## 2.思考：为什么JDK动态代理中要求目标类实现的接口数量不能超过65535个？

**Java 虚拟机规范规。**

**Class类文件结构**

　　Class 文件是一组以 8 位字节为基础单位的二进制流，各个数据项目严格按照顺序紧凑地排列在 Class 文件之中，中间没有添加任何分隔符，这使得整个 Class 文件中存储的内容几乎全部是程序运行的必要数据，没有空隙存在。

　　当遇到需要占用 8 位字节以上空间的数据项时，则会按照高位在前（Big-Endian）的方式分割成若干个 8 位字节进行存储。

        　　根据 Java 虚拟机规范的规定，Class 文件格式采用一种类似于 C  语言结构体的伪结构来存储数据，这种伪结构中只有两种数据类型：无符号数和表，后面的解析都要以这两种数据类型为基础，所以这里要先介绍这两个概念。

        　　无符号数属于基本的数据类型，以 u1、u2、u4、u8 来分别代表 1 个字节、2 个字节、4 个字节和 8 个字节的无符号数，无符号数可以用来描述数字、索引引用、数量值或者按照 UTF-8 编码构成字符串值。

       　　 表是由多个无符号数或者其他表作为数据项构成的复合数据类型，所有表都习惯性地以 “\_info” 结尾。表用于描述有层次关系的复合结构的数据，整个 Class 文件本质上就是一张表，它由表 6-1 所示的数据项构成。

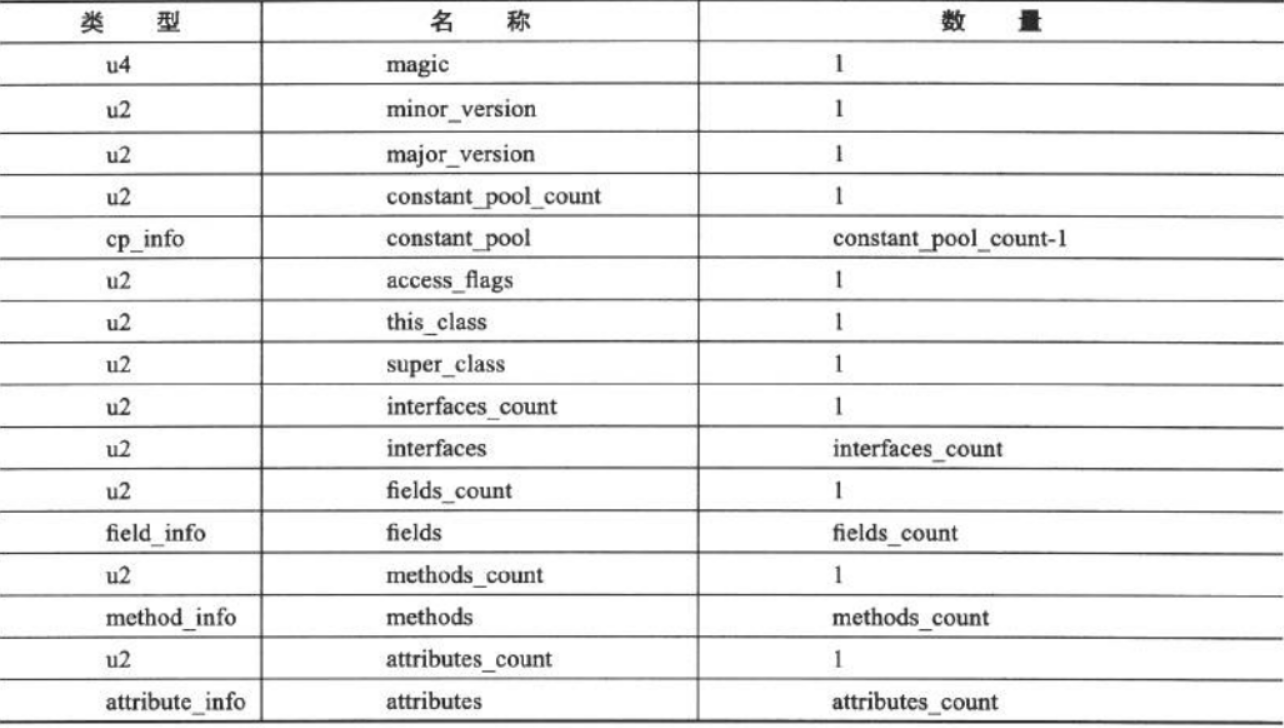


　　　　　　　　　　　　　　                                                     表6-1

  　　无论是无符号数还是表，当需要描述同一类型但数量不定的多个数据时，经常会使用一个前置的容量计数器，加若干个连续的数据项的形式，这时称这一系列的某一类型的数据为某一类型的集合。

　　Class的结构不像XML等描述语言，由于它没有任何分隔符号，所以在表6-1的数据项，无论是顺序还是数量，甚至于数据存储的字节序(Byte Ordering，Class 文件中字节序为 Big-Endian）这样的细节，

都是被严格限定的，哪个字节代表什么含义，长度是多少，先后顺序如何，都不允许改变。

根据class中interfaces\_count 接口计数器限定的是u2



interfaces\_count 的值表示当前类或接口的直接父接口数量。类型是u2， 2个字节，即 2^16-1 = 65536-1 = 65535

所以目标类实现的接口数量不能超过65535个