一、静态变量与局部变量的对比

1、参数表分配完毕之后，再根据方法内定义的变量的顺序和作用域分配。

2、我们知道类变量白哦有两次初始化的机会，第一次实在“准备阶段”,执行系统初始化，对类变量设置零值，另一次则是在“初始化”阶段，赋予程序在代码中定义的初始值。

3、和类变量初始化不同的是。局部变量表不存在啊系统初始化的过程，这意味着一旦定义了局部变量则必须人为的初始化，否则无法使用。

4、变量的分类：按照数据类型分：基本数据类型、引用数据类型

按照在类中声明的位置分：1、成员变量：在使用前都经历过默认初始化赋值

类变量：

在链接的准备阶段，给类变量默认值，然后再初始化阶段进行显式赋值，即是使用静态代码块赋值。对应到字节码指令就是clinit（）方法。

实例变量：

随着对象的创建，在堆区中开辟实例变空间，并进行默认赋值。

2、局部变量：不会给定默认值。在使用前必须要进行显式赋值，否则编译不通过。

3、形参变量：不用初始化也不能初始化

5、在栈帧中，与性能调优关系最为密切的部分就是前面提到的局部变量表。在方法执行时，虚拟机使用局部变量表完成的传递，

6、局部变量表中的变量也是重要的垃圾*回收根节点之一*，只要被局部变量表中直接或间接引用的对象都不会被回收。

7、还有局部变量表越大栈帧越大，栈帧占的内存以及数量可能会导致栈溢出。