Type Conversions

Lecture 12

Robb T. Koether

Hampden-Sydney College

Thu, Feb 12, 2009
Outline

1. Conversion to Created Types
2. Conversion to Built-in Types
Frequently in a program an object must be converted from one type to another.

For the built-in types, this is done automatically whenever it is sensible.

- Convert `float` to `int`.
- Convert `int` to `float`.

How can it be done with created types?
A class uses its constructors to define rules for converting an object of another type to an object of that type.
Example

Example (Convert int to Rational)

```cpp
// Rational constructor
Rational::Rational(int n) {
    num = n;
    den = 1;
    return;
}

// Usages
Rational r(100);
Rational r = 100;
r = (Rational)100;
r = Rational(100);
```
Example

How would you convert a `Point2D` to a `Vectr`?
Sometimes we want to convert an object of a created type to an object of a built-in type. For example, we might want to convert

- A Rational to a double.
- A Time to an int.

For this we need a conversion operator.
Conversion Operators

Conversion Operator Prototype

\[ \text{Class-name}::\text{operator} \ Other\text{-type}() \ \text{const}; \]

- The operator converts the created-type object to the built-in type and returns the object of the built-in type.
Example (Conversion Operators)

Rational::operator double() const;
Time::operator int() const;
Example (Convert Rational to double)

```cpp
Rational::operator double() const
{
    return (double)num/den;
}
```
Some Interesting Conversions

- Download and run ConversionTest.cpp