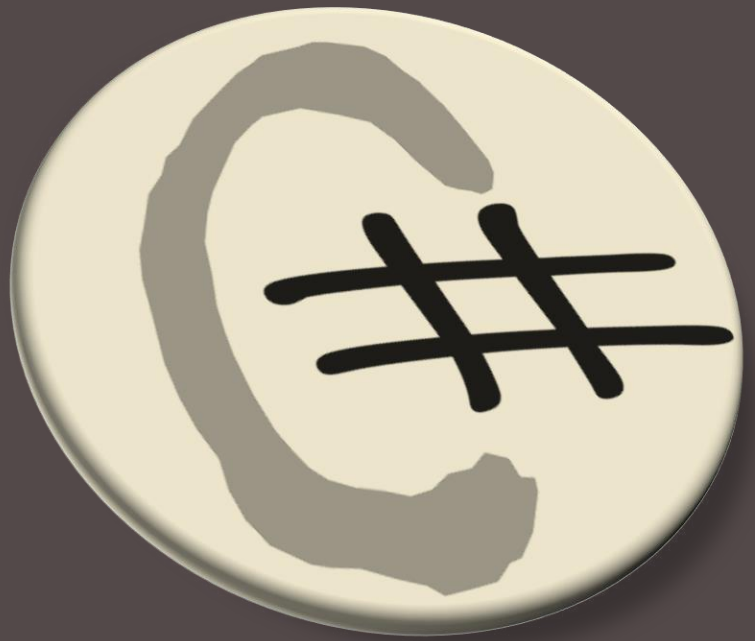
C#

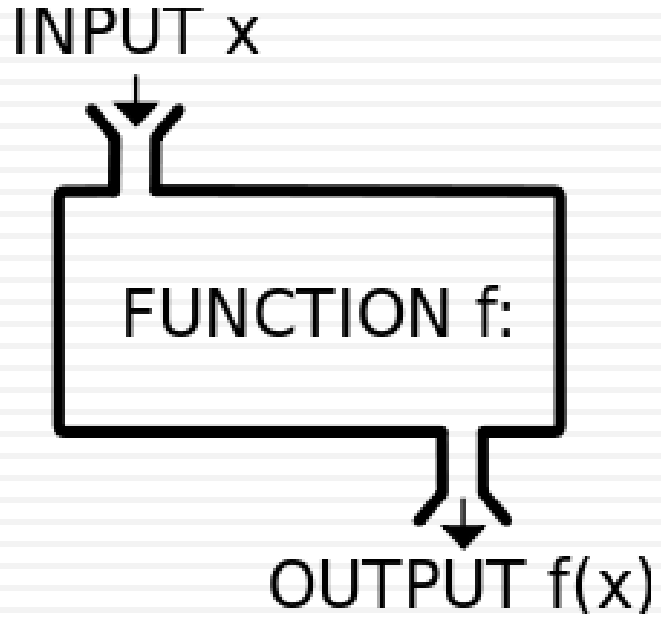
INTRODUCTION TO PROGRAMING



Part 3

Methods

- Function Name
- Function Parameters
- Function Return Type
- Function Body



Methods

- <Access Specifier> <Return Type> <Method Name>(Parameter List)

{

Method Body

}

Void Methods

```
static void PrintLogo()
{
   Console.WriteLine("Microsoft");
   Console.WriteLine("www.microsoft.com");
}
```

Return type Methods

```
static double GetRectangleArea(double width, double height)
{
   double area = width * height;
   return area;
}
```

```
using System;
namespace CalculatorApplication
{
    class NumberManipulator
    {
        public int FindMax(int num1, int num2)
        {
            /* local variable declaration */
            int result;

            if (num1 > num2)
                result = num1;
            else
                result = num2;
            return result;
        }
        static void Main(string[] args)
        {
            /* local variable definition */
            int a = 100;
            int b = 200;
            int ret;
            NumberManipulator n = new NumberManipulator();

            //calling the FindMax method
            ret = n.FindMax(a, b);
            Console.WriteLine("Max value is : {0}", ret );
            Console.ReadLine();
        }
    }
}
```

Recursion for factorial method

```
public static int fact(int number)
{
    int fact = 1;
    if (number == 0)
        return 1;

    else
    {
        for (int i = number; i >= 1; i--)
        {
            fact = fact * i;
        }
        return fact;
    }
```

Major Topics

- Object-oriented concepts
- Unified Modeling Language
- Use case and other UML diagrams
- Using UML

Object-Oriented Concepts

- Objects
- Classes

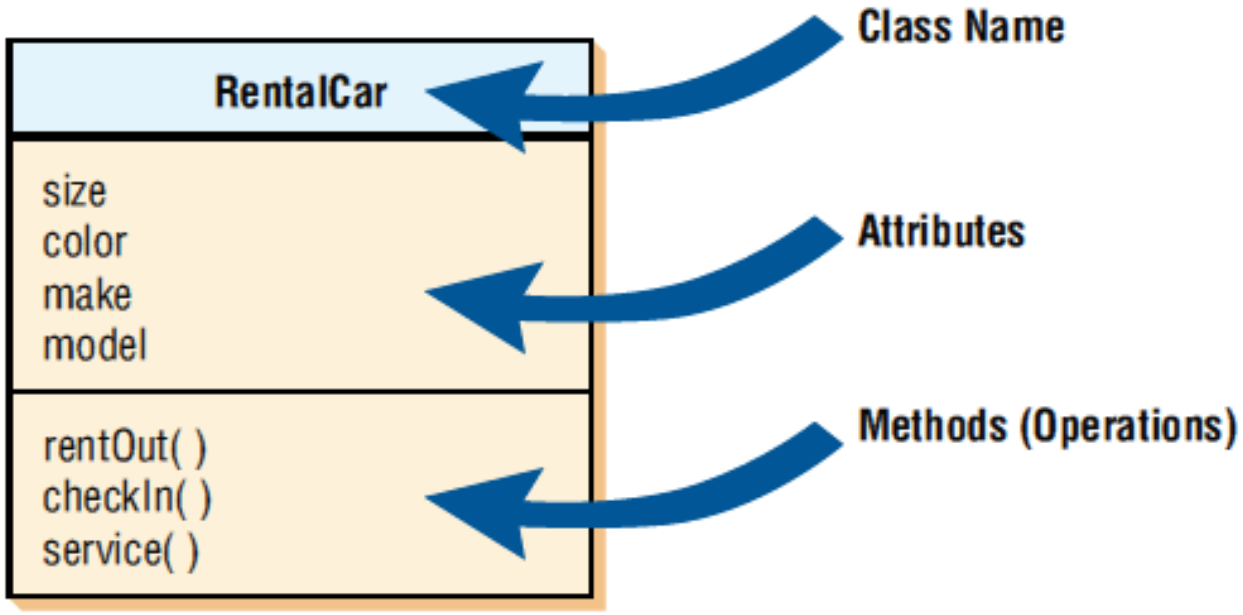
Classes

- Defines the set of shared attributes and behaviors found in each object in the class
- Should have a name that differentiates it from all other classes
- Instantiate is when an object is created from a class
- An attribute describes some property that is possessed by all objects of the class
- A method is an action that can be requested from any object of the class

Objects

- Persons, places, or things that are relevant to the system being analyzed
- May be customers, items, orders, and so on
- May be GUI displays or text areas on a display

An Example of a UML Class: A Class Is Depicted as a Rectangle Consisting of the Class Name, Attributes, and Methods



The Unified Modeling Language (UML) Concepts and Diagrams

- Things
- Relationships
- Diagrams

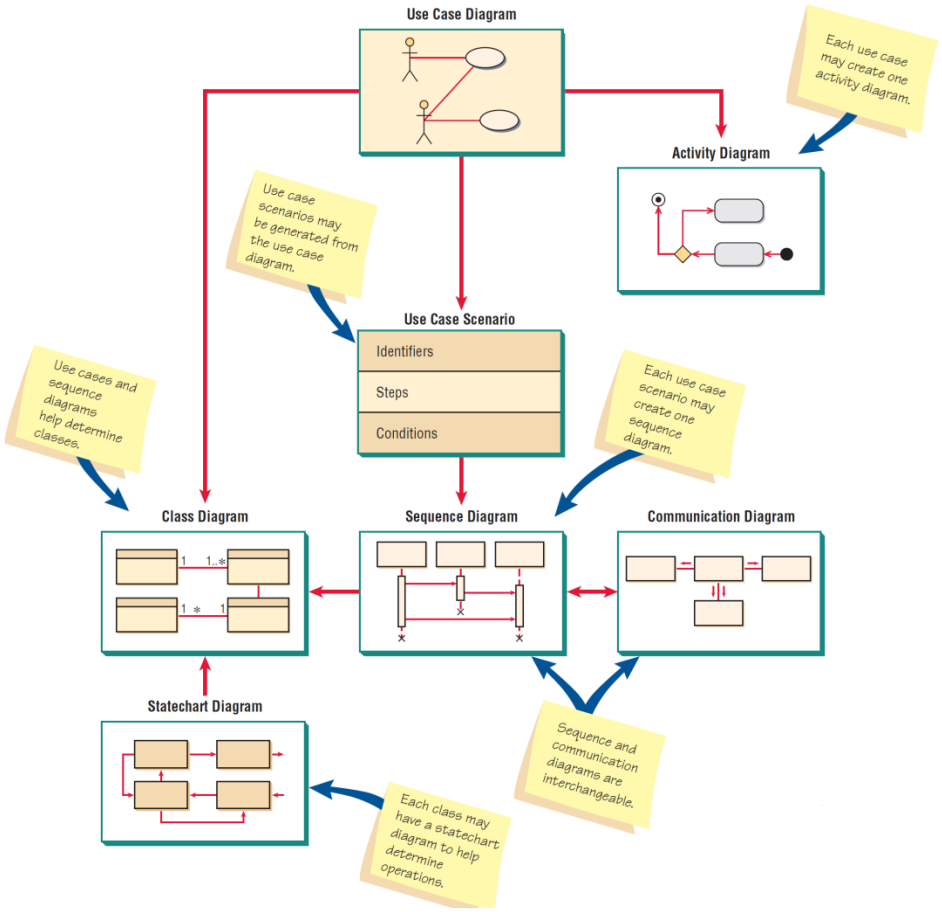
Commonly Used UML Diagrams

- Use case diagram
  - Describing how the system is used
  - The starting point for UML modeling
- Use case scenario
  - A verbal articulation of exceptions to the main behavior described by the primary use case
- Activity diagram
  - Illustrates the overall flow of activities

Commonly Used UML Diagrams (continued)

- Sequence diagrams
  - Show the sequence of activities and class relationships
- Class diagrams
  - Show classes and relationships
- Statechart diagrams
  - Show the state transitions

An Overview of UML Diagrams Showing How Each Diagram Leads to the Development of Other UML Diagrams



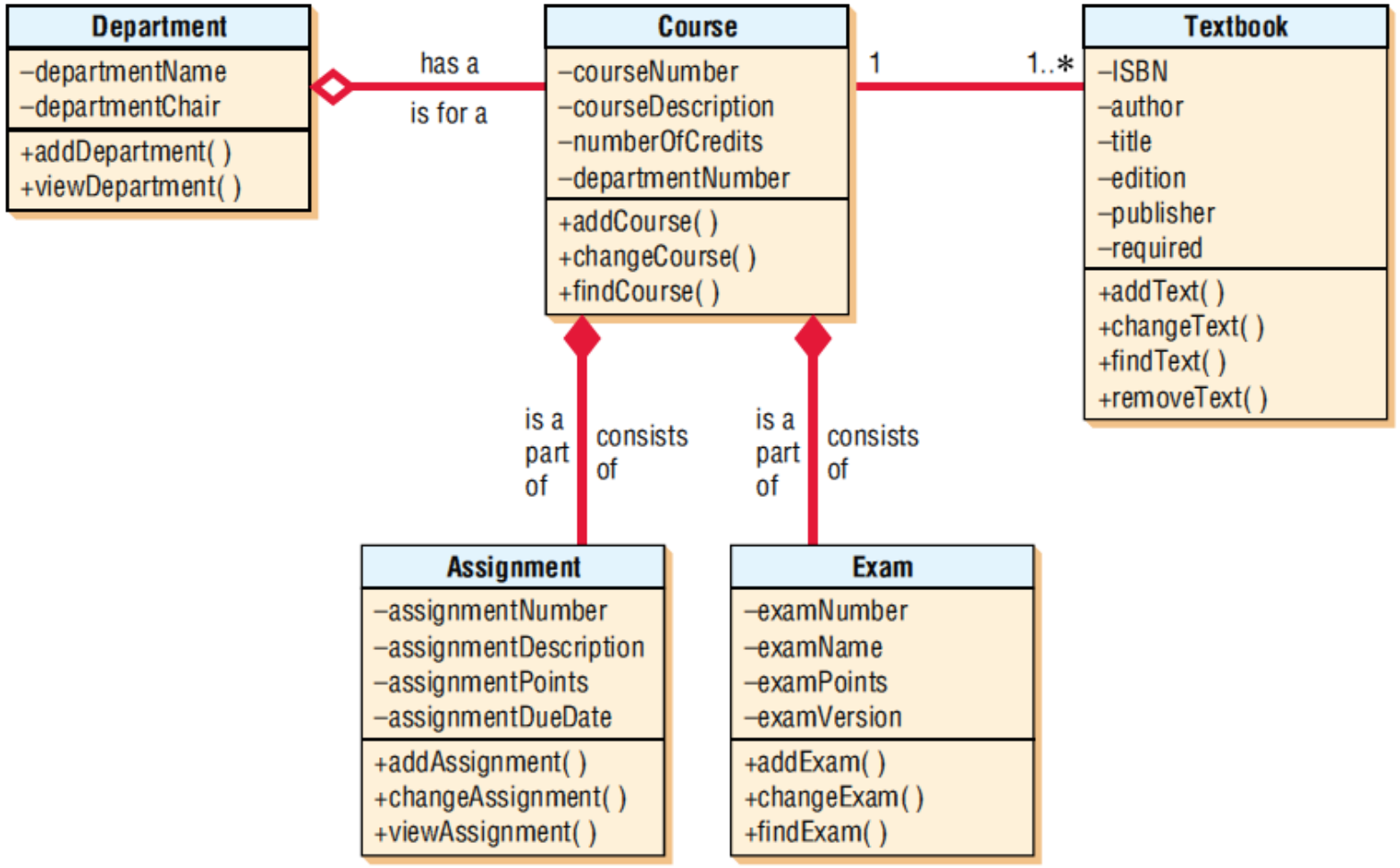
Class Diagrams

- Show the static features of the system and do not represent any particular processing
- Show the nature of the relationships between classes
- Show data storage requirements as well as processing requirements

Class Diagrams (continued)

- Classes
- Attributes
  - Private
  - Public
  - Protected
- Methods
  - Standard
  - Custom

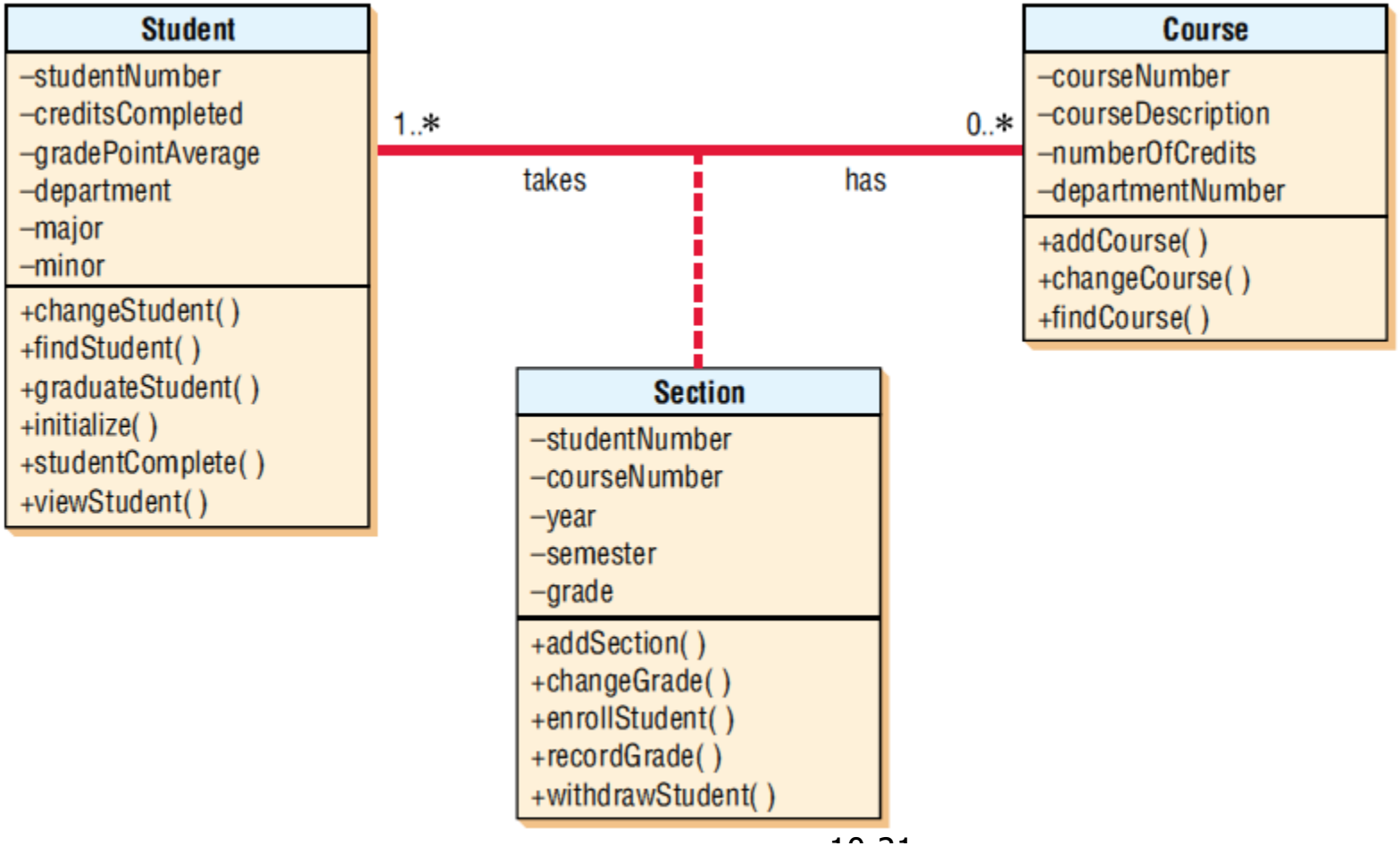
A Class Diagram for Course Offerings: The Filled-In Diamonds Show Aggregation and the Empty Diamond Shows a Whole-Part Relationship



Relationships

- The connections between classes
  - Associations
  - Whole/part

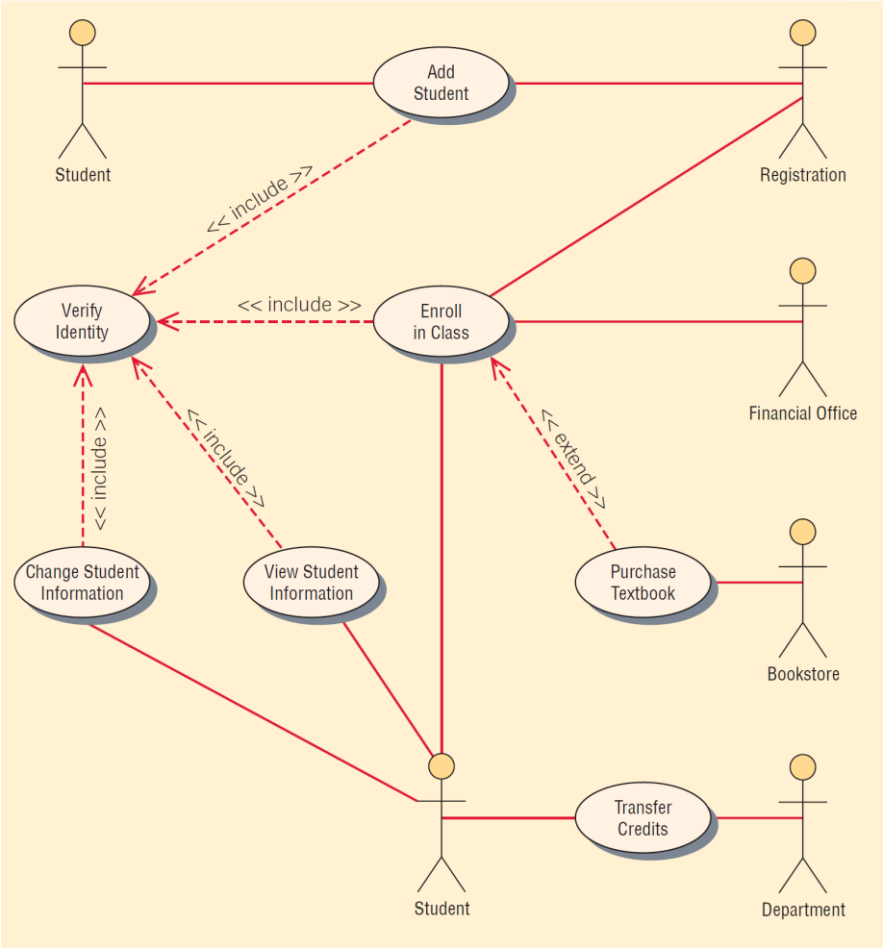
An Example of an Associative Class in Which a Particular Section Defines the Relationship between a Student and a Course



Use Case Modeling

- Describes what the system does, without describing how the system does it
- Based on the interactions and relationships of individual use cases
- Use case describes
  - Actor
  - Event
  - Use case

A Use Case Example of Student Enrollment (Figure 10.6)




Thank You!