CSC Proprietary and Confidential

**OBJECT ORIENTED PROGRAMING - INHERITANCE Đào Anh Vũ**

**Content**

• Introduction – Structured programming – Concept of OOP – Classes & Objects – Features of OOP

✓Abstraction & Encapsulation ✓Inheritance

o Multiple inheritance o Abstract class o Interface ✓Polymorphism

CSC Proprietary and Confidential 2 December 2, 2015

**Structured Programming**

• Program = Data structure + Algorithm

output program Input data

Structured Programming:

• Low – level

• Focus on actions (action oriented)

• Less abstract

CSC Proprietary and Confidential 3 December 2, 2015

**CONCEPTS OF OOP**

• Motivation: – Interactions in the real world are object-object interactions. – We recognize everything as objects.

CSC Proprietary and Confidential 4 December 2, 2015

**CLASSES & OBJECTS**

• Class – Is blueprint from which objects are created. – Define data and action of objects.

• Object – Consist of data and actions. – Objects are instances of classes. – In most cases we interact with object through its methods.

CSC Proprietary and Confidential 5 December 2, 2015

**Examples**

CSC Proprietary and Confidential 6 December 2, 2015

**C++ Java**

**CLASSES & OBJECTS**

• Access modifier: – This is the way to specify the accessibility of a class and its members with

respective to other classes. – Access modifiers support for OOP features – Used at 2 levels:

✓Top – level for Class & Interface. ✓Member – level

CSC Proprietary and Confidential 7 December 2, 2015

**CLASSES & OBJECTS**

• Top – level for Class & Interface: – Public – Package/Default modifiers

**Access modifier**

**Scope**

Public Inside and outside the package Package/default Just inside the package

CSC Proprietary and Confidential 8 December 2, 2015

**CLASSES & OBJECTS**

• Object member - Level

**Access public protected private**

Same class (base)

Yes Yes Yes

Derived class Yes Yes No

Outside classes Yes No No

CSC Proprietary and Confidential 9 December 2, 2015

**Abstraction**

• Focus on the meaning. – Suppress irrelevant “implementation” details.

• See objects from outside of them.

CSC Proprietary and Confidential 10 December 2, 2015

**Encapsulation**

• Just methods of an object can access its own data.

• This is used to enforce the principle of data hiding.

• Handle with the visibility of object’s members.

• Implement “Abstraction”.

CSC Proprietary and Confidential 11 December 2, 2015

**Encapsulation**

Why we make it’s harder for our program to access its data?

CSC Proprietary and Confidential 12 December 2, 2015

**Encapsulation vs Abstraction**

CSC Proprietary and Confidential 13 December 2, 2015

**INHERITANCE**

There are many cases that an object acquires some/all properties/methods of another object.

**classB**

**classA**

**classB**

**C**

**classA**

CSC Proprietary and Confidential 14 December 2, 2015

**INHERITANCE**

Define a new class base on existing classes.

– Existing class: base class – New class: derived class

CSC Proprietary and Confidential 15 December 2, 2015

**INHERITANCE**

CSC Proprietary and Confidential 16 December 2, 2015

**INHERITANCE**

The final keyword in Java

– Sometime we don’t want a specific class to be a super class. – Authors control the use of their code.

CSC Proprietary and Confidential 17 December 2, 2015

**ACCESS SCOPE**

Access public protected private Same class (base) Yes Yes Yes Derived class Yes Yes No Outside classes Yes No No

CSC Proprietary and Confidential 18 December 2, 2015

**METHOD OVERLOADING**

**same name but not the parameter list, type of parameters or return type.**

How about overriding???

CSC Proprietary and Confidential 19 December 2, 2015

**MULTIPLE INHERITANCE**

• A class can inherit from more than one class.

• Some languages those support multiple inheritance: – C++ – Common Lisp – Perl

• Java does not support multiple inheritance, but we can overcome this by using interface.

CSC Proprietary and Confidential 20 December 2, 2015

**DIAMOND PROBLEM**

• Pre-Java 8 does not allow multiple inheritance → we will not face this problem.

• For C++, we solve this problem by virtual inheritance.

CSC Proprietary and Confidential 21 December 2, 2015

**Abstract Classes**

• May or may not include abstract methods.

• Cannot be instantiated

CSC Proprietary and Confidential 22 December 2, 2015

**Interfaces**

• A reference type in Java.

• Interfaces are not classes.

• Can contain only constant and method signatures.

• Cannot be instantiated.

CSC Proprietary and Confidential 23 December 2, 2015

**Polymorphism**

• Object can be represented in many forms.

• In each form, it take a specified set of action.

Client

Object

I

CSC Proprietary and Confidential 24 December 2, 2015

n

t

e

r

f

a

c

e

1

I

n

t

e

r

f

a

c

e

2

**Polymorphism**

CSC Proprietary and Confidential 25 December 2, 2015

**Exercises**

CSC Proprietary and Confidential 26 December 2, 2015

**References**

• Stephen Prata, C++ Primer Plus, 4thEdition, Sams, 2004.

• Bjarne Stroustrup, The C++ Programming Language, 4thEdition, Addison-Wesley, 2013

• Marshall Cline, C++ FAQ Lite, 2ndEdition, Addison-Wesley, 2000

• Robert L. Kruse, Alexander J. RybaData, Data Structures and Program Design in C++, Prentice-Hall, Inc., 2000

• Bruce Eckel, Thinking in C++,Prentice Hall, 1998

• http://faculty.orangecoastcollege.edu/sgilbert/book/03-4-ObjectOrientedConcepts- B/index.html

CSC Proprietary and Confidential 27 December 2, 2015