

# 高级语言程序设计 (CE204107)

孙聪

网络与信息安全学院

2020-08-31

# 教材与参考书

## 教材

- Java实用程序设计，孙聪等，西安电子科技大学出版社，2015



- 随书代码: <https://github.com/suncongxd/java-book-code>
- 课程资料: <https://github.com/suncongxd/java-course>

## 参考书

- Thinking in Java (4th Edition), Bruce Eckel
- Introduction to Java Programming, D. Liang
- <https://docs.oracle.com/javase/8/docs/api/>

# 成绩评定

- 期末考试：60%
- 上机作业（3次）：30%
- 随堂测验及考勤：10%
  - 随堂测试当堂下发/收回

# 上机实验与上机作业

## 上机实验8学时，要求

- 运行并理解课件中的所有例子程序
- 完成上机作业

## 上机作业要求

- 提交到助教邮箱（伍亚飞：18291880308@163.com）
- 提交源代码和上机报告。源代码为.java文件，报告文档为word或pdf格式（请勿使用.wps格式）
- 为了方便助教复现你的程序，不要因为报告正文中包含了源代码就不提交源代码文件
- 按照规定的deadline提交，超过deadline扣除该次作业所有分数

# 课程内容

- Java概述
- 面向对象程序设计概念
- Java语言基础
- Java面向对象特性
- Java高级特征
- 容器类（在线）
- 常用预定义类（在线）
- 异常处理
- 输入输出
- 线程（在线）

# 课程内容

- Java概述
- 面向对象程序设计概念
- Java语言基础
- Java面向对象特性
- Java高级特征
- 容器类
- 常用预定义类
- 异常处理
- 输入输出
- 线程

# 提要

1 Java的发展历史与现状

2 什么是Java

3 Java的特征

# 提要

## 1 Java的发展历史与现状

## 2 什么是Java

## 3 Java的特征



# Java的发展历史



James Gosling   P. Naughton   John Gage   Bill Joy

# Java的发展历史



James Gosling   P. Naughton   John Gage   Bill Joy

- 1991, Sun公司成立Green小组, 研发面向家电的软件
  - C/C++ ... → 开发了Oak语言
- 1992.8, Star 7: Oak与Green OS、用户接口模块、硬件模块集成
  - Green小组 → First Person子公司
- 1994, First Person解体, Sun决定将Oak与WWW相结合

# Java的发展历史



James Gosling   P. Naughton   John Gage   Bill Joy

- 1995, Gosling和Naughton完成了
  - Oak新版 —— 后更名为Java
  - 基于Oak的浏览器原型Web Runner —— 后更名为HotJava
- 1995.5, Java的第一个发布版本
  - Java 1.0a2
  - HotJava浏览器

# Java的发展历史

## ● HotJava: 第一个支持Java的浏览器



# Java的发展历史

版本名称	发布时间
JDK 1.0	23rd Jan 1996
JDK 1.1	19th Feb 1997
J2SE 1.2	8th Dec 1998 (J2SE/J2EE/J2ME @1999)
J2SE 1.3	8th May 2000
J2SE 1.4	6th Feb 2002
J2SE 5.0	30th Sep 2004 (J2SE 1.5 → J2SE 5.0)
Java SE 6	11th Dec 2006 (Sun公司被Oracle公司收购@Apr 2009)
Java SE 7	28th Jul 2011
Java SE 8	18th Mar 2014
Java SE 9	21st Sep 2017
Java SE 10	20th Mar 2018
Java SE 11	25th Sep 2018
Java SE 12	19th Mar 2019

● [https://en.wikipedia.org/wiki/Java\\_version\\_history](https://en.wikipedia.org/wiki/Java_version_history)

# Java的发展现状

## 当前流行的高级语言

C, C++, Java, C#, Objective-C, Python, PHP, JavaScript, Ruby  
...

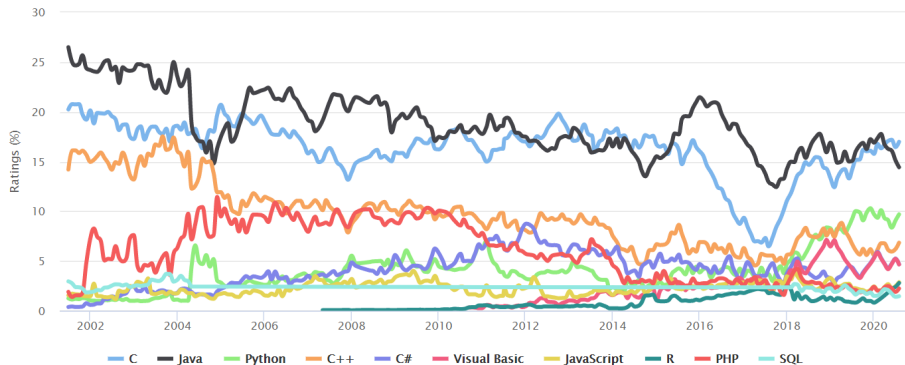
## TIOBE编程语言社区排行榜

- <https://www.tiobe.com/tiobe-index/>
- 基于互联网上有经验的程序员、课程和第三方厂商的数量，使用搜索引擎（Google, Bing, Yahoo!, Wikipedia, Amazon, YouTube, Baidu等）计算排名
- 用于衡量编程语言流行趋势和热门程度，开发新系统时的语言选择依据（不用于衡量现有代码的体量）

# TIOBE Index for August 2020

Aug 2020	Aug 2019	Change	Programming Language	Ratings	Change
1	2	▲	C	16.98%	+1.83%
2	1	▼	Java	14.43%	-1.60%
3	3		Python	9.69%	-0.33%
4	4		C++	6.84%	+0.78%
5	5		C#	4.68%	+0.83%
6	6		Visual Basic	4.66%	+0.97%
7	7		JavaScript	2.87%	+0.62%
8	20	▲	R	2.79%	+1.97%
9	8	▼	PHP	2.24%	+0.17%
10	10		SQL	1.46%	-0.17%
11	17	▲	Go	1.43%	+0.45%
12	18	▲	Swift	1.42%	+0.53%
13	19	▲	Perl	1.11%	+0.25%
14	15	▲	Assembly language	1.04%	-0.07%
15	11	▼	Ruby	1.03%	-0.28%
16	12	▼	MATLAB	0.86%	-0.41%
17	16	▼	Classic Visual Basic	0.82%	-0.20%
18	13	▼	Groovy	0.77%	-0.46%
19	9	▼	Objective-C	0.76%	-0.93%
20	28	▲	Rust	0.74%	+0.29%

# TIOBE Index for August 2020





# TIOBE Index for August 2020

Programming Language	2020	2015	2010	2005	2000	1995	1990	1985
Java	1	2	1	2	3	-	-	-
C	2	1	2	1	1	2	1	1
Python	3	7	6	6	22	20	-	-
C++	4	3	4	3	2	1	2	9
C#	5	5	5	9	9	-	-	-
JavaScript	6	8	8	10	6	-	-	-
PHP	7	6	3	5	21	-	-	-
SQL	8	-	-	-	-	-	-	-
Swift	9	16	-	-	-	-	-	-
R	10	12	54	-	-	-	-	-
Lisp	27	24	16	14	8	6	4	2
Fortran	31	25	23	15	16	4	3	5
Ada	33	29	24	17	14	5	6	3
Pascal	241	15	14	27	12	3	10	6

# 提要

1 Java的发展历史与现状

2 什么是Java

3 Java的特征

# 什么是Java

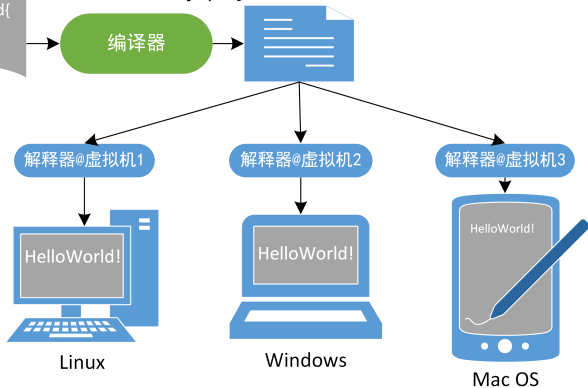
- Java是一种编程语言

- 一次编程，到处运行

源码: HelloWorld.java

```
public class HelloWorld{  
    ...  
}
```

字节码: HelloWorld.class



# 第一个Java应用程序

## HelloWorld.java

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

- 源文件扩展名必须是 .java
- 主类
  - 与源文件名称相同的类，通常定义为public class
  - 每个源文件中至多有一个public class
- 主类中的main() 方法声明
  - public static void main(String args[]){...}
  - 是Java程序入口点

# 第一个Java应用程序

## 安装JDK

- 下载: <https://www.oracle.com/technetwork/java/javase/downloads/index.html>
- 运行JDK的安装程序
- 改变环境变量
  - JAVA\_HOME = C:\Program Files\Java\jdk-12.0.2
  - PATH = [原PATH];%JAVA\_HOME%\bin

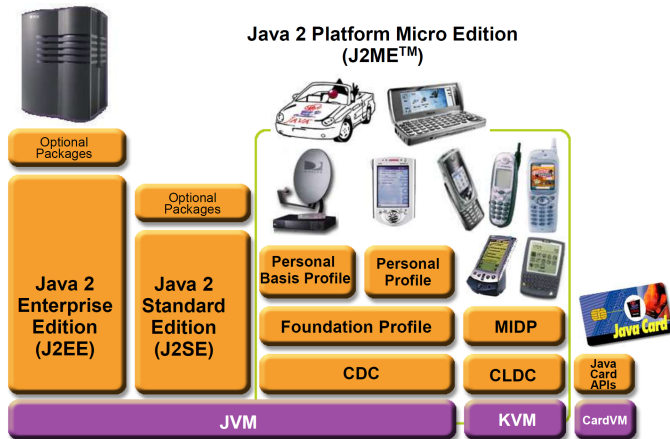
## 编译与执行

- HelloWorld.java  $\xrightarrow{\text{javac HelloWorld.java}}$  HelloWorld.class
- java HelloWorld

# 什么是Java

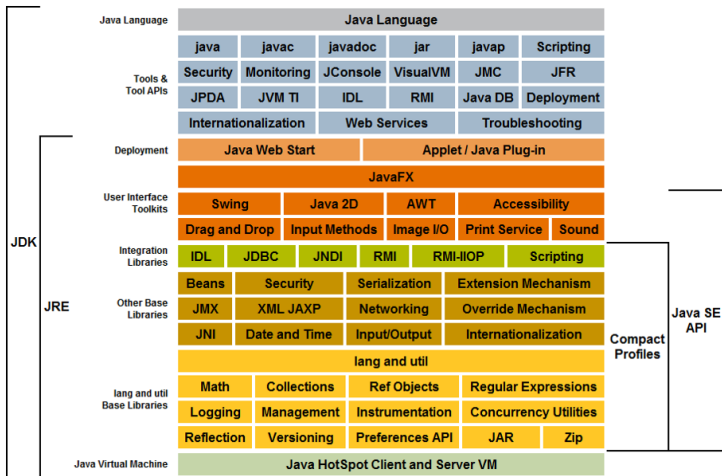
- Java是一种平台
  - 用于开发、部署和运行特定场景下的Java应用程序

## The Java™ Platform



# Java SE平台

- 针对运行于PC上的一般桌面应用程序
- Java SE平台 = Java运行时环境(JRE) + Tools & Tool APIs



# Java ME平台

- 针对嵌入式设备和移动设备
  - 微控制器和传感器
  - 电视置顶盒
  - 汽车导航系统
  - 打印机
  - ...
- 对Java SE进行语言精简，对运行环境进行优化
  - 配置（Configuration）：从内存，处理器，网络等方面对运行时环境的规约（常见：CLDC和CDC）
  - 轮廓（Profile）：扩充配置，形成可用的API集合，应用程序针对特定的轮廓编写



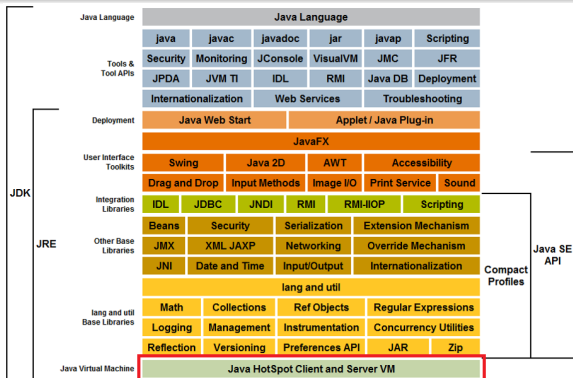
# Java EE平台

- 针对企业级和高端服务器的Internet应用开发
- 基于Java SE实现，包括
  - Enterprise JavaBeans (EJB)
  - Java Servlets API
  - Java Server Pages (JSP)
  - Java Message Services (JMS)
  - ...
- <http://www.oracle.com/technetwork/topics/newtojava/java-technology-concept-map-150250.pdf>
- 本课程此后提及的“Java”指Java SE

# Java虚拟机

## 定义

一种在真实计算机上通过软件仿真实现的虚构机器，用于执行Java字节码



● JRE = JVM + Java API + 其他组件

# Java虚拟机

## JVM的两个侧面

- 规范
- 实现

# Java虚拟机规范

- JVM规范定义了一组抽象的逻辑组件，包括
  - 指令集
  - .class文件的格式
  - 数据类型
  - 寄存器组：程序计数器、栈顶指针等
  - 类和接口的装载、链接及初始化方式
  - 栈（栈帧、局部变量）、堆（对象实例、数组）、方法区（运行时常量/符号表、字节码的方法代码）
- <http://docs.oracle.com/javase/specs/>
- 课本附录2“Java虚拟机体系结构”

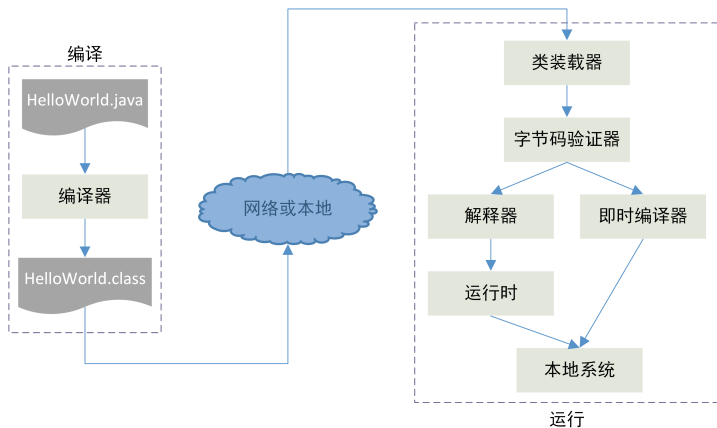
# Java虚拟机实现

- JVM需针对不同的操作系统和硬件平台做专门实现
- 各平台厂商对JVM有不同的实现(Oracle的HotSpot JVM, IBM的Jikes RVM)
- 实现要求: 必须符合JVM规范(使得所有运行系统能够执行统一格式的字节码)

## JVM实现的一般组成

- 类装载器
- 字节码验证器
- 解释器
- 即时编译器
- 运行时

# Java虚拟机实现



## Java字节码运行过程

- 代码的装入
- 代码的验证
- 代码的执行：解释方式、即时编译JIT方式

# 提要

1 Java的发展历史与现状

2 什么是Java

3 Java的特征

# Java的特征

## Sun公司Java白皮书

- Java语言是一种“简单，面向对象，分布式，解释型，健壮，安全，体系结构中立，可移植，高性能，多线程和动态”的编程语言
- <https://tech-insider.org/java/research/acrobat/9503.pdf>

## Oracle:

- 平台无关，高性能，易学，基于标准，适宜企业应用，丰富的工具生态系统



# Java的特征(1) - 简单性

- 语法和语义都比较单纯，基本语法沿承C和C++，容易学习和使用
- 忽略了C++的复杂特性
  - 指针
  - 多重继承
  - 运算符重载
  - goto
  - 内存管理（Java实现了自动内存回收机制）
- 提供功能丰富的可重用类库，简化了编程工作量

## Java的特征(2) - 面向对象

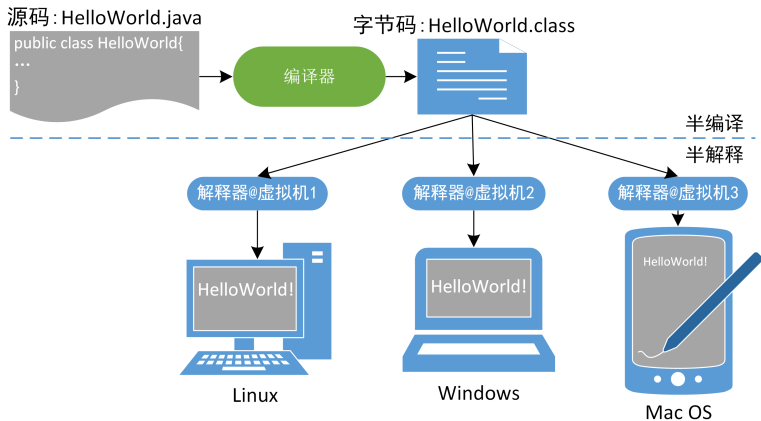
- Java是纯面向对象语言，更全面地支持面向对象方法
  - 通过封装、继承和多态来提供灵活性、模块化和可重用性
  - 封装性：Java对象有模块化性质和信息隐藏能力
  - 继承性：单继承，通过接口实现多重继承的效果
  - 多态性：在继承的基础上，通过重写、向上转型、动态绑定实现多态

## Java的特征(3) - 分布式

- 数据分布
  - 通过Java的URL类可以访问网上的各类信息资源，访问方式完全类似于对本地文件的访问
- 操作分布
  - Java程序（字节码）可以被传输并在互联网上的客户机上运行（例如网页中的Applet）
  - 将计算从服务器分布至客户机，避免网络拥挤，提高系统效率

## Java的特征(4) - 解释型

- 半编译半解释：兼具编译执行的效率优势和解释执行的灵活性



## Java的特征(5) - 健壮性

- 自动垃圾收集：内存管理对程序员透明
- 异常处理机制
  - 在编译时发现部分异常
  - 在运行时处理部分异常
- 强类型语言：编译时须经严格的类型检查

## Java的特征(6) - 安全性

- 内存分配模型
  - 内存布局由运行系统控制，对于程序员透明（C/C++：由编译器决定）
  - 取消了指针，编译器通过符号指针来引用内存，运行系统在运行时将符号指针解释为实际内存地址
- 安全控制模型
  - 类装载器
  - 字节码验证器：确保代码遵循下列规则
    - 不存在伪造的引用
    - 未违反访问权限
    - 严格遵循对象访问规范来访问对象
    - 用合适的参数调用方法
    - 没有栈溢出

# Java 的特征 (7) - 可移植性

- 可移植性源于两方面：

- 半编译、半解释
- 体系结构中立：采用标准的独立于硬件平台的中间语言

源码: HelloWorld.java

```
public class HelloWorld{  
    ...  
}
```

字节码: HelloWorld.class

编译器

体系结构中立：  
字节码规范独立于硬件平台

解释器@虚拟机1

解释器@虚拟机2

解释器@虚拟机3

HelloWorld!

Linux

HelloWorld!

Windows

HelloWorld!

Mac OS

## Java的特征(8) - 高性能

- 字节码与机器码十分接近，使得两者间转换快捷
- 即时编译技术
  - 将要执行的字节码一次编译为机器码，以提高应用程序运行速度（通常针对部分关键代码）
  - 保留可移植性的同时，提高了性能
- Java与C++/C#谁更快？
  - <http://keithlea.com/javabench/index>
  - <http://www.shudo.net/jit/perf/>



## Java的特征(9) - 多线程

- Java在语言级实现的对并发的支持机制
  - 线程执行：被映射到实际操作系统线程，映射过程对程序员透明
  - 线程调度：基于java.lang.Thread类的内建机制
  - 线程同步：采用C. A. R. Hoare的临界区保护规则

## Java的特征(10) - 动态特性

- Java只在运行时动态装载必需的类
  - 使Java可以动态维护应用程序与其支持类之间的一致性
  - C++: 支持类库变化 → 应用程序重新编译

# 常用集成开发环境

- Eclipse:

- <http://www.eclipse.org/downloads/>
- 开源的、基于Java的可扩展开发平台
- 支持多种编程/建模语言

- NetBeans: Apache

- <https://netbeans.org/>

- IntelliJ IDEA: JetBrains

- <https://www.jetbrains.com/idea/>

- JCreator: Xinox Software

- <http://www.jcreator.com/>

## 思考与练习

找出并修改下面代码的错误

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

## 课后练习

- 请在自己的PC上安装JDK和Eclipse
- 编写并运行HelloWorld程序