept: types of of private / protected data members (also member functions used for ease of implementation)
- **Refactor** common data members and member functions between specialized and generalized classes to link the classes by inheritance
- **Integrate the hierarchy** with abstract (pure) interface
- Explore **extendability**
- *We illustrate with the phone hierarchy*

NPTEL

Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

NPTEL

Helper Classes

Helper Classes

Module M24

Partha Pratim Das

Objectives & Outlines

ISA Hierarchy Design by Inheritance

Helper Classes

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

Landline Phone

Mobile Phone

Smart Phone

Refactoring

Hierarchy Integration

Extended Hierarchy of Phones

Module Summary

Class	Description
`class PhoneNumber`	12-digit phone number
`class Name`	Subscriber Name (as `string`)
`class Photo`	Image & Subscriber Name as alt text
`class RingTone`	Audio & ring tone name
`class Contact`	`PhoneNumber`, `Name`, and `Photo` (optional) of a contact
`class AddressBook`	List of contacts

NPTEL

Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

NPTEL

Hierarchy of Phones by Interfaces

Hierarchy of Phones

Module M24

Partha Pratim Das

Objectives & Outlines

ISA Hierarchy Design by Inheritance

Helper Classes

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

Landline Phone

Mobile Phone

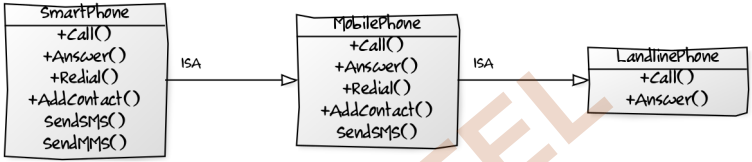
Smart Phone

Refactoring

Hierarchy Integration

Extended Hierarchy of Phones

Module Summary



- `MobilePhone` **ISA** `LandlinePhone`
  - `LandlinePhone` is *generalization*
  - `MobilePhone` is *specialization*
  - `MobilePhone` inherits the properties of `LandlinePhone`
- `SmartPhone` **ISA** `MobilePhone`
  - `MobilePhone` is *generalization*
  - `SmartPhone` is *specialization*
  - `SmartPhone` inherits the properties of `MobilePhone`
- **ISA** is *transitive*

NPTEL

Interfaces of Phones

Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

- **Landline Phone**
  - Call: By dial / keyboard
  - Answer
  - Caller ID (with special attached device)
- **Mobile Phone**
  - Call: By keyboard – shows number
    - By Number
    - By Name
  - Answer
  - Caller ID
  - Redial
  - Set Ring Tone
  - Add Contact
    - Number
    - Name
- **Smart Phone**
  - Call: By touchscreen – shows number & photo
    - By Number
    - By Name
  - Answer
  - Caller ID
  - Redial
  - Set Ring Tone
  - Add Contact
    - Number
    - Name
    - Photo

NPTEL

- There exists a substantial overlap between the functionalities of the phones
- A mobile phone is more capable than a land line phone and can perform (almost) all its functions
- A smart phone is more capable than a mobile phone and can perform (almost) all its functions
- **These phones belong to a Specialization / Generalization Hierarchy**

Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

NPTEL

Interfaces & State Variables of Phones

Interface & State Variable: Landline Phone

Module M24

Partha Pratim Das

Objectives & Outlines

ISA Hierarchy Design by Inheritance

Helper Classes

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

Landline Phone

Mobile Phone

Smart Phone

Refactoring

Hierarchy Integration

Extended Hierarchy of Phones

Module Summary

- **Landline Phone**
  - Call: By dial / keyboard
  - Answer

```
class LandlinePhone {
    PhoneNumber number_;
    Name subscriber_;
    RingTone rTone_;

public:
    LandlinePhone(const char *num, const char *subs);

    void Call(const PhoneNumber *p);

    void Answer();

    friend ostream& operator<<(ostream& os, const LandlinePhone& p);
};
```

NPTEL

Interface & State Variable: Mobile Phone

Module M24

Partha Pratim Das

Objectives & Outlines

ISA Hierarchy Design by Inheritance

Helper Classes

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

Landline Phone

Mobile Phone

Smart Phone

Refactoring

Hierarchy Integration

Extended Hierarchy of Phones

Module Summary

- **Mobile Phone**
  - Call: By keyboard – shows number
    - By Number
    - By Name
  - Answer
  - Redial
  - Set Ring Tone
  - Add Contact
    - Number
    - Name

```
class MobilePhone {
    PhoneNumber number_;
    Name subscriber_;
    RingTone rTone_;
    AddressBook aBook_;
    PhoneNumber *lastDial_;
    void SetLastDialed(const PhoneNumber& p);
    void ShowNumber();

public:
    MobilePhone(const char *num, const char *subs);

    void Call(PhoneNumber *p);
    void Call(const Name& n);

    void Answer();

    void ReDial();
    void SetRingTone(RingTone::RINGTONE r);
    void AddContact(const char *num = 0,
        const char *subs = 0);

    friend ostream& operator<<(ostream& os, const MobilePhone& p);
};
```

NPTEL

Interface & State Variable: Smartphone

- **Smart Phone**
  - Call: By touchscreen – shows number & photo
    - By Number
    - By Name
  - Answer
  - Redial
  - Set Ring Tone
  - Add Contact
    - Number
    - Name
    - Photo

```
class SmartPhone {
    PhoneNumber number_;
    Name subscriber_;
    RingTone rTone_;
    AddressBook aBook_;
    PhoneNumber *lastDial_;
    void SetLastDialed(const PhoneNumber& p);
    void ShowNumber();
    unsigned int size_;
    void DisplayPhoto();

public:
    SmartPhone(const char *num, const char *subs);

    void Call(PhoneNumber *p);
    void Call(const Name& n);

    void Answer();

    void ReDial();
    void SetRingTone(RingTone::RINGTONE r);
    void AddContact(const char *num = 0,
        const char *subs = 0);

    friend ostream& operator<<(ostream& os, const MobilePhone& p);
};
```

NPTEL

Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

NPTEL

Refactoring

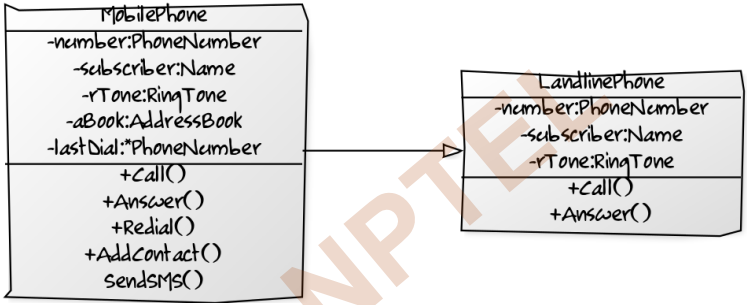
MobilePhone **ISA** LandlinePhone

```
class LandlinePhone { protected:
    PhoneNumber number_;
    Name subscriber_;
    RingTone rTone_;



public:
    LandlinePhone(const char *num,
        const char *subs) :
        number_(num), subscriber_(subs),
        rTone_() { }
    void Call(const PhoneNumber *p);

    void Answer();



    friend ostream& operator<<(ostream& os,
        const LandlinePhone& p);
};
```

```
class MobilePhone : public LandlinePhone { protected:
    //PhoneNumber number_;
    //Name subscriber_;
    //RingTone rTone_;
    AddressBook aBook_;
    PhoneNumber *lastDial_;
    void SetLastDialed(const PhoneNumber& p);
    void ShowNumber();
public:
    MobilePhone(const char *num,
        const char *subs) :
        LandlinePhone(num, subs), // Base ctor
        lastDial_(0) { }
    void Call(const PhoneNumber *p); // Override
    void Call(const Name& n);        // Overload
    //void Answer();                 // Inherited
    void ReDial();
    void SetRingTone(RingTone::RINGTONE r);
    void AddContact(const char *num = 0,
        const char *subs = 0);
    friend ostream& operator<< (ostream& os,
        const MobilePhone& p);
};
```

NPTEL

MobilePhone **ISA** LandlinePhone



NPTEL

SmartPhone **ISA** MobilePhone

Module M24

Partha Pratim Das

Objectives & Outlines

ISA Hierarchy Design by Inheritance

Helper Classes

Hierarchy of Phones by Interfaces

Interfaces & State Variables of Phones

Landline Phone

Mobile Phone

Smart Phone

Refactoring

Hierarchy Integration

Extended Hierarchy of Phones

Module Summary


```
class MobilePhone : public LandlinePhone { protected:
    //PhoneNumber number_;
    //Name subscriber_;
    //RingTone rTone_;
    AddressBook aBook_;
    PhoneNumber *lastDial_;
    void SetLastDialed(const PhoneNumber& p);
    void ShowNumber();


public:
    MobilePhone(const char *num,
        const char *subs) :
        LandlinePhone(num, subs), // Base ctor
        lastDial_(0) { }
    void Call(const PhoneNumber *p); // Override
    void Call(const Name& n);        // Overload
    //void Answer();                 // Inherited
    void ReDial();
    void SetRingTone(RingTone::RINGTONE r);
    void AddContact(const char *num = 0,
        const char *subs = 0);
    friend ostream& operator<< (ostream& os,
        const MobilePhone& p);
};
```

```
class SmartPhone : public MobilePhone { protected:
    //PhoneNumber number_;
    //Name subscriber_;
    //RingTone rTone_;
    //AddressBook aBook_;
    //PhoneNumber *lastDial_;
    //void SetLastDialed(const PhoneNumber& p);
    //void ShowNumber();
    unsigned int size_;
    void DisplayPhoto()
public:
    SmartPhone(const char *num,
        const char *subs) :
        MobilePhone(num, subs), // Base ctor
        lastDial_(0) { }
    void Call(const PhoneNumber *p); // Override
    void Call(const Name& n);        // Override
    //void Answer();
    void ReDial();                   // Override
    //void SetRingTone(RingTone::RINGTONE r);
    //void AddContact(const char *num = 0,
        //const char *subs = 0);
    friend ostream& operator<< (ostream& os,
        const SmartPhone& p);
};
```

NPTEL

Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

NPTEL

Hierarchy Integration

Hierarchy Integration

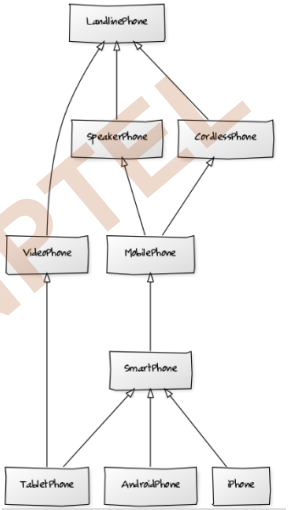
Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

```
// Abstract Base Class - A Pure Interface
class Phone { public:
    virtual void Call(const PhoneNumber *p) = 0;
    virtual void Answer() = 0;
    virtual void ReDial() = 0;
};
class LandlinePhone: public Phone {
protected:
    PhoneNumber number_;
    Name subscriber_;
    RingTone rTone_;
public:
    LandlinePhone(const char *num,
        const char *subs) :
        number_(num), subscriber_(subs),
        rTone_() { }
    // Implementations for interfaces
    void Call(const PhoneNumber *p);
    void Answer();
    // Dummy implementation not for use
    void ReDial()
    { cout << "Not implemented" << endl; }
    friend ostream& operator<<(ostream& os,
        const LandlinePhone& p);
};
```

```
class MobilePhone : public LandlinePhone { protected:
    AddressBook aBook_;
    PhoneNumber *lastDial_;
    void SetLastDialed(const PhoneNumber& p);
    void ShowNumber();
public:
    MobilePhone(const char *num, const char *subs) :
        LandlinePhone(num, subs), lastDial_(0) { }
    void Call(const PhoneNumber *p); // Override
    void Call(const Name& n);        // Overload
    void ReDial();                   // Override
    friend ostream& operator<< (ostream& os,
        const MobilePhone& p);
};
class SmartPhone : public MobilePhone {
protected: unsigned int size_;
    void DisplayPhoto();
public:
    SmartPhone(const char *num, const char *subs) :
        MobilePhone(num, subs), lastDial_(0) { }
    void Call(const PhoneNumber *p); // Override
    void Call(const Name& n);        // Override
    void ReDial();                   // Override
    friend ostream& operator<< (ostream& os,
        const SmartPhone& p);
};
```

NPTEL

Extended Hierarchy of Phones



NPTEL

Module Summary

Module M24
Partha Pratim Das
Objectives & Outlines
ISA Hierarchy Design by Inheritance
Helper Classes
Hierarchy of Phones by Interfaces
Interfaces & State Variables of Phones
Landline Phone
Mobile Phone
Smart Phone
Refactoring
Hierarchy Integration
Extended Hierarchy of Phones
Module Summary

- Using the Phone Hierarchy as an example analyzed the design process with inheritance

NPTEL