

High-level languages

Most programs are written in *high-level languages* that can be understood by both human and computers.

Such programs are then translated to binary codes (in a machine language).

An *interpreter* is a software system to perform such translation.

November 11, 2010

1

Interpreter



- Translates and executes a program one line at a time.
- Programs must be translated every time they are executed.
- Inefficient in general, but convenient!

November 11, 2010

2

High-level languages

Most programs are written in *high-level languages* that can be understood by both human and computers.

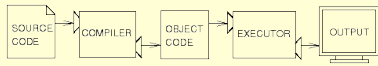
Such programs are then translated to binary codes (in a machine language).

A *compiler* provides another way to perform such translation.

November 11, 2010

3

Compiler



- Translates a program into binary/object code completely before execution.
- Program execution is done by running binary/object code.
- Very efficient but a bit harder to program.

November 11, 2010

4

Modern high-level languages

- Interpreted: JavaScript, Perl, Python, Ruby, Scheme, ...
- Compiled: C/C++, Fortran, Java, Pascal, ...

November 11, 2010

5

Python and Java

In this course, you learn how to program in two languages.

- **Python:** A simple language that supports a user-friendly interpreter.
- **Java:** The industry standard. A bit harder to learn than Python but very powerful for developing large-scale software, involving many programmers.

November 11, 2010

6

Python

```
print 'Hello, World!'
```

November 11, 2010

7

Java

```
class HelloWorld {  
    static public void main(String args[]) {  
        System.out.print("Hello, World!");  
    }  
}
```

November 11, 2010

8

What is Java?

November 11, 2010

9

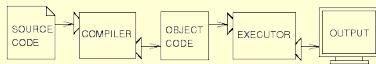
What is Java?

- The most modern, general-purpose, compiler-based language.
- Its root can be traced back to C.
- The first general-purpose language designed to work on the Internet (under the client/server paradigm).
- The first popular *object-oriented* language.
- Less efficient than C/C++, but far more versatile.

November 11, 2010

10

Java compiler



- Translates a program into object code that can be executed on any platform.
- Program execution is done by an executor called the Java Virtual Machine (JVM).
- This paradigm allows "Internet computing".

November 11, 2010

11

James A. Goslin (1955—)



November 11, 2010

<http://blogs.sun.com/jag/>

12

Classes and objects in Java

- In Java, an object must be defined as an instance of a class.
- Every Java class has a *constructor* that instantiates an object.
- In Java, attributes are called *fields*, and operations (or behaviors) are called *methods*.

November 11, 2010

13

Java class

```
public class Dog {
```

```
    Fields (attributes)
```

```
    Constructor
```

```
    Methods (operations or behaviors)
```

```
}
```

November 11, 2010

14

Accessibility

Classes, fields and methods must be declared with *access modifiers*.

- Classes and methods are usually declared as **public**—accessible by anyone from anywhere.
- Fields are usually declared as **private**—accessible only within the object.

November 11, 2010

15

Primitive data types

Fields are often defined using variables. In Java, variables must be declared with types. There are four primitive data types:

- `int`—integers: ..., -1, 0, 1, 2, ...
- `float`, `double`—reals: ..., 0.0, ...
- `char`—letters: 'a', 'b', 'c', ...
- `boolean`—boolean values: true, false

November 11, 2010

16

Defining fields

```
public class Dog {
```

```
    private String breed;  
    private char gender;  
    private float height;  
    private float weight;  
    private int age;
```

Constructor

Methods (operations or behaviors)

```
}
```

November 11, 2010

17
