Object Oriented Programming using C++

Aggregation vs. Inheritance

LECTURE-20
Using Inheritance or Aggregation

In general, the difference between inheritance and aggregation is the difference between the is-a relationship and the has-a relationship. For example,

an apple is fruit; thus, you would use inheritance to model the relationship between the classes Apple and Fruit.

A person has a name; thus, you would use aggregation to model the relationship between the classes Person and Name.
Using Inheritance or Aggregation, cont.

Sometimes, the choice between inheritance and aggregation is not obvious. For example, you may use inheritance to model the relationship between the classes Circle and Cylinder. One could argue that a cylinder consists of circles; thus, you might use aggregation to define the Cylinder class as follows:
Using Inheritance or Composition, cont.

class Cylinder {
    private:
        Circle circle;
    /*** Constructors */
    /*** Methods */
};
Aggregation

• A tire has a circle as its boundary:

class Tire
{
    private
    String rating;
    Circle boundary;
}
Example

• Car is a Vehicle – Inheritance
• Car has a set of Tires – Aggregation

class Car : public Vehicle
{
    . . .
    private:
    Tire tires[4];
}
Inheritance representation using UML

Aggregation representation using UML
Aggregation

- Class has objects of other classes as members
- Weak *Has-a* relationship
- Part can exist without the whole/composite
- Class A aggregates class B if A contains an instance of type B

**Examples:** A room has-a table, An organization has-a customer
Composition: Objects as Members of Classes

- Class has objects of other classes as members
- **Strong** Has-a relationship
- Part cannot exist without the whole/composite

- **Examples:** A circle has a point, A college has computer science department etc.
Examples

• **Composition**
  A car *has-a* metal frame, *has-an* engine, and *has-a* transmission.
  A personal computer *has-a* CPU, a motherboard, and other components.

• **Aggregation**
  A student *has-a* book, *has-a* bag, *has-a* pen
  A journalist *has-a* camera etc.