1. **Dalvik是Android4.4及之前使用的虚拟机，5.0以后ART虚拟机替代了Dalvik虚拟机。二者有哪些不同？**

Dalvik使用的是JIT（Just-In-Time）技术来进行代码转译，每次执行应用的时候，Dalvik将程序的代码编译为机器语言执行。

而ART(Android RunTime)虚拟机，其处理机制根本上的区别是：它采用AOT(Ahead-Of-Time)技术，会在应用程序安装时就转换成机器语言，不再在执行时解释，从而优化了应用运行的速度。在内存管理方面，ART也有比较大的改进，对内存分配和回收都做了算法优化，降低了内存碎片化程度，回收时间也得以缩短。

1. **在逆向Android程序时，string.xml文件和public.xml文件能够提供哪些线索？通过这些线索如何与smali代码（或翻译后的Java代码）产生关联？**

String.xml 文件中的所有字符串资源都在“gen/<packagename>/ R.java”文件的 String 类中被标识，每个字符串都有唯一的 int 类型索引值，使用Apktool反编译apk 文件后，所有的索引值保存在 string.xml文件同目录下的public.xml 文件中。

我们可以通过对public.xml下的索引值找到对应的调用该索引的函数位置，从而我们可以对对应的地方进行反汇编的操作分析该索引附近的逻辑结构，进行对我们想要的数据分析。

程序中用到的字符串会存储在String.xml文件或者硬编码到程序代码中，如果是前者的话，字符串在程序中会以id 的形式访问，只需在smali代码中搜索字符串的 id 值即可找到调用代码处；如果是后者的话，在smali代码中直接搜索字符串即可。

1. **列举smali语言中的数据类型与Java代码中的数据类型对应关系。**

Smali语言的数据类型和Java的数据类型对应如下：

Smali Java

