

# 讲课 知识点标记

## 走进java

### 关于虚拟机的一些知识

☑ java虚拟机和 virtualbox, android虚拟机等 不是一回事

☑ 微软也发布过虚拟机, java与.net 之间的恩恩怨怨

```
$ java -version
```

```
java version "1.8.0_144"
```

```
Java(TM) SE Runtime Environment (build 1.8.0_144-b01)
```

```
Java HotSpot(TM) 64-Bit Server VM (build 25.144-b01, mixed mode)
```

## 什么是内存

从最初的计算机模型开始讲起, 从二进制 形成汇编, 计算机与其子系统之间的速度的速度巨大差异

用办公室的位置 举例子

- 乱序执行
- 分支预测
- 多线程模型

专门的浮点数技术GPU, 计算卷积的CPU

硬件是非常重要的因素, 为什么 android 不如 iphone

## 我们为什么要使用多线程

等待磁盘io

gui

多核cpu的 出现  $p = fcv^2$

## 摩尔定律 \*\*Amdahl定律 \*\*

当价格不变时，集成电路上可容纳的元器件的数目，约每隔18个月便会增加一倍，性能也将提升一倍。

阿姆达尔定律

## java内存区域

- ☑ 关于 stackOverflow
- ☑ C语言当中的方法调用, StackFrame
  - 两个推论:
    - 所有的进程都要一个main方法
    - 所有语言的方法调用本质上都是 栈
    - ☑ 堆上的 JIT编译技术，逃逸分析。栈上分析，标量替换
- ☑ C语言当中的溢出攻击
- ☑ 面试题，java调用jni方法，内存需要经过转换吗？StringBuilder
- ☑ 新生代，老年代，永久代，这样划分的意义是什么？
- ☑ TLAB java各个线程划分 缓冲区的意义
- ☑ 什么是常量，静态变量，字面量，符号引用，
- ☑ 方法区 究竟在存放什么

## 垃圾回收

### mark-sweep

### copying

适合新生代。分为 Edon控件，From survivor,to survivor空间。分配比例为 8:1  
protected void finalize() throws Throwable

# Class 文件

反编译 `javap -c TestClass` (不是 `-p`, 书上写错了)

- [x] 讨论计算机历史, 从汇编的角度来讲。
- [X] C语言为什么不能跨平台, C语言的编译过程
- CISC MISC
- 编程当中不能直接使用某个接口。。适配器设计模式。。。

## - 我们不能使用苏州的接口, 所以使用中间件

☐ 64K问题

☒ 为什么中间件语言比较慢 window visa, 不能适用于股票交易系统, 军事系统

# 汇编

举个最简单的例子, 分别使用这两种指令集计算 “1+1” 的结果, 基于栈的指令集会是这样子的:

```
1.  iconst_1
2.  iconst_1
3.  iadd
4.  istore_0
```

两条iconst\_1指令连续把两个常量1压入栈后, iadd指令把栈顶的两个值出栈、相加, 然后把结果放回栈顶, 最后istore\_0把栈顶的值放到局部变量表的第0个Slot中。

如果基于寄存器, 那程序可能会是这个样子:

```
1.  mov eax, 1
2.  add eax, 1
```

mov指令把EAX寄存器的值设为1, 然后add指令再把这个值加1, 结果就保存在EAX寄存器里面。

基于栈的指令集主要的优点就是可移植，但是速度较慢。