

Object-Oriented Programming

Object-Oriented Programming

An ***algorithm*** is a step-by-step process.

A ***computer program*** is a step-by-step set of instructions for a computer.

Every computer program is an algorithm.

Algorithms have a long history in science, technology, engineering and math.

Object-Oriented Programming

Early computers were far less complex than computers are today.

Their memories were smaller and their programs were much simpler.



Object-Oriented Programming

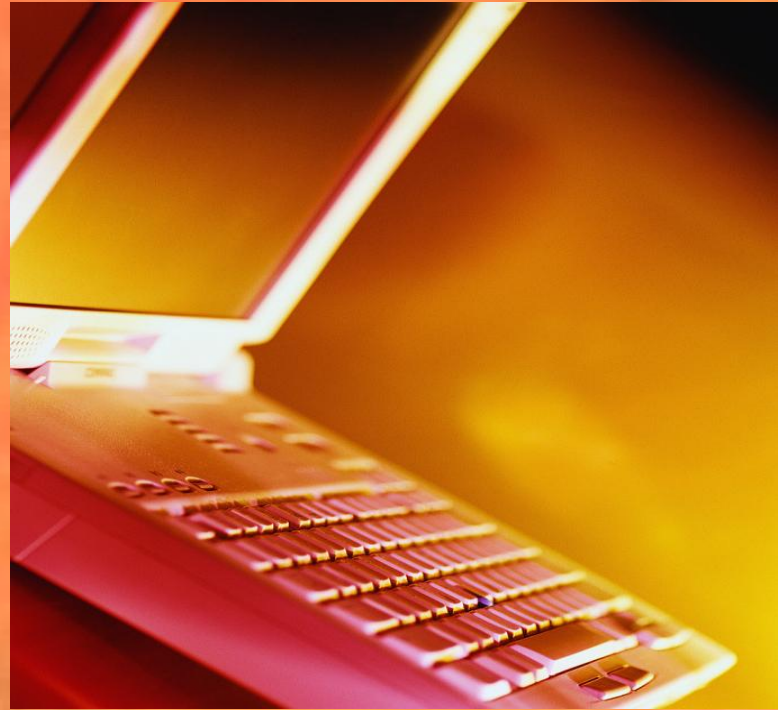
They usually executed only one program at a time.



Object-Oriented Programming

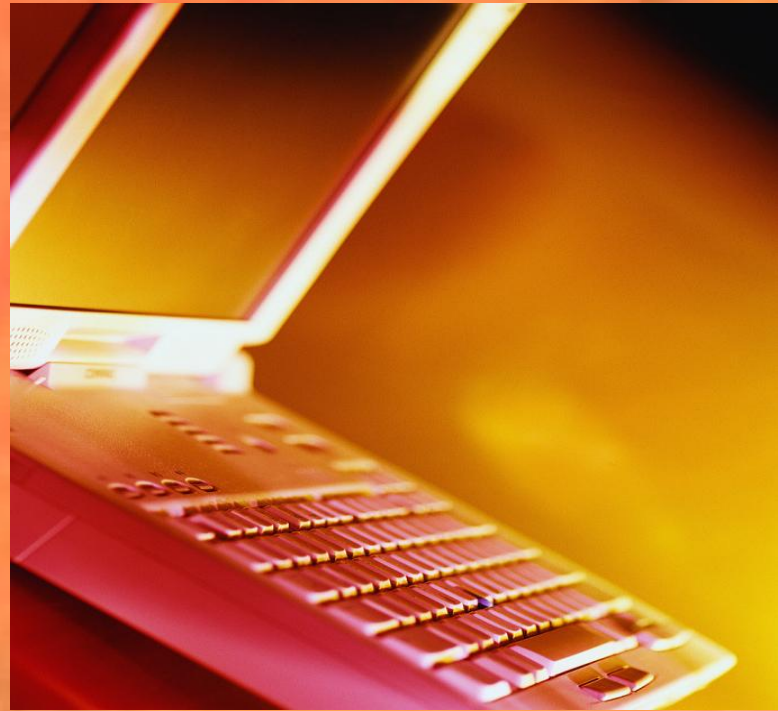
Modern computers are smaller, but far more complex than early computers.

The can execute many programs at the same time.



Object-Oriented Programming

Computer scientists have introduced the notion of **objects** and **object-oriented programming** to help manage the growing complexity of modern computers.



Object-Oriented Programming

An **object** is anything that can be represented by data in a computer's memory and manipulated by a computer program.

Object-Oriented Programming

An **object** is anything that can be represented by data in a computer's memory and manipulated by a computer program.

Numbers



Object-Oriented Programming

An **object** is anything that can be represented by data in a computer's memory and manipulated by a computer program.

Text



Object-Oriented Programming

An **object** is anything that can be represented by data in a computer's memory and manipulated by a computer program.

Pictures



Object-Oriented Programming

An **object** is anything that can be represented by data in a computer's memory and manipulated by a computer program.

Sound



Object-Oriented Programming

An **object** is anything that can be represented by data in a computer's memory and manipulated by a computer program.

Video



Object-Oriented Programming

An object is anything that can be represented by data.



Object-Oriented Programming

An object can be something in the physical world or even just an abstract idea.

An airplane, for example, is a physical object that can be manipulated by a computer.



Object-Oriented Programming

An object can be something in the physical world or even just an abstract idea.

A bank transaction is an example of an object that is not physical.



The image shows a close-up of a bank statement. At the top, it says "11/16" and "Monthly Service". Below that, there is a section titled "Daily Balance" with a checkbox. The table below lists dates and corresponding amounts.

Date	Amount
10/20	\$ 738.97
10/21	526.82
10/22	590.53
10/23	524.21
10/26	362.24
10/27	308.42

Object-Oriented Programming

To a computer, an object is simply something that can be represented by data in the computer's memory and manipulated by computer programs.



Object-Oriented Programming

The data that represent the object are organized into a set of *properties*.

The values stored in an object's properties at any one time form the *state* of an object.

Name: PA 3794

Owner: US Airlines

Location: 39 52' 06" N 75 13' 52" W

Heading: 271°

Altitude: 19 m

AirSpeed: 0

Make: Boeing

Model: 737

Weight: 32,820 kg

Object-Oriented Programming

Computer programs implement algorithms that manipulate the data.

In object-oriented programming, the programs that manipulate the properties of an object are the object's **methods**.

```
class Bicycle {  
  
    int cadence = 0;  
    int speed = 0;  
    int gear = 1;  
  
    void changeCadence(int newValue) {  
        cadence = newValue;  
    }  
  
    void changeGear(int newValue) {  
        gear = newValue;  
    }  
  
    void speedUp(int increment) {  
        speed = speed + increment;  
    }  
  
    void applyBrakes(int decrement) {  
        speed = speed - decrement;  
    }  
}
```

Object-Oriented Programming

We can think of an object as a collection of properties and the methods that are used to manipulate those properties.

Properties

Methods

```
class Bicycle {  
    {  
        int cadence = 0;  
        int speed = 0;  
        int gear = 1;  
    }  
    {  
        void changeCadence(int newValue) {  
            cadence = newValue;  
        }  
        void changeGear(int newValue) {  
            gear = newValue;  
        }  
        void speedUp(int increment) {  
            speed = speed + increment;  
        }  
        void applyBrakes(int decrement) {  
            speed = speed - decrement;  
        }  
    }  
}
```

Object-Oriented Programming

A **class** is a group of objects with the same properties and the same methods.

```
class Bicycle {  
  
    int cadence = 0;  
    int speed = 0;  
    int gear = 1;  
  
    void changeCadence(int newValue) {  
        cadence = newValue;  
    }  
  
    void changeGear(int newValue) {  
        gear = newValue;  
    }  
  
    void speedUp(int increment) {  
        speed = speed + increment;  
    }  
  
    void applyBrakes(int decrement) {  
        speed = speed - decrement;  
    }  
}
```

Object-Oriented Programming

Each copy of an object from a particular class is called an *instance* of the object.



Object-Oriented Programming

The act of creating a new instance of an object is called **instantiation.**



Object-Oriented Programming

A class can be thought of as a blueprint for instances of an object.



Object-Oriented Programming

Two different instances of the same class will have the same properties, but different values stored in those properties.



Object-Oriented Programming

Object

Property

The same terminology is used in most object-oriented programming languages.

Method

Instantiation

Instance

State

Class